Sakarya University

Journal of Computer and Information Sciences

e-ISSN 2636-8129

VOLUME 7 ISSUE 3

DECEMBER 2024

VOLUME: 7 ISSUE: 3
E-ISSN 2636-8129

DECEMBER 2024
http://saucis.sakarya.edu.tr/tr/

SAKARYA UNIVERSITY JOURNAL OF COMPUTER AND INFORMATION SCIENCES

SAKARYA
ÜNİVERSİTESİ

Language Editors

Layout Editor

Indexing

Contents

Research Article

Review

SAKARYA UNIVERSITY JOURNAL OF COMPUTER AND INFORMATION SCIENCES
http://saucis.sakarya.edu.tr/
e-ISSN: 2636-8129
Publisher: Sakarya University
Vol. 7, No. 3, 2024, 325-337
DOI: 10.35377/saucis...1503018
SAKARYA UNIVERSITY
RESEARCH ARTICLE

Improving Deep Learning Forecasting Model Based on LSTM for Türkiye's Hydro-Electricity Generation

Mehmet Bulut

Electricity Generation Co. Inc., General Management Consultant, Ankara, Türkiye

Corresponding author:
Electricity Generation Co. Inc., General Management Consultant, Ankara, Türkiye,
mehmetbulut06@gmail.com

ABSTRACT
Electricity production in hydraulic power plants depends on the amount of water coming into the basin. This varies depending on precipitation such as snow and rain during the year, but when looking at the years, production is shaped according to the years when meteorological data are similar to each other. LSTM (Long Short-Term Memory) plays an important role in hydropower forecasting, as it is a special artificial neural network designed to model complex relationships on time series data, which is affected by various meteorological factors such as precipitation, temperature, and hydrological data such as water level, such as hydroelectric power production. Therefore, in this study, a forecast system based on the LSTM network model which is one of the deep learning methods was proposed for monthly hydropower-based electricity production forecast in Türkiye. The developed deep learning-based hydropower forecast model provides future production planning based on time series based on actual hydropower production data. Using real production data and LSTM learning models of different structures, monthly hydraulic electricity production forecasts for the next year were made and the models' performances were examined. As a result of this study, RMSE 32.4245 and MAPE 16.03% values and 200-layer LSTM model trained with 12-year data with 144 monthly data points containing hydroelectric generation information was obtained as the best model, and the performance values of the model showed that it was the correct forecasting model. The overall efficiency parameters of the found LSTM model were checked with NSE 0.5398 and KGE 0.8413 values. The performance of the model was found to be a high-accuracy model within acceptable limits and with a correlation value of R2 0.9035 to be very close to reality. The results obtained from this study have shown that deep learning models developed based on many years of production data give successful results in hydroelectric production prediction and can be used as a basis for electricity production planning.

Keywords: Hydroelectric power, Electricity production forecasting, deep learning, Long short term memory

Article History:
Received: 20.06.2024
Accepted: 06.09.2024
Published Online: 30.10.2024

1. Introduction

Research shows that renewable energy sources have the capacity to meet two-thirds of the world's total energy demand. They can also contribute to the drastic reduction of greenhouse gas emissions needed to keep the average global surface temperature rise below 2°C by 2050 [1]. Examples of major renewable energy sources are solar, wind, hydrogen, hydroelectric, wave and geothermal. The biggest environmental advantages of renewable energy sources are that they are renewable due to their continued existence in nature, they do not harm nature by less carbon emissions, and they are clean and sustainable energy sources [2]. Sunlight, wind speed and precipitation parameters, which form the basis of renewable energy sources, change seasonally

Discontinuous and intermittent renewable energy sources depend on meteorological weather conditions and parameters such as temperature and rainfall [3]. Estimating the amount of electricity to be obtained from hydroelectric power plants is of great importance in terms of planning production according to resources and ensuring the continuity of production [4]. The fact that renewable energy resources vary throughout the year and even during the day and depend on seasonal conditions from year to year makes it important to estimate production in the electricity grid to meet the supply. With the development of deep learning algorithms, it has achieved successful results in many areas, especially computer learning and prediction studies. In applications, deep learning models are created based on time series data, many of which are applied to real life, such as precipitation-flow modeling [5]. Deep learning network structures allow learning the complex relationships between the input and output sets of the learning structure and the complex relationships between the data [6]. Today, Long Short-Term Memory (LSTM) networks attract widespread attention and have many practical applications for time-series-based forecasting systems [7]. Wang et al. examined the methods used in estimating production based on renewable energy sources

Cite as: M. Bulut, "Improving Deep Learning Forecasting Model Based on LSTM for Türkiye's Hydro-Electricity Generation," *Sakarya University Journal of Computer and Information Sciences*, vol.7, no. 3, pp. 325-337, 2024. Doi: 10.35377/saucis…1503018

and the degree of accuracy in the predictions of these methods [8]. Cheng et al. focused on estimating the power demand needed in the network to accurately meet the demand in a network [9]. Other deep learning-based studies include predicting the amount of energy demanded monthly 3 days in advance [10], estimating the short-term electricity demand curve [11], design a new specific power demand prediction algorithm based on LSTM Deep Learning method with respect to end-user power demand patterns[12], using meteorological data of the last 24 hours. Methods have been developed for estimating the demand amount based on [13], and for predicting the next 3-day production based on meteorological data in a wind power plant [14]. They propose a power demand forecasting model based on a neural network. Li et al. tried improving the prediction performance based on deep learning by fusing different production time data components such as daily, weekly and long time of a hydraulic power plant [15]. Del Real et al., on the other hand, used a mixed deep learning architecture consisting of a convolutional neural network (CNN) combined with an artificial neural network (ANN) to perform energy demand estimation [16].

Hydroelectric generation is one of Türkiye's leading renewable energy sources [17]. The amount of electricity production from hydroelectric power plants depends on the amount of water coming to the dam basin. Production values change on a monthly basis depending on the amount of precipitation during the year, and the closer the water level is to the design values, the more efficiency in production increases. Accurate prediction of electricity supply based on data on generation resources connected to an electricity system provides a solution to meet the demanded power [18]. However, methods other than resource-based prediction can be used by using historical electrical load flow data in the electrical system [19, 20]. To create a prediction model with Türkiye's monthly electricity generation dataset, methods such as LSTM based on a deep learning algorithm were used using time series [21]. Renewable energy sources such as solar, wind and hydroelectricity mainly depend on local environmental and meteorological conditions such as temperature and precipitation-runoff rates. In many studies in the literature, local environmental and meteorological conditions such as temperature and precipitation-flow are used in the medium-term monthly forecasting of the amount of energy to be produced by Hydroelectric Power Plants (HPP). In this article, a deep learning model is applied/trained using time series to create a forecasting model using Turkey's monthly hydroelectric power plant production data set. Monthly Hydroelectric Generation Forecasting System was developed for forecasting hydroelectric production based on the LSTM Network-Based Deep Learning Model. Besides, as a result of global warming, increasing droughts and decreasing precipitation directly affect hydroelectric production, creating a decreasing effect on the generation of electricity. Hydroelectric energy production is based on converting the potential energy of water into kinetic energy, and the decrease in water resources due to drought will reduce the amount of energy produced by decreasing the water level in dams. Also, droughts make the flow regimes of rivers irregular, then Seasonal flow differences increase and water levels may drop unexpectedly. This negatively affects the efficiency of hydroelectric power plants [22]. In order to use water more effectively in drought conditions, water management strategies come to the fore, and it is necessary to ensure the sustainability of hydroelectric energy [23]. In this sense, the estimation studies of electricity production in the country's hydroelectric power plants play an important role in determining plans and strategies.

In the study, LSTM algorithm from artificial neural networks was used to predict the hydraulic electricity production of Türkiye. Compared to other artificial neural networks, LSTM's ability to hold data in memory increases its predictive power. By using this advantage, it is aimed to predict with high accuracy. As a result of the study, the amount of electricity that Türkiye will produce from hydraulic power plants is forecasted on a monthly basis annually, one year in advance, and production planning can be done more consistently in line with these estimated production values. When we look at the setup of this study, firstly, information about Turkey's hydroelectric production to be used in the study is given in the second section. Then, information about the creation of the model to be used in the model and the methods to be used to measure the performance of the models are shared. In the third section, the estimation results obtained from the models are given and the results are evaluated.

2. Material and Methods

2.1. Hydroelectricity Generation in Türkiye

Electricity generation from renewable energy sources in Türkiye shows an increasing trend every year in terms of hydroelectric and geothermal resources, especially wind and solar. Türkiye's installed electricity capacity reached 96,270 MW at the end of 2020. Türkiye's installed capacity has increased 3.5 times in 20 years. Renewable energy sources make up 51% of this total installed power, and the total installed power capacity of the plants based on renewable energy sources has reached 49,111 MW. By the end of 2020, a total of 305.4 billion kWh electricity was produced in Türkiye. As the share of resources in electricity generation, 45.90 billion kWh, which corresponds to 15%, was obtained from solar, wind and geothermal power plants, and 78.12 billion kWh, which corresponds to 25.6%, was obtained from hydraulic power plants. In 2020, the share of renewable energy sources in electricity generation was 42.4% [24, 25]. For Türkiye the hydraulic energy installed power as shown in Figure 1.

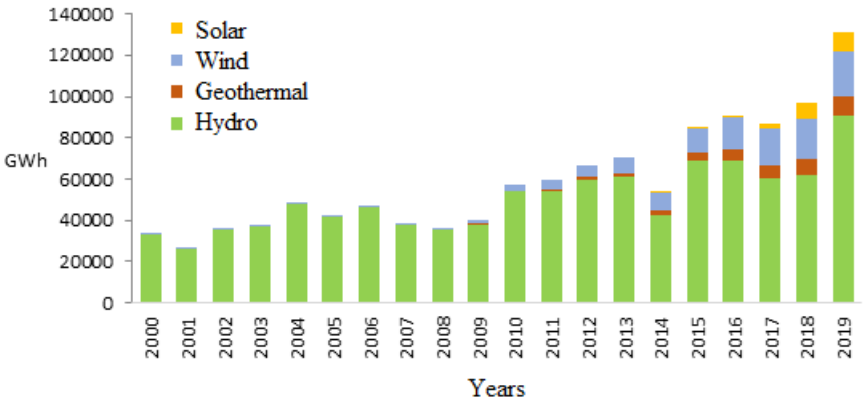


Figure 1. Electricity Generation from Various Renewable Sources Including Dam Type Hydro

Türkiye's hydroelectric production varies in some years, showing a continuous upward trend in installation and generation, consistent with total electricity generation [24]. Dataset used in this study for forecasting of hydro electrical generation is taken from the statistics section of the publicly available Load Dispatch Information System (YTBS) web portal [26]. Considering the 12-year production covering the hydroelectric production data of Türkiye in between 2007–2018, the minimum, maximum and average production values of the productions corresponding to the months between January and December are given in Figure 2. It is seen that the highest production are realized in May, and the lowest production was in October.

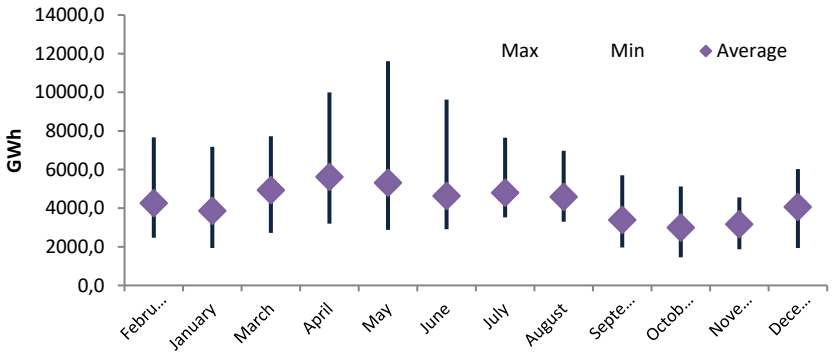


Figure 2. Monthly Maximum, Minimum and Average Generation Values of Hydroelectric Power Plants on a Long-Term Basis

2.2. Structure of Long Short-Term Memory (LSTM) Networks

Artificial Neural Networks (ANNs) represent learning algorithms developed based on the structure and functioning of human neural networks based machine learning. Deep learning (DL) networks, on the other hand, have been presented as a solution to many complex artificial intelligence (AI) problems that have existed for many years. In fact, DL models are described as deeper variants of linear or nonlinear multilayer artificial neural networks (ANNs).

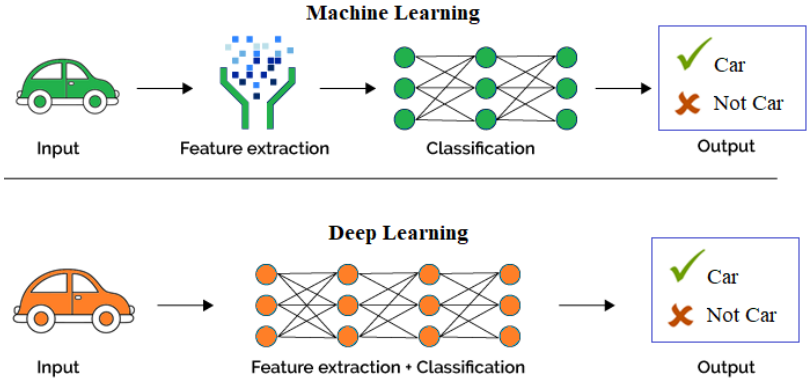


Figure 3. Difference Between Machine Learning and Deep Learning Approximation

Difference between machine learning and deep learning approximation is shown Figure 3. The ability of DL models to learn

hierarchical properties from various data types makes them powerful in recognizing, regressing and solving semi-supervised and unsupervised problems [27]. ANNs can predict whether the image is a square or an equilateral rectangle. However, due to the fact that ANNs do not carry memory, they are insufficient in many cases where prior data information is important due to the nature of some problems. LSTMs can remember information for a long time. Many variations have been developed to solve the problem posed in a deep Recursive Neural Network (RNN). To solve this problem, LSTM uses gate units to decide what information to keep or remove from the previous state [28, 29]. A LSTM Network consists of three different Gates, they are Forget gate, Input gate and Output gate. Gates control the flow of information from memory to memory [30].

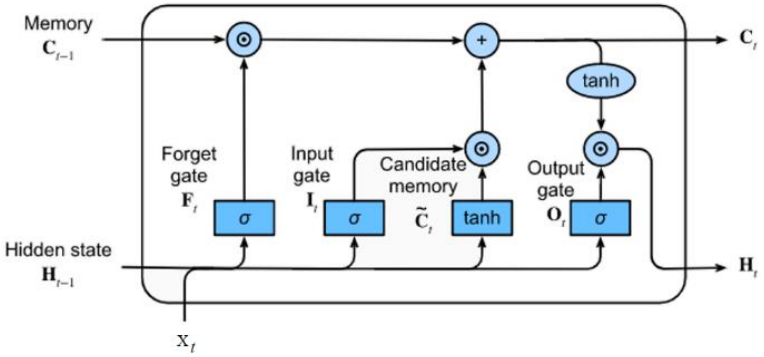


Figure 4. Gate Connections of an LSTM Cell [30]

The core component of LSTMs is a memory cell that can hold information over time controlled by gates. It can maintain its state over time, consisting of an open memory (also called a cell state vector) and gate units. Gate units regulate the flow of information entering and leaving the memory [31, 32]. The input gate is used to check whether the state in the current cell allows it to be overridden by external information as shown in Figure 4. The entrance gate in the LSTM cell structure used in the study is shown in Equation 1.

$$i_t = \sigma_k(W_i x_t + U_i h_{t-1} + b_i) \quad (1)$$

Here;
i_t : inlet gate vector,
σ_k : sigmoid function,
x_t: input vector,
W_i and U_i : matrices to parameter
b_i: bias vector.

The output vector decides whether to keep the state in the current cell, which will affect other cells, and is defined as in the Equation 2.

$$o_t = \sigma_k(W_o x_t + U_o h_{t-1} + b_o) \quad (2)$$

Another cell gate defined in the LSTM memory cell is the forgotten container, which enables the state of the LSTM to be reset and is defined as follows.

$$f_t = \sigma_k(W_f x_t + U_f h_{t-1} + b_f) \quad (3)$$

As a result, Equations 4 and 5 shows how the cell state and the output vector is revealed from the input gate, forget gate and output gate.

$$c_t = f_t \odot c_{t-1} + i_t \odot tanh\left(W_f x_t + U_c h_{t-1} + b_c\right) \quad (4)$$

$$h_t = o_t \odot \sigma_h(c_t) \text{ or } h_t = o_t \odot tanh(c_t) \quad (5)$$

Here, $\odot$ represents the Hadamart product, σ_c and σ_h represents the hyperbolic tangent functions.

2.3. Forecasting Time Series of Hydroelectric Generation Model

The hydroelectric generation estimation study consists of the following steps. In this section, the hydroelectric generation estimation framework made with LSTM is given in Figure 5.

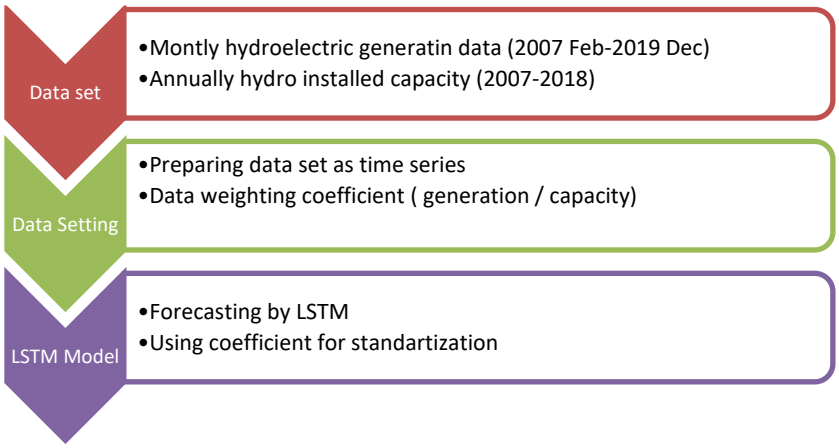


Figure 5. LSTM Hydropower Generation Forecasting Framework

It can be express the definition of the hydroelectric production estimation problem considered in this study as follows:
1- Obtaining a hydroelectric generation time series by using the monthly total electricity generation values realized in hydroelectric power plants for many years.
2- Using this obtained time series, forecasting annual hydroelectric generation monthly in the future.
3- Extract the internal feature to help better predict hydropower generation: Using the 12-year hydropower generation, the monthly correction coefficient is obtained by obtaining the average hydropower generation for each month of the year during January-December and standardizing them in a max-min range.
4- Historical observations of previous t - 1 time steps $Xt-1 = \{X1, X2, \cdots, Xt-1\}$ and internal features D(y): January;...;December, considering the hydroelectric generation forecast, is to learn a model that predicts electricity generation value. Time t, that is, in time step t of the electricity generation value X t {Xt | Estimating Xt <1, Ft}.

$$Xn(t)=Xn(1), Xn(2),...., Xn(t-1),..., Xn+1(1), Xn+1(2),....,Xn+1(t-1),...XN(1), XN(2),...,XN(t-1) \quad (5)$$

Here;
n=1..N, N=12 (January,…,December)
t = number of past years,
X(n+1)(t)=n in year t. hydroelectric production of the month.

The production of a dam hydroelectric power station is affected by keeping the water level too high or too low due to the change in water flow. Similarly, the generation of river or canal hydroelectric power plants depends on the water regime affected by precipitation. The most important parameter showing the generation characteristics of hydroelectric power plants is the capacity factor. The capacity factor shows how many hours the hydraulic plant produces in 8765 hours of the year and is used for performance comparison. Net plant capacity factor (CF) is referred to as full capacity plant can produce energy part of the total energy generated in a given period.

$$CF = ANP / (IP *365 days*24 hours/day) \quad (6)$$

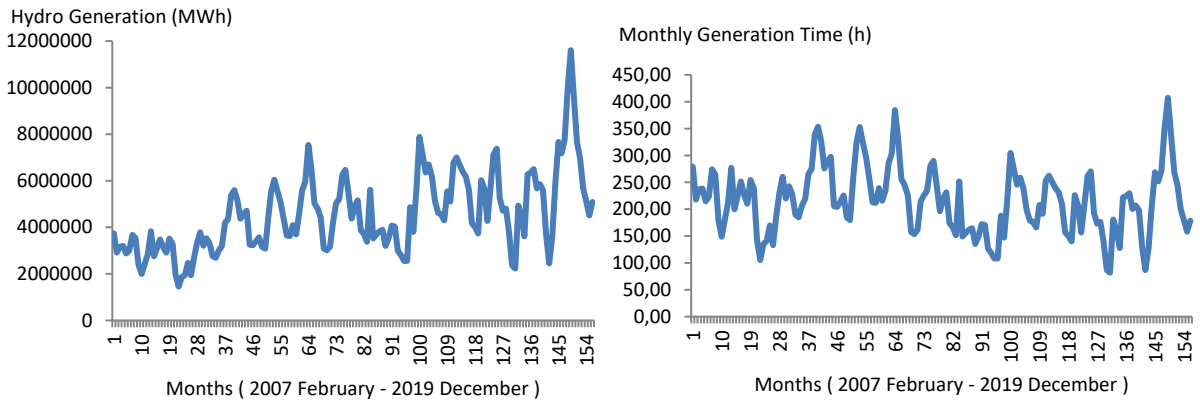
Here, ANP is annual electricity production and IP is installed capacity.

In hydroelectric power plants, the average capacity factor is around 40%. In other words, when the annual maximum operating time is taken into account as 8760 hours, hydroelectric power plants can only produce for 3500-4000 hours per year because the water coming to the reservoir is not always continuous. This means that hydroelectric power plants can only produce for 40% of the annual time period.

In this situation; Since the hydroelectric production data used in this study covers a period of 13 years, the installed capacity value that provides the production for each year is not the same due to each hydraulic installed capacity increase. To save the hydroelectric generation time data from this installed capacity value change, the time series based on monthly generation performance are obtained using the following formula (Figure 6. a, b).

$$GP = MEP/IP \quad (7)$$

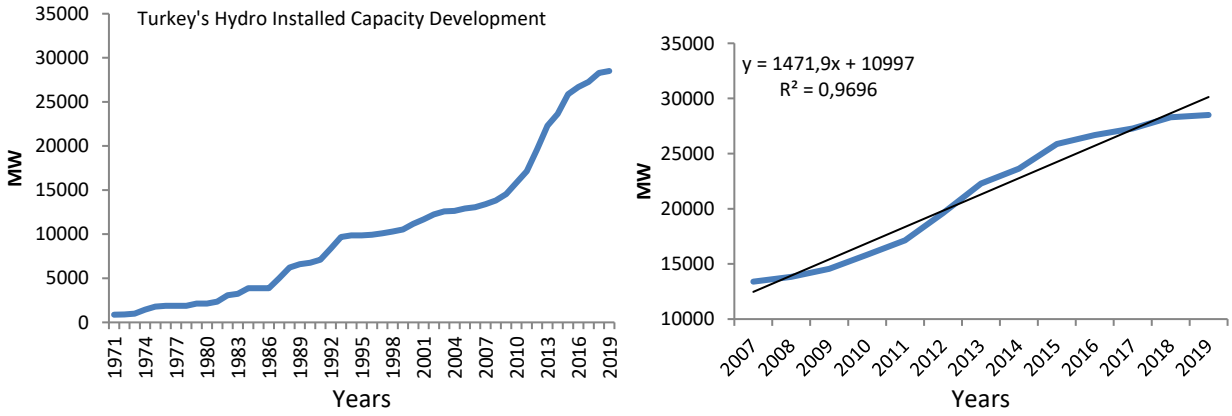
Here, GP is generation performance and MEP is monthly electricity production. Two separate data inputs explained below are fed into the LSTM Based hydroelectric system.



(a) (b)

Figure 6. a) Monthly Hydroelectric Production B) Data Set Standardization as Monthly Hydro Generation Time

One is monthly hydroelectric generation covering a 12-year time interval (Figure 6-a) and second is hydroelectric installed capacity value for these years as shown in Figure 7-a. The monthly estimated standardized hydroelectricity production values as generation duration (Figure 6b) are combined with the annual hydroelectricity estimation data, using the installed power values obtained from the regression estimation model in Figure 7-b.



(a) (b)

Figure 7. a) Graphic of Hydro Installed Capacity Data Set, B) Regression Model for Hydro Installed Capacity Data Set

Data preparation and calculation of final forecast production value;

1- Since the total installed capacity providing hydraulic electricity generation is different for each year, the annual hydraulic electricity production monthly per MW capacity must be standardized. To this end, monthly hydroelectric production (MWh) is proportioned to the total installed capacity value (MW) of the taken year. Then, monthly time curves are produced based on the number of hydroelectric production hours for each month.

2- Since the annual hydroelectricity are standardized using the annual installed capacity value, the estimated year's values are multiplied by the estimated year's total installed capacity value to obtain the monthly hydroelectricity production (MWh) estimated values

$$EMP = EPT * IP \tag{8}$$

Here, EMP is estimated monthly production as unit of MWh and EPT is estimated production time as unit of hours.

2.4. Performance Measurement and Evaluation of the Prediction Models

Hydroelectric production estimation is of great importance in terms of efficiency and reliability of energy systems, and the accuracy of these estimations allows more accurate decisions to be made in energy planning. Therefore, various criteria are important to evaluate the estimation performance. Since hydroelectric production estimation performance criteria are important for evaluating the accuracy and reliability of the model, choosing the right criterion is of great importance for developing the model and managing energy systems more effectively.

In this study, multi-criteria evaluation are performed using more than one criterion instead of a single criterion for performance evaluation. As Hydroelectric Production Estimation Performance Criteria in the study; Mean Absolute Percentage Error (MAPE), which is useful when making comparisons for data sets of different scales, Root Mean Squared Error (RMSE), which gives more weight to large errors and is used to evaluate the overall performance of the model, Coefficient of Determination (R), which shows how well the model explains the data, where the R-squared value varies between 0 and 1. As it approaches the value of 1, the explanatory power of the model increases. In addition, the Nash-Sutcliffe Efficiency Coefficient (NSE), which shows how well the model simulates the observed data, varies between -∞ and 1. The performance of the model increases as the value approaches 1, the KGE (Kling-Gupta Efficiency) method, which is a statistical measure frequently used in hydrological modeling and can evaluate how well the model fits the observed data, especially in time series data such as streamflow forecasting, is used to evaluate how well the model performs in terms of both its mean value and variance and shape.

Average absolute Percent Error (MAPE) and Root Mean Square Error (RMSE) metrics are used to select the best prediction model with the smallest estimation error by using the hydroelectric monthly generation data time series. RMSE and MAPE values were used to compare the prediction accuracy performances of different long short-term memory structures of models belonging to different time series and to measure the results obtained. RMSE is a quadratic scoring rule that also measures the mean magnitude of error and is defined as follows.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(v_{estimated} - v_{real})^2} \tag{9}$$

MAPE is a widely used statistical method that measures how close the forecast result made by a forecasting system is to the truth. It measures accuracy as a percentage. It is defined as follows

$$MAPE = \left(\frac{1}{N}\sum\frac{|v_{real} - v_{estimated}|}{|v_{real}|}\right) * 100 \tag{10}$$

And, to check the overall efficiency of the LSTM models which are used for the prediction of hydro generation, Nash-Sutcliffe Efficiency (NSE), Kling-Gupta Efficiency (KGE) and coefficient of determination (R^2) metrics are used. NSE coefficient is used to measure the accuracy of many hydrological predictions, determining the relative magnitude of the persistent variance compared to the variance of the observation data. NSE is calculated by Equation 11,

$$NSE = 1 - \frac{\sum_{i=1}^{N}\left(X_{i=1}^{obs} - X_{i=1}^{cal}\right)^2}{\sum_{i=1}^{N}\left(X_{i=1}^{obs} - X^{avg}\right)^2} \tag{11}$$

Where; X_i^{obs} is the ith value of the observed monthly flows, X_i^{cal} is the ith value of the calculated monthly flows, X^{avr} is Average of observed monthly flows and N represents the total number of observations. NSE ranges from $-\infty$ to 1. Here, NSE=1 proves that the method is physically excellent. A value between 0 and 1 for NSE generally indicates that the method performance is acceptable. The value is less than 0 indicates that the method performance is insufficient.

KGE is originally developed to compare the modelled and observed time series. KGE is a model evaluation criterion that can be differentiated in the contribution of mean, variance and correlation to model performance. KGE range from $-\infty$ to 1. KGE = 1, indicating excellent agreement between simulations and observations. The KGE score for mean flow comparison in hydrological models is KGE ≈ -0.41. The closer the KGE value is to 1 in performance measurement, the more accurate the model will be.

$$KGE = 1 - \sqrt{(r-1)^2 + \left(\frac{\sigma_{cal}}{\sigma_{obs}} - 1\right)^2 + \left(\frac{\mu_{cal}}{\mu_{obs}} - 1\right)^2} \tag{12}$$

Where; $\alpha = \frac{\sigma_{cal}}{\sigma_{obs}}$, $\beta = \frac{\mu_{cal}}{\mu_{obs}}$ and where $(\mu_{cal}, \sigma_{cal})$ and $(\mu_{obs}, \sigma_{obs})$ are the mean and standard deviation of estimation and observation. KGE calculates the Euclidean distance (ED) of the three components from the ideal point. This ideal point avoids underestimation of variability and enables comparison of the term bias across monthly. Like NSE, KGE=1 indicated perfect connection between estimations and observations.

R^2 measures the strength of the linear relation between x and y pairs, and the results are expected to be between 0 and 1. The closer the result is to 0, the more the model diverges from reality. The Equation 13 of R^2 is given below.

$$R^2 = \left(\frac{(n\sum xy) - (\sum x)(\sum y)}{\sqrt{n\sum x^2 - (\sum x)^2}\sqrt{n\sum y^2 - (\sum y)^2}}\right)^2 \tag{13}$$

3. Results and Discussions

In this study, a data set covering monthly hydroelectric production values of Türkiye for the period January 2007-December 2018 was used. Using long-term monthly hydroelectric generation information, a forecasting model based on LSTM networks has been developed that predicts 12-month hydraulic production annually. The block diagram of the LSTM-based deep learning hydroelectric generation system aimed in this study is given in Figure 8. When estimating hydroelectricity generation, first, the generation dataset is standardized by proportioning the annual installed capacity value to the boxed capacity, and after training the LSTM deep network with this generation-based time series, to convert the monthly-based generation forecast values of the next year to the generation values in MWh, the hydroelectric generation board of that year. The capacity value was estimated by the regression model. To make an accurate estimation of the regression model used here high R^2 monovalent annual capacity value of the last five years to achieve linear function it is used.

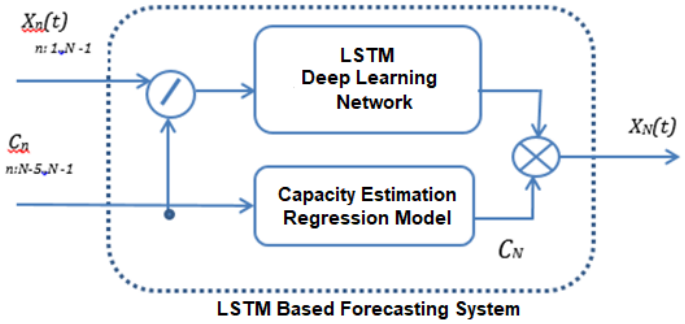


Figure 8. Block Diagram Structure of Proposed the Forecasting System

The data used in this study are taken from the public web page of Türkiye's Transmission System Operator. The results obtained in this study were obtained by running the LSTM algorithm under the Matlab program. Data set used in this study, the monthly value by Türkiye's electric transmission system operator in the country (Transmission System Operator-TSO) has established is provided from open sources [25-27]. This data set, starting from January 2007, includes a total of 144 months of hydroelectric production information, covering the month of December 2018. This data set is divided into three parts and designed to work with three LSTM estimated models. Detailed description of the data set used in study are shown in Table 1.

Table 1. Characteristics of the Dataset

Data set	Time Range/Feature	Parameter
Hydropower generation time series (MWh)	01.2007 – 12.2018	n*t number of months 144 pieces
Data time series 72 months: 120 months: 144 months:	 01.2007 – 12.2012 01.2007 – 12.2016 01.2007 – 12.2018	*Xn(t)*, production value in t. month of n. year
Installed Capacity (MW)	2007 - 2018	Annually: 12 pieces

In order to find the best LSTM model, LSTM models with different number of layers were used on data with different lengths of time intervals. The effect of the number of layers on the estimation of the model was measured by choosing the number of layers starting from 25 and increasing up to 400. In the estimation system, three different data with monthly production values as 6 years, 10 years and 12 years were used. In the data sets used, 72 data points for 6 years, 120 data points for 10 years data set and 144 data points for 12 years data set were used. In these data sets, 1 year of data is reserved for validation. In the remaining data set, the 1-year data set was used for testing and the remaining data was used for training. These data are given in Table 2, which shows the results for each model. Although LSTM models use the same estimation system, they differ from each other according to the data set used for training. Accordingly, the LSTM-1 model was trained using a 72-month training set, the LSTM-2 model using a 10-month training set, and the LSTM-3 model using a 12-month training set. For all LSTM models, versions consisting of 25, 50, 100, 200 and 400 layers were used and results were obtained. In the study, the graphs of the estimation results with the number of layers that give the best result out of the estimation results obtained for the LSTM estimation models with 6, 10 and 12 year production time series in the estimation system are given. LSTM networks with 25, 50, 100, 200 and 400 hidden layers were used in 3 different models, each of which was operated with monthly production data of 72 months, 120 months and 144 months. An LSTM layer includes an RNN layer that learns the long-term dependencies between the time steps in the time series and the sequence data.

As the first LSTM-1 Model; covering the year 2007-2012 and 72 monthly time series of hydroelectric production was discussed. According to the results obtained from the LSTM-1 model and given in Table 2, the model obtained using the 400-layer long-short-term memory structure gave the best results and is presented in Figure 9.

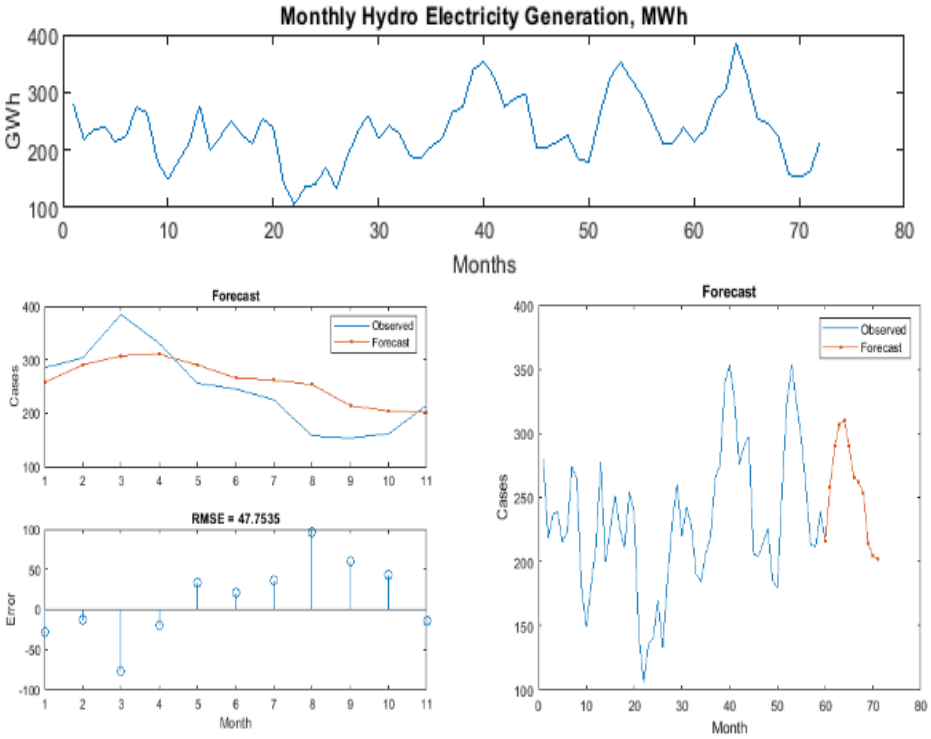


Figure 9. Forecast Results, Error Graph and RMSE Values of 400 Layers Using 72-Month Production Data Series

As the second LSTM-2 Model; the 120-month hydroelectric generation time series covering the years 2007-2016 is discussed. According to the results obtained from the LSTM-2 model and given in Table 2, the model obtained using the 400-layer long-short-term memory structure gave the best results and is presented in Figure 10.

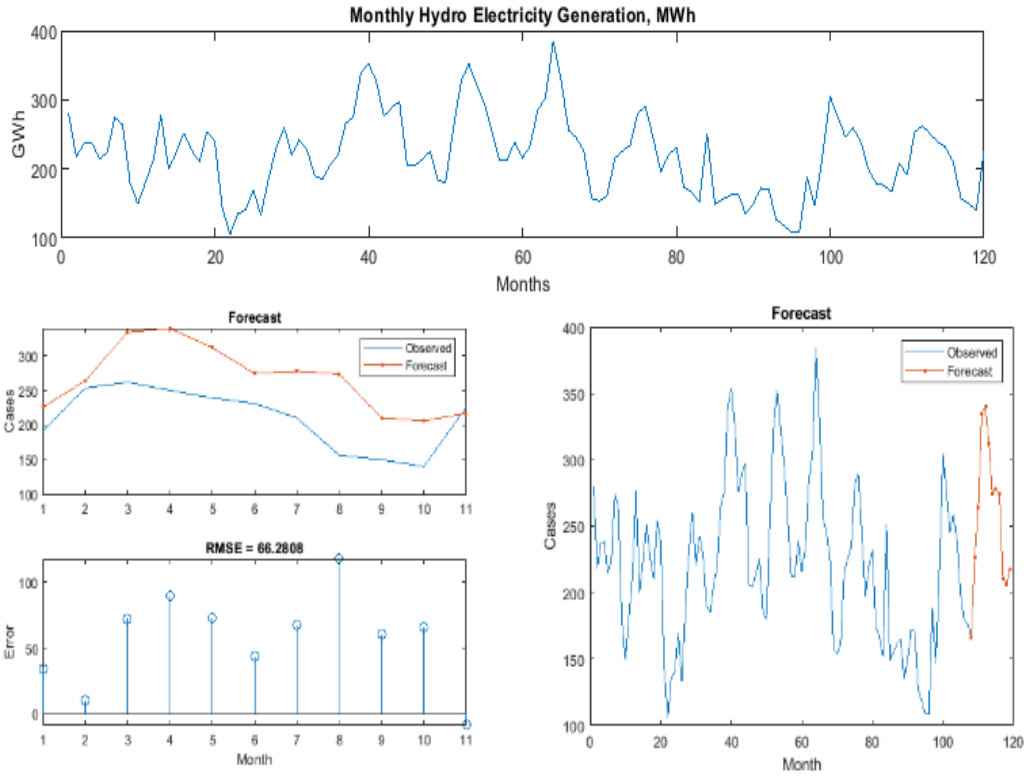


Figure 10. Forecast Results, Error Graph and RMSE Values of 400 Layers Using 120-Month Production Data Series

As the third LSTM-3 Model; the 144-month hydroelectric generation time series covering the years 2007-2018 is discussed. According to the results obtained from the LSTM-3 model and given in Table 2, the model obtained using the 200-layer long-short-term memory structure gave the best results and is presented in Figure 11.

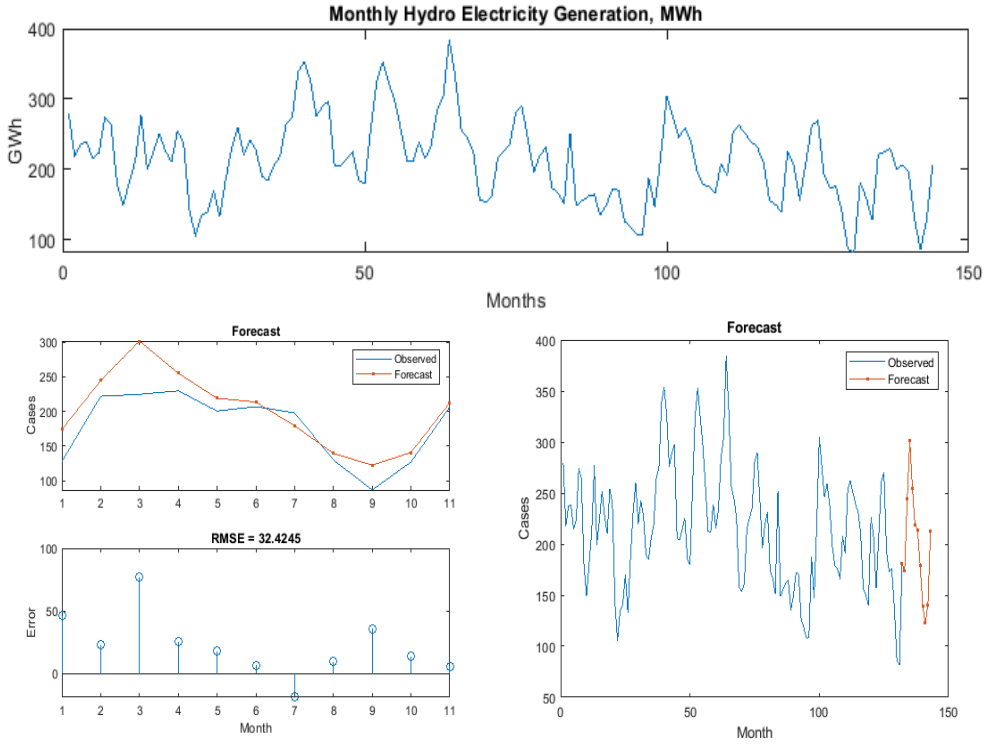


Figure 11. Forecast Results, Error Graph and RMSE Values of 200 Layers Using 144-Month Production Data Series

When the performance values and model efficiency of these models developed to estimate hydraulic production are evaluated; according to the results obtained in Table 2, which includes the RMSE and MAP values of the results obtained in the study; It is seen that the 200-layer LSTM-3 model, which includes 12 years of hydroelectric time data and 144 data points on a monthly basis, is the most predictive model with an annual RMSE of 32.4245 and an annual MAPE of 0.1603. Looking at the results in the Table 2, it is seen that the number of layers of the model with the best estimate for each LSTM model is close to each other. When 400 layers were used for LSTM-1 and LSTM-2, more suitable results were obtained for the prediction values compared to the others. However, the LSTM-3 model, which gave the best results, was found to have 200 layers. RMSE=32.4245 and MAPE=16.03% values and 200-layer LSTM model trained with 12-year data with 144 monthly data points containing hydroelectric generation information was obtained as the highest model, and the performance values of the model showed that it was the correct forecasting model. The overall efficiency parameters of the found LSTM model were checked with NSE=0.5398 and KGE=0.8413 values, the performance of the method was found to be a high-accuracy model within acceptable limits and with the correlation value of R2=0.9035 to be very close to reality. The results showed that the LSTM based forecasting model can be used as an acceptable hydropower generation forecasting model.

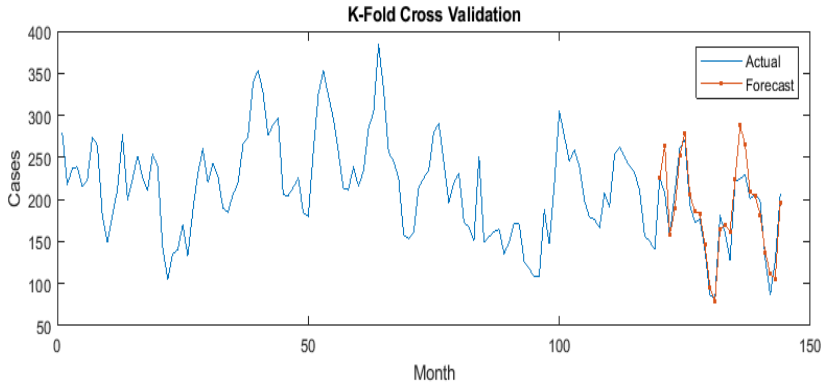


Figure 12. Results of K-Fold Validation for Best LSTM Model, which is 200 Layers Using 144-Month Production Data Series

The k-fold cross-validation method separates the data into equal parts according to the specified k number and ensures that each part is used for both training and testing, thus minimizing deviations and errors caused by scattering and fragmentation. The method was applied for the 200-layer LSTM model using a 144-month data set with the best performance and efficiency values. Statistical validation was made to analyze the predictions of LSTM, which was chosen as the most successful model for the results obtained, the data set was divided into 6 parts, 5 parts for training and one set for testing. The result obtained to show the performance of the model is shown in Figure 12 and it is seen that the selected model produces a value suitable for the prediction.

Table 2. Model Efficiency Metrics for Epoch Number=500

Model No	Time Interval (months)	Training Part (months)	Testing Part (months)	Layer Number	Model Evaluation		Efficiency of Model		
					RMSE	MAPE (%)	NSE	KGE	R^2
LSTM-1	72	60	12	25	57.7255	0.2265	0.3470	0.3350	0.7203
				50	70.0400	0.2450	0.0387	0.3407	0.5704
				100	58.6001	0.2025	0.3271	0.4170	0.6689
				200	52.3823	0.2124	0.4623	0.4041	0.7448
				400	**47.7535**	**0.1919**	**0.5531**	**0.5197**	**0.7944**
LSTM-2	120	108	12	25	72.2326	0.3422	-1.9366	0.5662	0.8679
				50	69.2135	0.3261	-1.6963	0.6709	0.8819
				100	68.4791	0.3108	-1.6394	0.7124	0.8062
				200	75.8111	0.3447	-2.2348	0.6012	0.8174
				400	**66.2808**	**0.3031**	**-1.4726**	**0.7141**	**0.8321**
LSTM-3	144	132	12	25	47.1694	0.2762	0.0260	0.7445	0.8368
				50	43.5890	0.2669	0.1683	0.5146	0.6305
				100	47.0654	0.2905	0.0303	0.7782	0.8367
				200	**32.4245**	**0.1603**	**0.5398**	**0.8413**	**0.9035**
				400	36.8064	0.2298	0.4070	0.6516	0.8938

RMSE) and MAPE metrics were used to analyze the performance of three learning algorithms based on 72-month, 120-month and 144-month generation hydroelectric generation data used in the study. The best prediction was obtained in the 200-layer LSTM model using 144 months (12 years) hydroelectric generation time data with the lowest MAPE percentage and lowest RMSE value. The NSE, KGE values of this model were higher than 0.5, close to the ideal 1 value, and the coefficient of correlation value (R^2) was found to be satisfactory with a value of 0.9035 in terms of estimation efficiency.

The more data available, the more accurate the predictions are, the more accurate the results are demonstrated here. However, if a similar study is conducted on a larger training set covering longer years, it is predicted that the algorithms used for the proposed estimation system will yield better results in their performance.

4. Conclusions

As a result of global warming, increasing droughts and decreasing rainfall directly affect hydroelectric production, water management strategies come to the fore in order to use water more effectively in dams, and it is necessary to ensure the sustainability of hydroelectric energy. Hydroelectric production is a process that is constantly changing over time and is changing with changing global warming. LSTMs are naturally designed to work with time series data, allowing them to better understand how data changes over time and more accurately predict future values. It was preferred in this study for hydropower forecasting due to its features such as capturing long-term dependencies and being robust to noise.

Electricity production from renewable energy sources depends mainly on meteorological conditions such as temperature, humidity, wind speed and rainfall in the geography where the facility is located. Therefore, due to the intermittent nature of renewable resources, hydroelectricity production depends on the amount of rainfall and the amount of incoming water, and due to the fluctuating nature of the production, it is important to estimate the hydroelectricity production to be provided to the electricity grid. As in many countries in the world, electricity generation from hydroelectric sources in Turkey is among the important renewable energy sources, and at times approximately 30% of the country's electricity production is provided by hydropower plants. In this sense, it is evaluated that the deep learning-based models proposed in this article will contribute to the studies on production estimation of hydroelectric power plants, which have an important share in our country's electricity production.

This study aims to choose the best LSTM for energy estimation, its performance in the overall model structure of LSTM is analyzed. For this, LSTM models with different training data sets and different number of layers are designed. The performances of the LSTM based on these criteria were given comparatively on a table and the model with the best result was tried to be determined. This study focuses on the potential of using deep learning LSTM to forecast annual hydroelectric power demand on monthly basis. For this, an estimation system based on annual hydroelectricity installed capacity development with monthly hydroelectricity production time data has been proposed. The dataset used in this study was divided into three parts 6, 10, and 12 years, estimation was conducted with three different LSTM models and the effect of the dataset lengths on the prediction was tried to be observed. Thus, three different LSTM models were created with datasets with separate time intervals and predictions were made. A the acceptability level of the dataset used to train the model was investigated to obtain satisfactory prediction results, and it was aimed to observe the short, medium and long-term prediction performance of the production dataset. According to the study, the results of the research show that LSTM provides a robust architecture for the prediction of hydropower production in medium or long-term forecasts such as at least 120 months and 144 months. It is observed that the LSTM network-based forecasting system was successful within the acceptance limits by using the time series data of hydroelectric generation and the installed power values.

References

[1] D. Gielen, F. Boshell, D. Saygin, M.D. Bazilian, N. Wagner, R. Gorini, "The role of renewable energy in the global energy transformation", *Energy Strategy Reviews*, 24: 38-50, 2019.

[2] P.A. Owusu, S. Asumadu-Sarkodie, "A review of renewable energy sources, sustainability issues and climate change mitigation", *Cogent Engineering*, 3(1):1167990, 2016.

[3] M.O. Abdeen, "Energy use and environmental impacts: A general review", *Journal of Renewable and Sustainable Energy*, 1:053101, 2009.

[4] H. Alrayess, Gharbia S., Beden N., A. Ulke, "Using Machine Learning Techniques and Deep Learning in Forecasting the Hydroelectric Power Generation in Almus Dam, Türkiye", *5th International Symposium on Dam Safety*, Istanbul,Türkiye, Oct. 27-31, 2018.

[5] F. Kratzert, Klotz D., Brenner C., Schulz K., M. Herrnegger, "Rainfall–runoff modelling using Long Short-Term Memory (LSTM) networks", *Hydrology and Earth System Sciences*, 22: 6005–6022, 2018.

[6] K. Amarasinghe, Marino, D.L., Manic, M. "Deep neural networks for energy load forecasting", *IEEE 26th Int. Symposium on Industrial Electronics (ISIE)*, Edinburgh, 1483-1488, 2017.

[7] Choi, J.Y., Lee, B, "Combining LSTM Network Ensemble via Adaptive Weighting for Improved Time Series Forecasting", *Hindawi Mathematical Problems in Engineering*, 2470171:1-8, 2018.

[8] Wang, H., Lei, Z., Zhang, X., Zhou, B. and Peng, J. (2019). A review of deep learning for renewable energy forecasting", *Energy Conversion and Management*, 198:111799.

[9] Y. Cheng, Xu C., Mashima, D., Thing, V., and Y. Wu, "PowerLSTM: Power Demand Forecasting Using Long Short-Term Memory Neural Network", In: Cong G., Peng WC., Zhang W., Li C., Sun A. (eds) *Advanced Data Mining and Applications.* ADMA, 727-740, 2017.

[10] N. Al Khafaf, Jalili M., P. Sokolowski, "Application of Deep Learning Long Short-Term Memory in Energy Demand Forecasting", in: Macintyre J., Iliadis L., Maglogiannis I., Jayne C. (eds) *Engineering Applications of Neural Networks.* EANN, 31-42, 2019.

[11] J. Zheng, Cencen, Xu, Zhang, Z., and L. Xiaohua, "Electric load forecasting in smart grids using Long-Short-Term-Memory based Recurrent Neural Network*", 51st Annual Conference on Information Sciences and Systems (CISS)*, Baltimore, MD, 1-6, 2017.

[12] E. Choi, Cho, S., D.K. Kim, "Power Demand Forecasting Using Long Short-Term Memory (LSTM) Deep-Learning Model for Monitoring Energy Sustainability", *Sustainability*, 12:1109, 2020.

[13] D. Altunkaya, Yilmaz, B., "Multivariate Short-term Load Forecasting Using Deep Learning Algorithms", the Eurasia Proceedings of Science, Technology, *Engineering & Mathematics (EPSTEM)*, 11:14-19, 2020.

[14] T.G. Barbounis, Theocharis J.B., Alexiadis M.C., and Dokopoulos P.S., "Long-term wind speed and power forecasting using local recurrent neural network models", *IEEE Transactions on Energy Conversion*, 21(1):273-284, 2006.

[15] Li, L., Yao, F., Huang, Y., and Zhou, F., "Hydropower Generation Forecasting via Deep Neural Network", *2019 6th International Conference on Information Science and Control Engineering (ICISCE)*, Shanghai, China, 324-328, (2019).

[16] R.Torres, Dorado, F., and Durán, J., "Energy Demand Forecasting Using Deep Learning: Applications for the French Grid", *Energies*, 13(9):2242, 2020.

[17] O. Yuksek, Komurcu, M.I., Yuksel, I., and K. Kaygusuz, "The role of hydropower in meeting Türkiye's electric energy demand", *Energy Policy*, 34:3093–3103, 2006.

[18] I. Kollia and Kollias, S., "A Deep Learning Approach for Load Demand Forecasting of Power Systems", *IEEE Symposium Series on Computational Intelligence (SSCI)*, India, 912-919, 2018.

[19] He, W., "Load Forecasting via Deep Neural Networks", *Procedia Computer Science*, 122:308-314, 2017.

[20] M. Bulut, Tora H. and B. Magdi, "Comparison of Three Different Learning Methods of Multilayer Perceptron Neural Network for Wind Speed Forecasting", *Gazi University Journal of Science*, 34.2: 439-454, 2021.
[21] R. Ünlü,"A Comparative Study of Machine Learning and Deep Learning for Time Series Forecasting: A Case Study of Choosing the Best Prediction Model for Türkiye Electricity Production", *Suleyman Demirel University Journal of Natural and Applied Sciences*, 23:635-646, 2019.
[22] S. Bernstein, J. Spring and D. Stanway, Insight: Droughts shrink hydropower, pose risk to global push to clean energy, August 14, 2023, https://www.reuters.com/business/sustainable-business/inconvenient-truth-droughts-shrink-hydropower-pose-risk-global-push-clean-energy-2021-08-13/ [Accessed : 05.08.2024].
[23] F. A. Kaplan, The Effect of Climate Change On Hydroelectric Power Plants and Hepp, Ges Hybrid Model, Dsi Technical Bulletin Issue: 141, January 2023
[24] M. Bulut, İzzet A. "Prospects of renewable energy penetration to Türkiye's electric power grid", *International Journal of Global Warming* 25:3-4, 366-377, 2021.
[25] Türkiye Electricity Transmission Inc., (TEİAŞ) https://www.teias.gov.tr/turkiye-elektrik-uretim-iletim-istatistikleri. [Accessed : 22.09.2020].
[26] Electricity Market Regulatory Authority (EMRA) of Türkiye, https://www.epdk.gov.tr/Detay/Icerik/3-0-167/resmi-istatistikleri. [Accessed : 23.09.2020].
[27] Turkish Load Dispatch Information System (YTBS), https://ytbsbilgi.teias.gov.tr/ytbsbilgi/frm_istatistikler.jsf. [Accessed : 22.09.2020].
[28] G. Montavon, Samek, W., K-R. Müller. "Methods for interpreting and understanding deep neural networks", *Digital Signal Processing*, 73:1–15, 2018
[29] A. Sherstinsky, "Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network, *Physica D: Nonlinear Phenomena*, 404:132306, 2020.
[30] S. Hochreiter., J. Schmidhuber,"Long short-term memory", *Neural Computing*, 9(8):1735–1780, 1997.
[31] X. Wang, Y. Zhao and F. Pourpanah, "Recent advances in deep learning", *Int. Journal of Machine Learning and Cybernetics*, 11:747–750, 2020.
[32] V. Houdt, V., Greg & Mosquera, G. Carlos & Nápoles, "A Review on the Long Short-Term Memory Model", *Artificial Intelligence Review*, 53. 10.1007/s10462-020-09838-1, 2020.

Article Information Form

Author(s) Contributions
All procedures of the article were designed by Mehmet Bulut, data was prepared, results were obtained and the article was written.

Conflict of Interest Notice
The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper

Ethical Approval and Informed Consent
It is declared that during the preparation process of this study, scientific and ethical principles were followed, and all the studies benefitted from are stated in the bibliography.

Availability of data and materials
Not Applicable

Plagiarism Statement
This article has been scanned by iThenticate ™.

SAKARYA UNIVERSITY JOURNAL OF COMPUTER AND INFORMATION SCIENCES
http://saucis.sakarya.edu.tr/
e-ISSN: 2636-8129
Publisher: Sakarya University
Vol. 7, No. 3, 2024, 338-345
DOI: 10.35377/saucis...1539409
RESEARCH ARTICLE

Emotion Recognition on Turkish Mobile Operator Turkcell's Call Center Calls

Yüksel Yurtay[1], Hüseyin Demirci[1*], Hüseyin Tiryaki[2], Tekin Altun[2]

[1] Sakarya University, Faculty of Computer and Information Sciences, Sakarya, Türkiye
[2] Turkcell Global Bilgi Marketing Consultancy and Call Center Services Inc., İstanbul, Türkiye

Corresponding author:
Hüseyin Demirci, Sakarya University, Faculty of Computer and Information Sciences, Sakarya, Türkiye,
huseyind@sakarya.edu.tr

Article History:
Received: 27.08.2024
Accepted: 30.09.2024
Published Online: 30.10.2024

ABSTRACT
A fundamental component of human intelligence is the capacity for feeling. In addition to being founded on logic and reason, human conduct is also greatly influenced by the emotions that people experience. For the purpose of this study, we classified one thousand real-life call center client voice data in the Turkish language based on the way they expressed their emotions using text emotion detection. We made use of Ekman's emotional labeling and techniques from the field of artificial intelligence, such as deep learning and other similar methods.

Keywords: Emotion recognition, Call center, Deep learning, Artificial intelligence

1. Introduction

Emotion is a complex feeling involving consciousness, physical sensation, and behavior that represents the individual meaning of an object, an occasion, or a situation [1]. An essential aspect of human intelligence is emotion. Human behavior is significantly impacted by emotion in addition to being based on logic and reason. Psychological research has identified three primary methods for modeling emotions [2,3].

The categorical method is grounded in the concept that there is a limited set of fundamental emotions that are universally acknowledged [2]. The prevailing paradigm utilized in research on emotion recognition is the one developed by Ekman [3], which encompasses six fundamental emotions: disgust, surprise, fear, anger, sadness and happiness.

The dimensional methodology claims that emotional states are interconnected and exhibit a systematic relationship with one another[2]. This method encompasses the diversity of emotions in three aspects [4,5]:

- Valence: This dimension pertains to the degree of positivity or negativity associated with an emotion.
- Arousal: This dimension pertains to the level of excitement or indifference associated with an emotion.
- Power: This dimension pertains to the extent of authority or control [2].

The appraisal-based technique can be viewed as an expansion of the dimensional technique. The system employs componential notions of emotion derived from appraisal concept [6]. According to this theory, an individual can feel an emotion when it is elicited by an assessment of circumstances, taking into account the individual's past experiences, objectives, and possibilities for action. Emotions are observed by examining alterations in various important elements, such as cognition, physiology, motivation, motor reactions, sensations, and expressions [2].

Human-computer interaction (HCI) has increased dramatically since human society entered the information age. Researchers from a wide range of subjects, including psychology, neurophysiology, cognitive science, computer science, and others, have been contributing to the study of emotional computing over the course of the past two decades. This technology has already been used in a number of different domains. The applications can be broadly classified into five categories: (1) Formal instruction. Robots can enhance the quality of teaching and learning by monitoring students' emotional states and focus in class. (2) Medical care. The medical robot incorporates cognitive and affective computing to aid doctors in treating psychiatric disorders and offers emotional solace to patients. (3) Sector of the economy that provides intangible goods or services to consumers. Robots employed in many sectors, such as banking, hospitals, catering, government services, and other businesses, can enhance consumers' experience by offering efficient and effective services across the full-service process. (4)

Cite as: Y. Yurtay, H. Demirci, H. Tiryaki, T. Altun, "Emotion Recognition on Turkish Mobile Operator Turkcell's Call Center Calls," *Sakarya University Journal of Computer and Information Sciences*, vol.7, no. 3, pp. 338-345, 2024. Doi: 10.35377/saucis…1539409

The entertainment sector. Integrating emotion recognition and interaction technologies into computer games can enhance the realism of virtual environments, alleviate player exhaustion, and enhance the overall enjoyment of the game. (5) Autonomous driving. The integration of emotion analysis and fatigue detection technologies in intelligent driving plays a crucial role in preventing traffic accidents.

Call centers play a crucial role in the realm of marketing and advertising. A call center is a collective of customer service specialists who manage telephone inquiries from prospective or current clients regarding a company's goods or products. Some call centers promote customer satisfaction and assistance, while others focus on sales expansion, lead generation, and customer acquisition. Call centers play a crucial role in creating a favorable client experience. Therefore, it is imperative that they continually deliver exceptional service in order to cultivate and maintain connections. Therefore, call center personnel must have a wide range of knowledge, display patience, and show helpfulness when interacting with customers. Given that call center personnel are typically the initial point of contact with consumers, it is crucial for them to effectively and appropriately respond to client emotions. Currently, it is advisable to apply emotion recognition algorithms to categorize customers based on how they are feeling and assign them to call center employees with varying levels of experience. This approach aims to enhance customer experience, optimize the response of call center employees, and improve overall company satisfaction.

In the present day, people are becoming more and more dependent on computers to carry out their day-to-day activities, which has resulted in an increased demand for the enhancement of human-computer interactions. The absence of fundamental knowledge hinders a computer's ability to identify and produce emotions. Consequently, extensive research has been carried out on the recognition of emotions. Emotion recognition can be broken down into three primary categories: emotion recognition based on facial expressions, emotion recognition based on voice, and emotion recognition based on text.

In this article, we analyze emotion through human speech. Since human speech is the main subject, we will not focus on human facial expressions regarding emotion recognition. Speech signals are the fastest and most basic means of human communication. Speeches are utilized as a rapid technique of establishing a connection between humans and computers. The speech signal contains valuable information that is not immediately apparent. Thus far, various forms of research have been conducted in the domain of recognition of speech. Despite significant progress in this domain, there remains a substantial disparity between the natural connection of computers and humans. The primary factor for this issue is the computer's incapacity to comprehend the user's emotions. Recognition of speech has emerged as a very complex and demanding area of study within speech processing, garnering significant interest from numerous experts in recent years. In addition, speech emotion recognition has the ability to extract significant meanings from speech, which in turn improves the performance of speech recognition systems [7–13].

In this paper, we studied speech-emotion recognition on call center speech data, which are real-life customer data provided by "Turkcell Global Bilgi Pazarlama Danışmanlık ve Çağrı Servisi Hizmetleri Inc." and handled the speech-emotion recognition in terms of text-based recognition. The common element of speech emotion recognition in literature is that most of the previous work is done by analyzing the voice characteristics of speech and speakers. In addition, the data sets used by researchers were generally recorded in a studio environment and by professional voice actors. However, in the real world, people express their emotions while speaking with more natural tones of voice and non-fluent speech. For example, a person who is frightened or confused should not be expected to speak correctly in terms of diction and grammar. This could cause the suggested models from earlier works to fail on real-life speech data. When we examine the literature based on emotion recognition in text, we can clearly see that most of the works are done on corpora and lexicons. These corpora and lexicons are very well written and labeled by professionals. Using these works in real-life speech data can also cause failure. In our work, we converted 1000 real-life speech data in the Turkish language, which are provided by "Turkcell Global Bilgi Pazarlama Danışmanlık ve Çağrı Servisi Hizmetleri Inc." and 5000 sample data in the English language into text conversation data and labeled it with Ekman's emotional labels. The novelty and significance of the work is that the data we used are purely natural human speech, which is collected via call center calls from customers of Turkish mobile operator Turkcell. This makes our work unique in emotion recognition based on voice and text areas. Also, the speech-to-text technique we use challenges emotion recognition based on voice and text areas, making our work valuable.

The rest of the paper is organized as follows. In Section 2, we mentioned text-based emotion recognition works in literature. Later on, in Section 3, the methodology of our study and dataset attributes are given in Section 3. Finally, results and discussions are mentioned in Section 4.

2. Background Work

Recognizing emotions in text, especially implicit emotion recognition, is a challenging job in Natural-Language-Processing (NLP) that necessitates Natural-Language-Understanding (NLU). Textual emotion recognition can be categorized into various levels: document, paragraph, phrase, and word. The challenge arises at the sentence-level, where emotions are conveyed through the semantics and interconnections of words; as the level progresses, the problem becomes more intricate. However, not all concepts are spoken with clarity; some may involve metaphors, sarcasm, or irony.

Various methodologies have been employed to identify emotions in written language. Studies have been conducted on methods that use keywords to accurately identify and classify explicit emotions. In order to recognize implicit emotions in

text, a number of other ways have been specially presented. These include rule based methods [14,15], classical-learning-based methods [16,17], deep-learning methods [18,19], and hybrid methods [20,21]. In this paper, due to the challenging structure of our methodology, we used deep-learning algorithms to classify emotions in text; therefore, in the rest of the paper, we focus on deep-learning approaches.

2.1. Related Works on Deep Learning Approaches in Text Based Emotion Recognition

Deep learning is a subset of machine learning that involves algorithms acquiring knowledge from experience and comprehending the world based on a hierarchical structure of concepts, with each notion becoming progressively simpler. This methodology enables software to acquire intricate concepts by constructing them using foundational ones. The predominant deep learning model utilized in this context is Long-Short-Term-Memory (LSTM). The LSTM is a distinctive variant of Recurrent-Neural-Networks (RNN) that is proficient in managing long term dependencies. LSTM mitigates the issue of the disappearing or inflating gradient problem, which is frequently encountered with RNNs. Figure 1 delineates the primary stages of LSTM for the purpose of identifying emotions in text. Initially, the emotion dataset undergoes text preprocessing. The preprocessing procedures often involve tokenization, the elimination of stop-words, and lemmatization. Subsequently, the embedding-layer is constructed and then sent to single or several LSTM layers. Subsequently, the resulting data is inputted into a Densely-connected-Neural-Network (DNN) with the same number of units as the emotion tags, utilizing a sigmoid activation function to carry out the classification process.

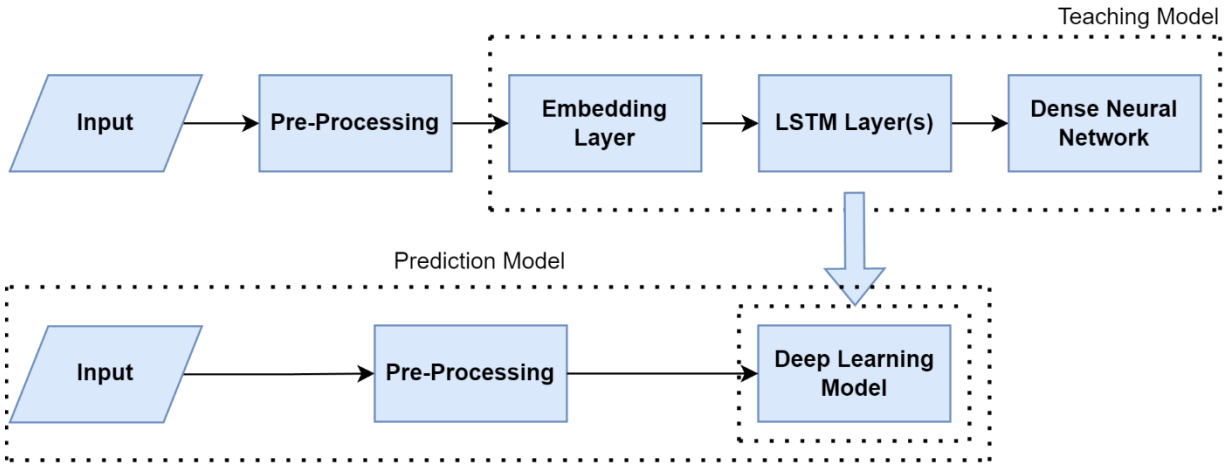


Figure 1. Main Steps of LSTM for Text Emotion Recognition

Shrivastava et al. [22] introduced a deep learning approach that utilizes the word2vec model to recognize emotions in multimedia text. The findings indicate that the accuracy of emotion labels anger and fear surpasses that of other emotion labels, although the recall and F1-score of pleasure outperform other emotion labels.

Rathnayaka et al. [23] introduced a deep-learning framework for detecting several emotions in microblogs. They employed an ekphrasis tool to prepare the data. In the work [24], they defined emotion recognition in the text as a binary classification task, employing two word embedding models: ConceptNet Numberbatch and fastText. The embedding layer was inputted into a Bidirectional-Gated-Recurrent-Unit (Bi-GRU) layer, and the pooling scores were inputted into a Deep Neural Network (DNN).

In the work [25], the authors introduced a deep learning model for identifying emotions in written conversations. They utilized three pre-trained embeddings, namely word2vec-twitter, GloVe, and ekphrasis. An embedding layer was inputted into a Bidirectional Long Short-Term Memory (Bi-LSTM) layer, which was then came next by an attention layer and a Convolutional Neural Network (CNN) layer. The results of the Bi-LSTM and CNN layers were combined, and global max pooling was performed. The scores of pooling were inputted into a Deep Neural Network (DNN) that utilized a softmax activation function to perform classification.

In the work [26], they introduced a deep learning model for identifying emotions in written conversations. They replaced emojis with appropriate emotion terms and inputted the embeddings into a Bi-LSTM layer. The output of the Bi-LSTM was subjected to an inner product operation with the attention weights, and the result was then sent into another Bi-LSTM layer. The scores of pooling were inputted into an LSTM layer and subsequently passed through a DNN with a softmax activation function.

In article [27], researchers introduced a deep-learning algorithm for identifying emotions in written conversations.. The model involves combining three parts of the dialogue into an embedding layer. The output is then passed through three subsequent layers of Bi-LSTM, which are trained using average stochastic gradient descent. Self-attention technique is utilized, then average pooling is performed. The distinction between the two pooled scores is used as input to a two-layer deep neural

network (DNN), which is followed by a softmax function to obtain emotion labels. The language model was trained using the Wikitext-103 dataset.

Xiao [28] presented a deep-learning framework for detecting emotions in written conversations. The model utilized an ekphrasis tool to preprocess the text. The researchers optimized multiple models, such as the Universal-Language-Model (ULM), the BERT model, OpenAI's Generative-PreTraining (GPT) model, DeepMoji model, and a DeepMoji model trained via NTUA embedding. The ULM model exhibited superior performance, whilst the DeepMoji model, trained via NTUA embedding, achieved the second-highest ranking. Nevertheless, combining these models yielded the highest outcome.

In the work [29], they introduced a deep learning framework for detecting emotions in written conversations. The framework has four submodels: a three input submodel (INP3), two output submodels (OUT2), a sentence encoder submodel, and a Bidirectional-Encoder-Representation from Transformers (BERT). The results demonstrated the characteristics acquired by the INP3 and OUT2 componenets yielded superior efficiency compared to the characteristics acquired by the USE and BERT componenets. Nevertheless, the combination of the four componenets yielded the most optimal performance outcome using SVM-n.

3. Material and Methods

3.1 Methodology

This paper outlines two primary methodologies for identifying emotions from voice and textual data. After the data collected from the transcription of audio recordings into text, a Text-Based Classification model is utilized in order to classify the various emotional states. In the second step of the process, a Voice-Based Classification model is created in order to estimate emotional states by directly extracting characteristics from recordings. Furthermore, its objective is to conduct a more extensive analysis by merging the outcomes of these two approaches. This study conducts a comparison of the three strategies and specifically aims to ascertain the most efficacious way.

For the voice model training, English voice data were used, where features such as understanding emotional expressions, stress, and innuendo were similar. This approach aims to enable the model to better understand the tonal characteristics and accents of the language. English audio data provides a basis for learning universal features of the language and integrating this information with text data from different languages. During this phase of the study, we utilized Turkish speech data obtained from Turkcell. However, the quantity of data was insufficient for training and testing the model. To compensate for this, we incorporated sample speech recordings in English to identify universal speech characteristics that can be used to detect emotions in human speech.

In the text model training, completely Turkish text data is used to learn the structural and semantic features of the language. In this way, while the model learns the essential elements of the language, such as grammar rules, word usage, and sentence structure, it also masters the cultural and local features of the language. Turkish text data is critical to ensure that the model better understands the semantic layer of the language and interprets it correctly. The text-based classification technique is specifically designed to perform sentiment classification on textual data obtained from audio sources. This model consists of a series of deep learning layers specifically designed to understand and analyze the features of textual content. Every layer is specifically developed with a unique purpose to optimize the effectiveness and accuracy of the model. These layers form of four components: the Embedding Layer, Bidirectional GRU Layer, Attention Mechanism, and Flatten and Dense Layers.

After both models underwent specialized training in their respective fields, the results obtained using voice and text data were compared and analyzed. Thanks to this integration, the model aims to understand better and interpret both the tonal characteristics of the voice and the structural and semantic elements of the voice. As a result, this bidirectional training approach plays a crucial function in enhancing accuracy. and effectiveness of our language model.

The audio data we used consists of the following attributes;

- Sampling Rate: 8 kHz
- Bit Depth: 8 bits
- Data Labels: Positive, Negative, Neutral
- Distribution:
- Initial data set: Even distribution for Positive, Negative, and Neutral
- 5000 sample records in English and 1000 records in Turkish

We used %70 of the data for training, %15 of the data for validation, and %15 for the data for testing. Each sample record consists of multiple sentences that can express different emotional states in conversations. For instance, a dialogue can commence with favorable sentiments. As the talk unfolds, the client may experience anger or frustration, leading to the characterization of the interaction as negative. In this example, during the initial phases of the conversations, the sentences obtained from the call may be categorized as positive, but in the later stages of the conversation, the extracted sentences can

be classified as negative. As a result, this conversation can be labeled negative for a client. Nevertheless, this interaction might be classified as neutral since the operator fulfilled their duty by providing the client with the most optimal answer. Based on this frame, it is evident that the Turkish recordings dataset consists of around 3000 sentences, equivalent to 1000 recordings.

In this study, different approaches were adopted to process and analyze voice and text-based data. While traditional machine learning algorithms are used on audio data, deep learning models are preferred for text-based data. The Voice-Based classification approach utilizes Mel-frequency cepstral coefficients (MFCC) features extracted from audio recordings to train various machine learning models. The models include Support Vector Machines (SVM), Support Vector Regression (SVR), and Logistic Regression. We used Librosa and pydub libraries for voice analysis and processing. Using these libraries, we processed the audio files at an 8 kHz sample rate. These models are preferred for classification and regression analysis of audio data.

Deep learning models have been developed using LSTM and GRU-based models, dropout and normalization layers, Adam optimizer, and unique learning rate settings to process text-based data. These structures are designed to learn long-term dependencies and relationships in text data. While TensorFlow is used in model training and development, PyTorch support is also provided. Thanks to GPU support, TensorFlow offers a powerful platform, especially for training deep learning models.

Machine learning and feature extraction models for audio data and deep learning techniques for text data were optimized to meet the different needs of this study.

To assess the model, a precise matching criterion was employed to analyze three distinct types of outcomes. False negative (FN) and false positive (FP) refer to inaccurate negative and positive predictions, respectively. True positives (TP) refer to accurate positive predictions that align with the actual correct predictions. The assessment is founded on the performance metrics precision (P), recall (R), and F-score (F1). Recall refers to the proportion of positive cases that are correctly identified as positive, and it is calculated by dividing the number of correctly labeled positive results by the total number of positive cases. Equations for calculating precision (P), recall (R), and F-score (F1) are given in Equation 1, Equation 2, and Equation 3, respectively.

$$\mathrm{P} = \frac{TP}{TP + FP} \tag{1}$$

$$R = \frac{TP}{TP + FN} \tag{2}$$

$$\mathrm{F1} = \frac{2 * P * R}{P + R} \tag{3}$$

4. Results and Discussion

The model developed in this study is an essential step for sentiment analysis in the Turkish language. The results obtained in the initial stages demonstrate the model's capacity to learn basic structures and its potential to classify emotional expressions. Ongoing studies aim to improve performance on the generalized data set and enable the model to be used more effectively in real-world scenarios.

Since we aim to learn the emotional state of the human being via audio data, which can be obtained from a phone call with deep learning methods, we only focused on positive and negative emotional states and ignored neutral states in the results. However, we trained our model to distinguish all the emotional states: positive and negative. The prediction results of the model are given in Table 1.

Table 1. Predicted Result of Both English and Turkish Data

	Actual Positive	**Actual Negative**
Predicted Positive	396	104
Predicted Negative	36	464

Table 2. Statistical Analysis of Both English and Turkish Data

	Positive Emotions	**Negative Emotions**
Recall	0.915662651	0.816513761
Precision	0.791666667	0.927083333
F1	0.849162011	0.868292683

As we can see in Table 2, our model has achieved a recall value of 0.915662651 in positive emotions and 0.816513761 in negative emotions. Also, the precision value of positive emotions is 0.791666667, and the value of negative emotions is 0.927083333. We can calculate the F1 scores of positive and negative emotions with these values: positive emotions=0.849162011 and negative emotions=0.868292683.

The statistical analysis and the results show us our model can distinguish positive and negative emotions from each other with a slight margin of error. The data we used in our model are real-world data gathered from call center conversations, which have never been used in literature and are purely human. The originality of the work is that we used actual human voice data. In other related works, all recordings are made in a studio and are not natural conversations between two humans. Also, we used a hybrid model that decodes speech into text and uses this text data to analyze a semantical emotion in the conversation context.

5. Conclusion

As a result, this document provides a research article that focuses on the identification of emotions in call center conversations involving the Turkish mobile operator Turkcell. The study employs text emotion detection techniques and artificial intelligence methods, specifically deep learning, to categorize real-life call center client voice data according to their expressed emotions. The findings demonstrate that the model exhibits a high level of accuracy in discerning between positive and negative emotions, with a minimal degree of error. This research is a joint effort between Sakarya University and Turkcell, with the goal of enhancing the model and expanding the dataset for future emotion recognition.

This work will provide assistance to both our collaborator Turkcell as well as the marketing and customer service departments of businesses. In our future works, we will exclude the English voice analysis and augment the dataset exclusively with Turkish call center recordings. We will improve the training of the model by incorporating the expanded dataset to capture the structural and phonetic characteristics of the Turkish language. During the final stage, we will conduct simultaneous testing of the trained model on live call center calls.

This work is an early work of a project between Sakarya University and "Turkcell Global Bilgi Pazarlama Danışmanlık ve Çağrı Servisi Hizmetleri" Incorporation.

References

[1] R. C. Solomon, "Emotion Definition, Examples, Scope, Structures, & Facts Britannica." Jan. 2024. Accessed: Jan. 17, 2024. [Online]. Available: https://www.britannica.com/science/emotion

[2] D. Grandjean, D. Sander, and K. R. Scherer, "Conscious emotional experience emerges as a function of multilevel, appraisal-driven response synchronization," *Conscious. Cogn.*, vol. 17, no. 2, pp. 484–495, Jun. 2008, doi: 10.1016/j.concog.2008.03.019.

[3] P. Ekman, "Basic Emotions," in *Handbook of Cognition and Emotion*, John Wiley & Sons, Ltd, 1999, pp. 45–60. doi: 10.1002/0470013494.ch3.

[4] H. Binali and V. Potdar, "Emotion detection state of the art," in *Proceedings of the CUBE International Information Technology Conference*, in CUBE '12. New York, NY, USA: Association for Computing Machinery, Sep. 2012, pp. 501–507. doi: 10.1145/2381716.2381812.

[5] X. Jin and Z. Wang, "An Emotion Space Model for Recognition of Emotions in Spoken Chinese," in *Affective Computing and Intelligent Interaction*, J. Tao, T. Tan, and R. W. Picard, Eds., in Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2005, pp. 397–402. doi: 10.1007/11573548_51.

[6] K. R. Scherer, "Appraisal theory," in *Handbook of cognition and emotion*, Hoboken, NJ, US: John Wiley & Sons Ltd, 1999, pp. 637–663. doi: 10.1002/0470013494.ch30.

[7] J. Nicholson, K. Takahashi, and R. Nakatsu, "Emotion Recognition in Speech Using Neural Networks," *Neural Comput. Appl.*, vol. 9, no. 4, pp. 290–296, Dec. 2000, doi: 10.1007/s005210070006.

[8] A. Mikuckas, I. Mikuckiene, A. Venckauskas, E. Kazanavicius, R. Lukas, and I. Plauska, "Emotion Recognition in Human Computer Interaction Systems," *Elektron. Ir Elektrotechnika*, vol. 20, no. 10, pp. 51–56, Dec. 2014, doi: 10.5755/j01.eee.20.10.8878.

[9] W.-J. Yoon, Y.-H. Cho, and K.-S. Park, "A Study of Speech Emotion Recognition and Its Application to Mobile Services," in *Ubiquitous Intelligence and Computing*, J. Indulska, J. Ma, L. T. Yang, T. Ungerer, and J. Cao, Eds., in Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2007, pp. 758–766. doi: 10.1007/978-3-540-73549-6_74.

[10] D. J. France, R. G. Shiavi, S. Silverman, M. Silverman, and M. Wilkes, "Acoustical properties of speech as indicators of depression and suicidal risk," *IEEE Trans. Biomed. Eng.*, vol. 47, no. 7, pp. 829–837, Jul. 2000, doi: 10.1109/10.846676.
[11] T. H. Falk and W.-Y. Chan, "Modulation Spectral Features for Robust Far-Field Speaker Identification," *IEEE Trans. Audio Speech Lang. Process.*, vol. 18, no. 1, pp. 90–100, Jan. 2010, doi: 10.1109/TASL.2009.2023679.
[12] S. Patil and G. K. Kharate, "A Review on Emotional Speech Recognition: Resources, Features, and Classifiers," in *2020 IEEE 5th International Conference on Computing Communication and Automation (ICCCA)*, Greater Noida, India: IEEE, Oct. 2020, pp. 669–674. doi: 10.1109/ICCCA49541.2020.9250765.
[13] H. Gunes and M. Pantic, "Automatic, Dimensional and Continuous Emotion Recognition," *Int. J. Synth. Emot. IJSE*, vol. 1, no. 1, pp. 68–99, Jan. 2010, doi: 10.4018/jse.2010101605.
[14] S. Y. M. Lee, Y. Chen, and C.-R. Huang, "A Text-driven Rule-based System for Emotion Cause Detection," in *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, D. Inkpen and C. Strapparava, Eds., Los Angeles, CA: Association for Computational Linguistics, Jun. 2010, pp. 45–53. Accessed: Jan. 17, 2024. [Online]. Available: https://aclanthology.org/W10-0206
[15] O. Udochukwu and Y. He, "A Rule-Based Approach to Implicit Emotion Detection in Text," in *Natural Language Processing and Information Systems*, C. Biemann, S. Handschuh, A. Freitas, F. Meziane, and E. Métais, Eds., in Lecture Notes in Computer Science. Cham: Springer International Publishing, 2015, pp. 197–203. doi: 10.1007/978-3-319-19581-0_17.
[16] C. O. Alm, D. Roth, and R. Sproat, "Emotions from Text: Machine Learning for Text-based Emotion Prediction," in *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, R. Mooney, C. Brew, L.-F. Chien, and K. Kirchhoff, Eds., Vancouver, British Columbia, Canada: Association for Computational Linguistics, Oct. 2005, pp. 579–586. Accessed: Jan. 17, 2024. [Online]. Available: https://aclanthology.org/H05-1073
[17] S. Aman and S. Szpakowicz, "Identifying Expressions of Emotion in Text," in *Text, Speech and Dialogue*, V. Matoušek and P. Mautner, Eds., in Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2007, pp. 196–205. doi: 10.1007/978-3-540-74628-7_27.
[18] C. Baziotis *et al.*, "NTUA-SLP at SemEval-2018 Task 1: Predicting Affective Content in Tweets with Deep Attentive RNNs and Transfer Learning," in *Proceedings of the 12th International Workshop on Semantic Evaluation*, M. Apidianaki, S. M. Mohammad, J. May, E. Shutova, S. Bethard, and M. Carpuat, Eds., New Orleans, Louisiana: Association for Computational Linguistics, Jun. 2018, pp. 245–255. doi: 10.18653/v1/S18-1037.
[19] A. Ezen-Can and E. F. Can, "RNN for Affects at SemEval-2018 Task 1: Formulating Affect Identification as a Binary Classification Problem," in *Proceedings of the 12th International Workshop on Semantic Evaluation*, M. Apidianaki, S. M. Mohammad, J. May, E. Shutova, S. Bethard, and M. Carpuat, Eds., New Orleans, Louisiana: Association for Computational Linguistics, Jun. 2018, pp. 162–166. doi: 10.18653/v1/S18-1023.
[20] W. Amelia and N. U. Maulidevi, "Dominant emotion recognition in short story using keyword spotting technique and learning-based method," in *2016 International Conference On Advanced Informatics: Concepts, Theory And Application (ICAICTA)*, Aug. 2016, pp. 1–6. doi: 10.1109/ICAICTA.2016.7803131.
[21] S. Gievska, K. Koroveshovski, and T. Chavdarova, "A Hybrid Approach for Emotion Detection in Support of Affective Interaction," in *2014 IEEE International Conference on Data Mining Workshop*, Dec. 2014, pp. 352–359. doi: 10.1109/ICDMW.2014.130.
[22] K. Shrivastava, S. Kumar, and D. K. Jain, "An effective approach for emotion detection in multimedia text data using sequence based convolutional neural network," *Multimed. Tools Appl.*, vol. 78, no. 20, pp. 29607–29639, Oct. 2019, doi: 10.1007/s11042-019-07813-9.
[23] P. Rathnayaka *et al.*, "Gated recurrent neural network approach for multilabel emotion detection in microblogs," *arXiv.org*. Jul. 2019. Accessed: Jan. 20, 2024. [Online]. Available: https://arxiv.org/abs/1907.07653v1
[24] A. Seyeditabari, N. Tabari, S. Gholizadeh, and W. Zadrozny, "Emotion detection in text: focusing on latent representation," *arXiv.org*. Jul. 2019. Accessed: Jan. 20, 2024. [Online]. Available: https://arxiv.org/abs/1907.09369v1
[25] S. Ge, T. Qi, C. Wu, and Y. Huang, "Thu_ngn at semeval-2019 task 3: dialog emotion classification using attentional lstm-cnn," in *Proceedings of the 13th International Workshop on Semantic Evaluation*, Minneapolis, Minnesota, USA: Association for Computational Linguistics, 2019, pp. 340–344. doi: 10.18653/v1/S19-2059.
[26] L. Ma, L. Zhang, W. Ye, and W. Hu, "PKUSE at SemEval-2019 Task 3: Emotion Detection with Emotion-Oriented Neural Attention Network," in *Proceedings of the 13th International Workshop on Semantic Evaluation*, Minneapolis, Minnesota, USA: Association for Computational Linguistics, 2019, pp. 287–291. doi: 10.18653/v1/S19-2049.
[27] W. Ragheb, J. Azé, S. Bringay, and M. Servajean, "LIRMM-Advanse at SemEval-2019 Task 3: Attentive Conversation Modeling for Emotion Detection and Classification," in *Proceedings of the 13th International Workshop on Semantic Evaluation*, Minneapolis, Minnesota, USA: Association for Computational Linguistics, 2019, pp. 251–255. doi: 10.18653/v1/S19-2042.
[28] J. Xiao, "Figure Eight at SemEval-2019 Task 3: Ensemble of Transfer Learning Methods for Contextual Emotion Detection," in *Proceedings of the 13th International Workshop on Semantic Evaluation*, Minneapolis, Minnesota,

USA: Association for Computational Linguistics, 2019, pp. 220–224. doi: 10.18653/v1/S19-2036.
[29] A. Basile, M. Franco-Salvador, N. Pawar, S. Štajner, M. Chinea Rios, and Y. Benajiba, "SymantoResearch at SemEval-2019 Task 3: Combined Neural Models for Emotion Classification in Human-Chatbot Conversations," in *Proceedings of the 13th International Workshop on Semantic Evaluation*, Minneapolis, Minnesota, USA: Association for Computational Linguistics, 2019, pp. 330–334. doi: 10.18653/v1/S19-2057.

Article Information Form

Authors Contributions: All authors contributed equally to the writing of this paper. All authors read and approved the final manuscript.

Conflict of Interest Notice
The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper

Ethical Approval and Informed Consent
It is declared that during the preparation process of this study, scientific and ethical principles were followed, and all the studies benefitted from are stated in the bibliography.

Plagiarism Statement: This article has been scanned by iThenticate ™.

SAKARYA UNIVERSITY JOURNAL OF COMPUTER AND INFORMATION SCIENCES
http://saucis.sakarya.edu.tr/
e-ISSN: 2636-8129
Publisher: Sakarya University
Vol. 7, No. 3, 2024, 346-360
DOI: 10.35377/saucis...1504388
SAKARYA UNIVERSITY
RESEARCH ARTICLE

Evaluation-Focused Multidimensional Score for Turkish Abstractive Text Summarization

Nihal Zuhal Kayalı[1,2*], Sevinç İlhan Omurca[2]

[1]Turkish-German University, Faculty of Engineering, Department of Computer Engineering, İstanbul, Türkiye
[2]Kocaeli University, Faculty of Engineering, Department of Computer Engineering, Kocaeli, Türkiye

Corresponding author:
Nihal Zuhal Kayalı, Turkish-German University, Faculty of Engineering, Department of Computer Engineering, İstanbul, Türkiye, nihal.kayali@tau.edu.tr

Article History:
Received: 25.06.2024
Accepted: 11.10.2024
Published Online: 30.10.2024

ABSTRACT
Despite the inherent complexity of Abstractive Text Summarization, which is widely acknowledged as one of the most challenging tasks in the field of natural language processing, transformer-based models have emerged as an effective solution capable of delivering highly accurate and coherent summaries. In this study, the effectiveness of transformer-based text summarization models for Turkish language is investigated. For this purpose, we utilize BERTurk, mT5 and mBART as transformer-based encoder-decoder models. Each of the models was trained separately with MLSUM, TR-News, WikiLingua and Fırat_DS datasets. While obtaining experimental results, various optimizations were made in the summary functions of the models. Our study makes an important contribution to the limited Turkish text summarization literature by comparing the performance of different language models on existing Turkish datasets. We first evaluate ROUGE, BERTScore, FastText-based Cosine Similarity and Novelty Rate metrics separately for each model and dataset, then normalize and combine the scores we obtain to obtain a multidimensional score. We validate our innovative approach by comparing the summaries produced with the human evaluation results.

Keywords: Natural language processing, Abstractive summarization, Transformers, Evaluation metrics, ROUGE

1. Introduction

Text summarization is the task of generating a short and illustrative representation of the source text. The primary motivation of this process is to emphasize not only the main ideas but also the significant details of the source text. Another important point is to filter out unnecessary information from a longer text. Due to the rapidly growing text data resources today, automatic text summarization is needed to save time, explore the content quickly, and draw useful information from huge text data sources. However, accurate text summarization is a challenging task because it requires not only fusing the primary information of the text but also understanding long dependencies, reasoning about the contents, and producing fluent and grammatically correct text [1]. From this perspective, several issues must be considered in a precise and efficient text summarization system. These points include the quality of the training data, the chosen neural language model, text feature specifications, the optimal summary length, the assessment of summarization performance, and the implementation of multilingual capabilities.

The summarization tasks have been extensively studied in the literature and broadly categorized into two groups: Extractive and Abstractive Text Summarization (ATS) [2]. In extractive summarization, sentences are selected from the source text based on their relevance scores. These sentences are chosen to preserve specific characteristics, such as keywords, main ideas, and critical concepts from the original text. Extractive text summarization is often used because it creates a quick and concise summary while preserving a significant portion of the text. On the other hand, new sentences not in the original text and expressing more related content are generated in the abstractive summarization. Therefore, ATS is one of the challenging tasks due to the need for a deeper understanding of text and language generation [3]. Although extractive methods are more commonly used in the literature, it has been observed that abstractive methods can produce higher-quality abstracts compared to extractive methods.

ATS involves linguistic difficulties and is done similarly to how people use cause-effect relationships when describing a text. This summarization method aims to generate a summary that looks like it was created by a human and uses semantic structure instead of structural elements. Achieving semantic understanding is difficult because the generated text needs to maintain proper grammar and fluency, which is currently a significant challenge for existing models [4,5]. Furthermore, the challenge is exacerbated by the variability in human language, where the same idea can be expressed in numerous ways, making it

Cite as: N. Z. Kayalı, S. İlhan Omurca, "Evaluation-Focused Multidimensional Score for Turkish Abstractive Text Summarization," *Sakarya University Journal of Computer and Information Sciences*, vol.7, no. 3, pp. 346-360, 2024. Doi: 10.35377/saucis…1504388

difficult for models to generalize effectively [6]. ATS has the potential to produce high-quality summaries that can generate an innovative summary utterly different from the statements contained in the original text and incorporate external knowledge bases [7]. From this point of view, the fields of Natural Language Processing (NLP), Natural Language Understanding (NLU), and generation tasks are critical. Deep learning methods have recently gained a significant interest in these three research areas. In particular, the attention mechanism [8] and the Transformer model [4] gained massive interest in deep neural networks, especially in sequence-to-sequence (Seq2Seq) tasks. Transformer, which is a network architecture constructed by Vaswani et al. [9], depends on feed-forward networks and a multi-head attention mechanism. Transformer-based models capture semantic connections by leveraging the self-attention mechanism, which allows them to consider the relationship between all words in the text simultaneously. Unlike traditional models like RNNs, Transformers do not rely on sequential processing, which makes them more efficient in capturing long-range dependencies across the entire text. Therefore, they are ideal for summarizing longer texts. In addition, Transformer models can more accurately summarize texts by considering language structure features where word order is important. Beyond that, another significant breakthrough in abstractive summarization tasks is pre-trained language models such as BERT [10] or GPT-3 [11], which are built on the Transformer model.

This study investigated the effectiveness of pre-trained Seq2Seq language models represented by BERTurk, mT5, and mBART in summarizing Turkish texts on four different datasets. We interpreted the experimental results separately for each model and dataset by looking at BERTScore, FastText-based Cosine Similarity, Novelty Rate, and the widely used ROUGE score. We normalized the obtained scores and transformed them into a holistic score called "Multidimensional Score" (MDS). MDS was formed with the dimensions of BERTscore for semantic accuracy, ROUGE for superficial word similarity, FastText Cosine Similarity for word-level semantic closeness, and Novelty Score for novelty and originality. Then, we compared the MDS results with human evaluation and interpreted the findings.

Our study is significant as it contributes to the limited research on Turkish abstractive text summarization using Transformer models, incorporating multiple evaluation metrics. Additionally, we provide a comprehensive comparative analysis by evaluating three different models across four datasets with five evaluation criteria. The study emphasizes evaluation metrics, an area with scarce research, all conducted within a consistent experimental setup. The remaining sections of the article are structured as follows: Section 2 gives the Turkish text summarization literature, Section 3 briefly explains the methods used in our study, Section 4 describes the datasets and evaluation criteria used in our experiments, and Section 5 discusses the quantitative and qualitative results. Lastly, conclusions are given in Section 6.

2. Related Works

Automatic text summarization systems have been a widely studied research area in NLP literature since early times. As a result of the in-depth examination of various ATS studies in the literature, some challenges have been highlighted by the researchers: selecting the most informative sentences from the source text, summarization of long single documents, evaluation of the computer-generated summary without human resources and generating a human-like abstractive summary [12]. While initial efforts in ATS were primarily focused on extractive summarization techniques, abstractive summarization has recently become the center of interest within the research community. This shift is due to its ability to address the challenges previously noted. However, there are few studies in the field of ATS in the Turkish language. The predominant approach in Turkish text summarization studies is extraction-based [13].

The morphological structure of the Turkish language is quite demanding and complex. Therefore, tasks such as automatic summarization and heading generation of Turkish texts become challenging. For example, in Turkish, word roots can be modified with various affixes to acquire different meanings. This makes it difficult to understand and summarize Turkish texts. Moreover, more complex summarization methods, such as abstractive summarization, have yet to be researched due to the difficulty of the Turkish language.

Ülker and Özer pointed out that there is not enough dataset for Turkish text summarization [14]. A TTSD-Turkish Text Summarization Dataset was presented for inferential and abstractive summarization tasks in a study. The results obtained from TextRank, Lexrank, LSA-based, Luhn, and Random methods are compared using the ROUGE evaluation metric. The presented dataset gave successful results in every method.

In their 2021 study, Beken Fikri, Oflazer and Yanıkoğlu stated that existing evaluation metrics for Turkish abstractive text summarization systems are insufficient and presented STSb-TR, the first semantic text similarity dataset developed for Turkish [15]. The presented dataset provided a high-quality translation with machine translation and showed that it can be used without the need for expensive human annotations. The study emphasizes that the ROUGE metric is insufficient in agglutinative languages such as Turkish and that semantic similarity models are more effective in this regard. In particular, it was shown with Pearson and Spearman correlation analyses that the proposed models provided higher correlation with human judgments compared to ROUGE. The quantitative and qualitative analysis results of the study revealed that the proposed models captured semantic equivalence more accurately and that these models can be used as evaluation metrics in Turkish abstractive summarization systems.

Baykara and Güngör investigated human evaluations summarization and heading generation tasks on a Turkish dataset using pre-trained Seq2Seq models [16]. They evaluated the performances of mBART, mT5, and BERT models on TR-News and MLSUM. Monolingual BERT models achieved better results than multilingual BERT models for both of the targeted tasks.

In 2022, a study stated that Turkish NLP studies have made significant progress in recent years, and Turkish NLP has become comparable to other languages [17]. Four new Turkish benchmark datasets have been introduced for Turkish NLP tasks, including language modeling, sentence segmentation, and spelling and correction. In addition, MUKAYESE, a comprehensive benchmark suite containing baselines for Turkish NLP tasks, has also been introduced, including language modeling, machine translation, named entity recognition, sentence segmentation, spelling and correction, summarization, and text classification.

Bech et al. [18] examined the current status of summarization studies in the Turkish language and the difficulties encountered in this field. They performed three different experiments: unsorted, token-based sorted, and novelty-based sorted on a dataset obtained by combining TR-News, WikiLingua, and MLSUM datasets at specific rates. ROUGE and score performances were measured using the mT5 model. It was determined that the token-based ordered model gave a better result than other models.

In another study conducted in 2023, an ATS study was conducted using the T5 model on a dataset containing Turkish news and summaries [19]. The researchers collected the dataset and published it for academic use. The summaries generated by the models were evaluated using the ROUGE score and BERTScore performance metrics. As a result of the evaluation, it was observed that more successful results were obtained compared to the studies in the literature.

Baykara and Güngör addressed the limitations of existing evaluation metrics for abstractive text summarization in morphologically rich languages like Turkish by proposing new evaluation metrics [20]. They pointed out that existing metrics, such as ROUGE and METEOR, are insufficient for assessing the performance of summarization systems in agglutinative languages, as these metrics rely on surface-level n-gram matching. This poses significant challenges, particularly for abstractive summarization, where words can be generated in various forms and enriched with affixes. In their study, they proposed using evaluation metrics that consider morphosyntactic properties and conducted correlation analyses with human judgments by training mT5 and BERTurk models on the TR-News dataset. The results demonstrated that using morphosyntactic tokenization during evaluation led to better alignment with human judgments compared to common metrics like ROUGE and METEOR. This study also emphasizes the importance of preprocessing and the morphosyntactic structure of the language in the evaluation process by presenting a new manually annotated dataset for Turkish.

Yüksel and Çebi published a dataset named "TR-News-Sum" which was created for Turkish summarization systems [21]. Attention Based, Pointer Generator, and Reinforcement Learning methods from Seq2Seq Neural Network models were studied on this dataset. ROUGE-1, ROUGE-2, and ROUGE-L were used as evaluation metrics.

3. Methods and Materials

3.1. Encoder-Decoder Architecture

Encoder-decoder networks are an essential and powerful tool in a wide range of Seq2Seq tasks, playing a crucial role in neural abstractive summarization. Encoder-Decoders consist of two main components: an Encoder that takes a word sequence as input and outputs a context vector, and a Decoder that takes the context vector and predicts the subsequent token in the target summary. While the responsibility of the encoder is to encode the entire sequence into a fixed-length vector called a context vector, the responsibility of the decoder is to decode the context vector into a desired summary.

3.2. Transformers

Recurrent models such as LSTM [22] have long been used for encoder-decoder models in various NLP tasks like text summarization. However, more recently, transformers [9] based on self-attention -which is the primary building block of the Transformer- have started to dominate the research field as state-of-the-art networks, especially for Seq2Seq models. One of the reasons behind this situation is that while transformers can parallelize text processing, recurrent models use sequential text processing over time. Another reason is that recurrent models couldn't handle long text sequences. There are some main qualities that make a transformer not suffer from long dependency issues as much as an LSTM network. Through the Attention mechanism [8], the information at the beginning of the sentence becomes equally well represented in the context vector, especially for long sentences. Beyond that, the attention mechanism can capture the words contributing more information from the whole input sequence. For text summarization systems, this means that some related words in the original text are considered more than nonrelated ones when creating the words in the summary. Another important contribution of transformers is the effectiveness of pre-trained language models such as BERT [10], BART [23], and T5 [24] with transformer structures, which has become evident. We use BERT2BERT, mBART, and mT5 models in our experiments. While BART and T5 use both encoder and decoder components, BERT uses encoders only.

3.3. BERTurk

Based on a masked language model, BERT is a contextualized text representation model that undergoes pre-training with a bidirectional transformer encoder architecture [9] BERT2BERT architecture uses a public BERT checkpoint to initialize the

encoder and again chooses the BERT model as the decoder for text generation. The encoder-decoder attention is randomly initialized [25]. When initiated with BERT decoder checkpoints, it autonomously generates summary text, much like Transformers [25], utilizing BERT's predictive capability for masked tokens with bidirectional text representations as its input. In our study, we use a publicly available checkpoint, BERTurk [26], which is a monolingual Turkish BERT model.

3.4. mT5

In their work, Raffel and colleagues [24] introduced the transformer-based T5 (Text-to-Text Transfer Transformer) framework with the intention of treating all text processing problems as 'text-to-text' challenges. This model uses a standard encoder-decoder Transformer as proposed by [9]. As in the BERT-BASE [10] configuration, there are 12 blocks in both the encoder and decoder, with each block composed of two basic components: a self-attention layer and a feed-forward network. A multilingual variation of the T5 model, which is trained on common crawl-based data and covers 101 different languages, is called mT5 [27]. Our studies used the mT5 model as the second language model after the BERTurk model.

3.5. mBART

BART, functioning as a denoising autoencoder in the pretraining of sequence-to-sequence models, excels notably when fine-tuned, especially in the context of text generation tasks. Its architecture is built with a bidirectional encoder and an auto-regressive decoder. Corrupting the original text with a noise function and then reconstructing the original text by a Seq2Seq model learning are two stages of Pretraining in this model [23]. In our study, we used mBART (Multilingual BART), which is a multilingual version of the BART, to fine-tune the datasets. While BART has been pre-trained only for English, mBART, utilizing the same BART architecture, has undergone large-scale monolingual pretraining on multiple languages [28].

3.6. Suggested Method

Our proposed summarization model encompasses the multi-dimensional approach in the performance evaluation process along with the improvement of the text summarization process using the advanced set of parameters. Each of the BERTurk, mT5, and mBART models mentioned in the previous sections is trained individually and separately for processing the dataset. After training the models, the "max_length" and "min_length" parameters are used to determine the maximum and minimum lengths of the summaries to be produced when summarizing the texts in the test dataset. This ensures that the summaries are short enough and strike a balance with the requirements of the content. The "num_beams" parameter specifies the number of beams used in the beam search algorithm, which contributes to a more comprehensive and accurate summarization. "no_repeat_ngram_size" prevents repetitive n-grams (groups of words) from being generated by the model, which increases the diversity and uniqueness of the text. "repetition_penalty" and "length_penalty" control how the model handles repetitions and length. early_stopping=True allows the model to stop summarizing when it reaches a good result. Combining these parameters improves the text summarization process's accuracy and efficiency while improving the quality of the model's output. We propose a comprehensive, multidimensional approach to evaluate these outputs using MDS: Rouge score, BERTScore, Novelty Rate, and FastText-based Cosine Similarity. We compare the MDS results with human evaluation and analyze its usability instead of human evaluation.

4. Implementations and Experiments

4.1. Datasets

In our experiments, four public datasets, MLSUM [29], TR-News [16], WikiLingua [30], and Fırat_DS [19], were used.

The MLSUM (Multi-Language Summarization) dataset is a multilingual large-scale summarization dataset containing more than 1.5 million articles/abstracts from online newspapers in five different languages: Turkish, French, German, Spanish, and Russian. The Turkish news set of MLSUM was taken from a news website. MLSUM dataset has 249277 news items and summary pairs in train, 11565 in validation, and 12775 in test.

The TR-News dataset is a monolingual dataset consisting of Turkish news taken from popular news websites between 2009-2020. The data set includes the news texts' URL, title, summary, content, subject, tags, date, author, and source information. TR-News dataset has 277573 news items and summary pairs in the train part, 14610 in the validation part, and 15379 in the test part.

WikiLingua is a large-scale dataset that can be used for NLP tasks such as summarizing in various languages and extracting semantics from text. The dataset includes summaries and full texts of articles from WikiHow, which is a high-quality data source that provides “how-to” guides covering different topics by various authors. WikiLingua has been prepared in 18 languages to increase language diversity and offer a rich resource for multilingual NLP models.

The dataset we refer to as Fırat DS, which we use in the experiments, is the dataset named "Text Summarization-Keyword Extraction Dataset" made available by the Fırat University Big Data and Artificial Intelligence Laboratory.

We applied the following preprocessing steps to normalize all documents before summarization. The duplicate lines and the lines with blank abstract or text content were deleted. The noise characters, such as unnecessary characters, numbers, and punctuation marks, were cleaned. The tokenization, which is the essential process of dividing text into smaller units, is

realized. All letters in the text are converted to lowercase. In the model training, the datasets are divided into 90% as a training subset, 5% as a validation subset, and 5% as a test subset.

4.2. Evaluation Metrics

Using more than one metric to measure the quantitative results may provide more information about summary quality. That's why we made a comprehensive assessment using different evaluation metrics commonly used in text summarization. In addition to the ROUGE scores, the BERTScore, the Novelty Rate, and FastText-based Cosine Similarity between the original text and summary have also been reported and discussed.

One of the most popular evaluation measures used in summarization systems is the ROUGE (Recall-Oriented Understudy for Gisting Evaluation) performance measure [31]. ROUGE is an n-gram metric that measures the overlapping n-gram units between the reference and model-generated summaries. The F-scores of ROUGE-1 (unigram), ROUGE-2 (bi-gram), and ROUGE-L (the longest common sequence) are reported. ROUGE-L is a Longest Common Subsequence (LCS) based ROUGE metric. LCS automatically identifies the longest co-occurrences in sequence n-grams, naturally taking into account structure similarity at the sentence level. The formula used to calculate the ROUGE-N score is given in equation (1), and the formulas used to calculate the ROUGE-L score are given in equations (2), (3) and (4).

$$\mathbf{ROUGE} - \mathbf{N} = \frac{\sum_{\mathbf{S}\in\{\mathbf{ReferenceSums}\}}\sum_{\mathbf{gram_m}\in\mathbf{S}}\mathbf{Count_{match}(gram_m)}}{\sum_{\mathbf{S}\in\{\mathbf{ReferenceSums}\}}\sum_{\mathbf{gram_m}\in\mathbf{S}}\mathbf{Count(gram_m)}} \tag{1}$$

$$\boldsymbol{P_{LCS}} = \frac{\boldsymbol{LCS(ModelSum, ReferenceSum)}}{\boldsymbol{n}} \tag{2}$$

$$\boldsymbol{R_{LCS}} = \frac{\boldsymbol{LCS(ModelSum, ReferenceSum)}}{\boldsymbol{m}} \tag{3}$$

$$\boldsymbol{F_{LCS}} = \frac{(\mathbf{1}+\boldsymbol{\beta^2})\times \boldsymbol{R_{LCS}} \times \boldsymbol{P_{LCS}}}{\boldsymbol{R_{LCS}} + (\boldsymbol{\beta^2}\times \boldsymbol{P_{LCS}})} \tag{4}$$

"m" is the length of sequences of the model summary, and "n" is the length of sequences of the reference summary. When calculating the F score, the β parameter controls the P importance of recall R and sensitivity. The F score is the harmonic mean of recall and precision. By setting the β value, recall or precision can be prioritized in the evaluation. When β is set to 1, recall and precision are weighted equally, resulting in a balanced F score. If one wants to give more importance to recall, stating that it is more important that most of the n-grams in the references are found in the candidates, one can increase the value of β to a value greater than 1. In this study, β is chosen to be 1.

We also used BERTScore [32] to evaluate our experiments. BERTScore computes a semantic similarity score by interpreting the reference and model-generated summaries. Unlike ROUGE, BERTScore measures text similarity by considering semantic similarity in addition to word-level similarity. Therefore, reporting ROUGE and BERTScore together is essential for a more detailed analysis of a text summarization system. The BERTScore-based precision, recall, and F-score are given in equations (5), (6), and (7), respectively.

$$\boldsymbol{P_{BERTScore}} = \frac{\mathbf{1}}{|\hat{\boldsymbol{x}}|}\sum_{\hat{\boldsymbol{x}}_j\in\hat{\boldsymbol{x}}} \boldsymbol{max}(\boldsymbol{x_i}\in\boldsymbol{x})\, \boldsymbol{x_i^T}\hat{\boldsymbol{x}}_j \tag{5}$$

$$\boldsymbol{R_{BERTScore}} = \frac{\mathbf{1}}{|\boldsymbol{x}|}\sum_{\boldsymbol{x_i}\in\boldsymbol{x}} \boldsymbol{max}(\hat{\boldsymbol{x}}_j\in\hat{\boldsymbol{x}})\, \boldsymbol{x_i^T}\hat{\boldsymbol{x}}_j \tag{6}$$

$$\boldsymbol{F_{BERTScore}} = \mathbf{2}\times\frac{\boldsymbol{P_{BERTScore}}\times\boldsymbol{R_{BERTScore}}}{\boldsymbol{P_{BERTScore}}+\boldsymbol{R_{BERTScore}}} \tag{7}$$

BERTScore converts the words in two texts that are compared into high-dimensional vectors using a BERT model. Each word is transformed into a vector through the model: BERT(x_i) and BERT($\hat{x}_j$), which is the vector representation of each word. These vectors are representations that capture the meaning of each word in a specific context. In the BERTScore formulas, "x" represents the reference summary representations, and "$\hat{x}$" represents the candidate summary representations. Precision and recall metrics are calculated for BERTScore by comparing each token representation "x_i" of the reference summary with each token representation "$\hat{x}_j$" of the candidate summary.

To evaluate text summarization in a broader context, we also used FastText-based Cosine Similarity to compare summaries with each other. FastText is a library developed by Facebook AI Research (FAIR), and its primary purpose is to produce word and sentence representations quickly and effectively for NLP tasks [33]. This library provides word embeddings for each word in the summary, representing them as high-dimensional vectors based on their semantic meaning to capture word relationships and similarities. Each word in the reference and generated summaries is mapped to its corresponding vector using FastText. The word vectors are averaged to form a single vector representing the entire summary. The cosine similarity between the two summary vectors is calculated.

In abstractive summarization, evaluating the abstractness level (text novelty) of reference summaries in data sets and the summaries created is important. The calculation of the Novelty Rate typically begins after the summarization process is complete by examining each sentence or phrase within the summary. This involves checking whether the original text's sentences, phrases, or n-grams already exist. If they are not exactly found in the original text, they are considered as new content. The Novelty Rate is calculated by dividing the number of new n-grams, sentences, and phrases in the entire summary by the total number of n-grams, sentences, and phrases. This metric, which is used to calculate Novelty Rates in detail for each data set to evaluate whether the abstract is creative and original, is presented in Formula 8.

$$\boldsymbol{Novelty\ Rate} = \frac{\boldsymbol{Count\ of\ Novel\ n-grams\ in\ Reference\ Summaries}}{\boldsymbol{Count\ of\ n-grams\ in\ Reference\ Summaries}} * \mathbf{100} \quad (8)$$

Unfortunately, the text can be assessed using various metrics, including ROUGE and BERTScore, which may describe an overlap between the reference and the generated summaries. Nevertheless, human evaluation should always be part of the process. Human evaluation adds an additional touch in determining the coherence, style, and context of the summaries, which most automated systems cannot provide. Humans can judge the subtleties of language, including irony, humor, and emotional tone, which computerized systems might overlook. Moreover, human evaluators can assess summaries' factual accuracy and overall quality, ensuring they are not only statistically like reference summaries but also meaningful and informative to readers. Given these considerations, our study has also incorporated this metric, reinforcing our findings with a comprehensive view that blends algorithmic precision with human insight. In addition, in order to observe whether the MDS results can replace human evaluation, the results of both were compared with each other.

5. Results

Our study was conducted on a server with 16 Core AMD Ryzen Thread ripper 1950X 16-Core Processor CPU, 32 GB RAM, and Quadro GV100 GPU graphics card with Ubuntu Server operating system. The models were trained with the datasets for three epochs, and each model took an average of 70 hours to train using the datasets. The Adam optimizer was utilized with a learning rate of 1e-3. We used cross-entropy loss for training, which is calculated by comparing the generated output with the reference summary. To prevent overfitting, a dropout rate of 0,1 was applied in the embedding and attention layers of the encoder. Additionally, Layer Normalization was extensively used throughout the encoder to stabilize and enhance the training process. Furthermore, a specific strategy was employed to improve summarization performance: the length of the input text was analyzed, and 10% of this value was set as the "min_length" parameter for the model's summary generation. This approach aimed to ensure that the generated summaries were both concise and adequately detailed, adapting to the input text's length to enhance overall summarization quality.

In our experiments, a method has been employed to enhance the summarization performance of the models. The length of the input text to be summarized has been determined, and 10% of this value has been set as the “min_length” parameter for the model's summary generation function. With this approach, it is aimed to generate more concise and comprehensive summaries by adjusting the length of the generated summary according to the length of the input text.

In evaluating summarization models, it is important to determine the level of abstraction between the reference summary and the model-generated summary. Because of that, our study discussed the novelty dimension in the text summarization task. The Novelty Rates of summaries are reflected as percentages in Table 1, and these results are calculated according to the rates between the whole content of the news text, the reference summary, and the generated summary.

Table 1. The Obtained Novelty Rates for Datasets

		BERTurk			**mT5**			**mBART**		
		1-gram	**2-gram**	**3-gram**	**1-gram**	**2-gram**	**3-gram**	**1-gram**	**2-gram**	**3-gram**
TR-News	**Content / Reference Summary**	0,410	0,655	0,763	0,410	0,655	0,763	0,410	0,655	0,763
	Content / Generated Summary	0,118	0,267	0,393	0,088	0,185	0,279	0,121	0,240	0,358

MLSUM	**Content / Reference Summary**	0,386	0,617	0,718	0,386	0,617	0,718	0,386	0,617	0,718
	Content / Generated Summary	0,123	0,285	0,432	0,095	0,211	0,315	0,121	0,245	0,360
WikiLingua	**Content / Reference Summary**	0,510	0,872	0,965	0,510	0,872	0,965	0,510	0,872	0,965
	Content / Generated Summary	0,071	0,144	0,202	0,060	0,117	0,166	0,089	0,155	0,231
Firat_DS	**Content / Reference Summary**	0,390	0,611	0,714	0,390	0,611	0,714	0,390	0,611	0,714
	Content / Generated Summary	0,120	0,275	0,410	0,205	0,300	0,399	0,299	0,387	0,503

In a summarization task, the newer words and phrases the generated summary contains compared to the original text, the higher the Novelty Rate. In other words, a high Novelty Rate score means that the generated summary contains different words and sentence structures from the original text content and its reference summary. The results in Table 1 reflect the Novelty Rates of each transformer model on each dataset. In Table 1, the lines expressed as "content-reference summary" for each dataset reflect the Novelty Rate between the original text content and the reference summary. Similarly, the lines expressed as "content-generated summary" for each data set reflect the Novelty Rate between the original text content and the model-generated summary. If we examine these results obtained on a 1-gram scale, it is seen that the Novelty Rates obtained with BERTurk, mT5, and mBART models are close to the Novelty Rates in the reference summaries. It is observed that the mT5 model mainly provides a higher Novelty Rate compared to the mBART and BERTurk models.

In our experiments, we use FastText-based Cosine Similarity scores to evaluate the term similarities between the original text and summaries. Table 2 illustrates the Cosine Similarity scores.

Table 2. FastText-based Cosine Similarity scores

Datasets	**Models**	**Content - Reference Summary**	**Content-Generated Summary**	**Reference Summary – Generated Summary**
TR-News	**BERTurk**	0.838	0.886	0.706
	mT5	0.838	0.791	0.748
	mBART	0.838	0.792	0.750
MLSUM	**BERTurk**	0.833	0.770	0.733
	mT5	0.833	0.809	0.734
	mBART	0.833	0.825	0.718
WikiLingua	**BERTurk**	0.888	0.878	0.875
	mT5	0.888	0.905	0.841
	mBART	0.888	0.908	0.871
Firat_DS	**BERTurk**	0.907	0.916	0.889
	mT5	0.907	0.934	0.860
	mBART	0.907	0.949	0.908

The score between Content and Generated Summary indicates how similar the summary generated by the model is to the original text content. A high-performance score implies that the model has skillfully captured the core concepts present in the original text content. When Table 2 is examined, it is seen that these scores are higher than both the "Content-Reference summary" similarity scores and "Reference Summary-Generated Summary" similarity scores. When the obtained results are analyzed, it is concluded that BERTurk and mBART performed the best on both TRNEWS and MLSUM datasets, with the mBART model achieving the highest scores on the MLSUM dataset. The mT5 model, on the other hand, achieved lower scores than BERTurk and mBART.

Table 3 and Table 4, respectively, present the ROGUE and BERT scores from all models for a more comprehensive comparison.

Table 3. ROUGE Scores

		BERTurk			**mT5**			**mBART**		
		R1	R2	RL	R1	R2	RL	R1	R2	RL
TR-News	P	0.388	0.291	0.338	0.390	0.269	0.344	0.280	0,176	0,223
	R	0.725	0.569	0.652	0.527	0.369	0.472	0,494	0,309	0,399
	F1	**0.490**	**0.372**	**0.431**	0.438	0.303	0.391	0,343	0,217	0,275
MLSUM	P	0.362	0.223	0.288	0.410	0.246	0.359	0.371	0.230	0.300
	R	0.503	0.301	0.393	0.399	0.230	0.345	0.419	0.253	0.340
	F1	**0.410**	**0.252**	0.325	0.392	0.228	**0.344**	0.378	0.232	0.307
WikiLingua	P	0.433	0.146	0.272	0.421	0.145	0.253	0.356	0.114	0.203
	R	0.210	0.070	0.134	0.236	0.081	0.141	0.279	0.087	0.161
	F1	0.271	0.091	0.171	**0.293**	**0.101**	**0.177**	0.286	0.089	0.163
Firat_DS	P	0.327	0.222	0.278	0.354	0.208	0.278	0.311	0.212	0.260
	R	0.594	0.434	0.521	0.527	0.320	0.412	0.561	0.388	0.475
	F1	0.407	**0.282**	**0.350**	**0.412**	0.247	0.324	0.387	0.265	0.325

In Table 3, precision (P), recall (R), and F1-scores for ROGUE-1 (R1), ROUGE-2 (R2), and ROUGE-L (RL) are reflected, and the highest F1-scores are highlighted. Considering the ROGUE scores, it is realized that the BERTurk model gives more successful results for datasets other than WikiLingua. To provide an example of other striking results, the precision values of the mT5 model had the higher results.

Table 4. BERT Scores

		BERTScore		
		Precision	Recall	F1
TR-News	**BERTurk**	0.648	0.805	**0.716**
	mT5	0.648	0.719	0.680
	mBART	0.544	0.650	0.589
MLSUM	**BERTurk**	0.611	0.686	0.646
	mT5	0.663	0.661	**0.661**
	mBART	0.611	0.650	0.627
WikiLingua	**BERTurk**	0.598	0.589	**0.592**
	mT5	0.599	0.566	0.580
	mBART	0.565	0.598	0.579

Firat_DS	**BERTurk**	0.581	0.733	**0.645**
	mT5	0.589	0.700	0.638
	mBART	0.589	0.720	**0.645**

Table 4 concludes that BERTurk, mT5, and mBART transformers have similar summarization abilities. However, BERTurk has the highest BERTScore value in all datasets except MLSUM.

Proper evaluation of text summarization systems requires incorporating qualitative and quantitative analysis. For the qualitative analysis, in Table 5 and Table 6, we present some sample summaries generated by four models in terms of semantic similarity to the original text contents. While these results are reflected, new words synthesized by the models or derived by the suffixes are highlighted in bold.

Table 5. Sample Summaries From TR-News and MLSUM Datasets.

TR-News	**MLSUM**
Original Text Content	**Original Text Content**
"Fransa'nın girişimiyle düzenlenen konferansın amacı, kimyasal silah kullanımını engellemek. 30 ülkenin katılımıyla düzenlenen konferansta Türkiye'yi Dışişleri Bakanı Mevlüt Çavuşoğlu temsil ediyor. Dışişleri Bakanı Mevlüt Çavuşoğlu'nun toplantı kapsamında ikili görüşmelerde bulunması da bekleniyor. Konferans sonunda katılımcı ülkeler, kimyasal silah kullanan ya da geliştirenlerin cezasız kalmamalarına yönelik bir dizi taahhütte bulunacak. Ayrıca ülkeler, kimyasal saldırıların failleri hakkında elde ettikleri bilgileri uluslararası kuruluşlarla paylaşma konusunda da mutabakata varacak."	"KAYSERİ'nin Develi ilçesine bağlı Yukarı Künye Mahallesinde bulunan Homurlu Şelalesi'ne ailesiyle birlikte gelen 18 yaşındaki Serkan Turan Timurbuğa, iddiaya göre fotoğraf çektirmek için şelalenin üst tarafındaki kayalıklara çıktı. Fotoğraf çekinirken dengesini kaybeden Timurbuğa, dereye düşerek gözden kayboldu. Yakınlarının haber vermesi üzerine olay yerine AFAD, jandarma, 112 ve gönüllü Türkuaz Arama Kurtarma ekipleri sevk edildi. AFAD ekipleri, jandarma ile Turkuaz ekipleri tarafından sürdürülen arama kurtarma çalışmaları devam ediyor."
Reference Summary	**Reference Summary**
"Dışişleri Bakanı Mevlüt Çavuşoğlu, "Kimyasal silah kullanımının cezasız kalmaması için uluslararası ortaklık" konulu dışişleri bakanları toplantısına katılacak."	"KAYSERİ'de bulunan Homurlu Şelalesi'nde kayalıklarda fotoğraf çekinirken dengesini kaybederek dereye düşen genci arama çalışmaları sürüyor."
Summary generated with BERTurk	**Summary generated with BERTurk**
"Fransa'nın girişimiyle düzenlenen **konferansta** Türkiye'yi temsil **eden** 30 ülkenin katılımıyla konferansta kimyasal silah **kullanımı** konusunda mutabakata **varacağı bildirildi**."	"KAYSERİ'de **bir kişi** fotoğrafı çektirmek için **şelaleye** çıktı. Ancak dengesini **kaybetti** ve gözden kayboldu."
Summary generated with mT5	**Summary generated with mT5**
"Fransa'nın girişimiyle düzenlenen konferansta Türkiye'yi Dışişleri Bakanı Mevlüt Çavuşoğlu temsil ediyor."	"Fotoğraf çektirmek için şelalenin üst tarafındaki kayalıklara **çıkan** 18 yaşındaki **genç** gözden kayboldu."
Summary generated with mBART	**Summary generated with mBART**
"Fransa'nın girişimiyle gerçekleştirilen konferansta Türkiye'yi Dışişleri Bakanı Mevlüt Çavuşoğlu temsil ediyor."	"Kayseri'nin Develi ilçesine bağlı Yukarı Künye Mahallesinde bulunan Homurlu Şelalesi'ne ailesiyle birlikte çıkan genç adam gözden kayboldu."

Table 6. Sample Summaries From WikiLingua and Firat_DS Datasets

WikiLingua	**Firat_DS**
Original Text Content	**Original Text Content**
"Kötü şans getiren şeylere ilişkin çoğu batıl inancı herkes bilir ama belli başlılarını tekrarlamak iyi bir fikir. Bu şekilde kötü şanstan kaçmak için stratejik olarak davranışlarını değiştirmeye çalışabilirsin. Kötü şanstan kaçış olmasa da işaretleri tanıyabilecek, tersine çevirmek için hemen harekete geçebileceksin. Kötü şansın bazı yaygın belirtileri şunlardır: Ayna kırmak – bunun yedi yıl kötü şans getirdiği söylenir. Karga görmek – karşına karga çıkmasının kötü şans getirdiği söylenir. Ama karşına iki karga çıkarsa kötü şans tersine döner. Merdiven altından geçmek – bunun kötü şans getirdiğine inanılır çünkü duvara dayanan merdiven üçgen oluşturur – bu, Kutsal Üçlü'nün yani Baba, Oğul ve Kutsal Ruh'un simgesidir. Üçgenin içinden geçmekle kutsal zemini bozmuş olursun. Kendine "uğursuzluk getirmek" – bu, sana olacağını düşündüğün kötü bir şeyi yüksek sesle söylemek demektir. Bir nevi kadere meydan okumaktır. Bunu tersine çevirmek için masaya ya da herhangi bir zemine 3 kez vur ama vurma sesinin duyulduğundan emin ol. Opal taşı takmanın kötü şans getirdiğine inanılır – tabii eğer Ekim'de doğmadıysan. Kaldırımdaki çatlaklara basmak. Eski bir deyişin söylediği gibi: "Bir çatlağa basarsan kader, annenin belini kırar!" Karşına kara kedi çıkmasının kötü şans getirdiği söylenir – bu batıl inanç, kedilerin cadılarla ve büyüyle olan bağlantısından gelir. İçeride şemsiye açmanın kötü şans getirdiği düşünülür – bu, gölge için şemsiye kullanan Eski Mısırlılardan gelen bir batıl inançtır. O zamanlarda içeride şemsiye açmanın Güneş Tanrısı'na hakaret olduğuna inanılırdı. Bazı batıl inançlar daha az bilinir. Böyle batıl inançlar hakkında okumak ve belli eylemlerin yaratacağı risklerin farkında olmak iyi fikir. Yoksa geri dönüşü olmayan bir şekilde kendine kötü şans getirebilirsin…"	"Yangın akşam saatlerinde Efeler Mahallesi 2296 sokaktaki Sağlık Evleri sitesi G Blok 2. Katta meydana geldi. Edinilen bilgiye göre; Seyhan T.'ye ait evde kiracı olarak oturan aile mutfakta olduğu sırada salon bir anda yanmaya başladı. Kısa sürede yangının büyümesi üzerine dairede oturanlar evden çıkarak canlarını kurtardı. Elektrik kontağından çıktığı düşünülen yangına itfaiye ekipleri hemen müdahale etti. Binanın doğalgazlı olması nedeniyle bölgeye ilgili ekipler de çağrıldı. İtfaiye ekipleri bir yandan yangını söndürmeye çalışırken, diğer yandan da binanın doğalgaz ve elektrikleri kesildi. Aydın Büyükşehir Belediyesi İtfaiye Dairesi Başkanı Serdar Adanır'ın da söndürme çalışmalarına katıldığı yangında, polis ekipleri de güvenlik önlemi aldı, sağlık ekipleri de her ihtimale karşı hazır bulundu. Söndürme çalışmaları tamamlandıktan sonra tahliye edilen binanın sakinleri elektrik ve doğalgazın açılması ile evlerine yerleşti. Polis olayla ilgili soruşturma başlattı."
Reference Summary	**Reference Summary**
"Kötü şans getiren eylem ya da durumlardan kaçın. Daha az bilinen batıl inançlar hakkında bilgi edin. Ölüm ya da hastalık habercisi olan batıl inanışlara karşı dikkatli ol., Tuz kullan. Kırık ayna parçalarını atma. Tütsü yak. Koruyucu tılsım taşı. Adaçayı yak. Kristal ya da taş kullan. Sevap işle. Çakralarını taze çiçeklerle arındır. Dua et. Evini topla. Işığın gücünü kullan. Başka bir ülkeye seyahat et., Şansın dönmeye başladığında bunu fark et."	"Aydın Efeler ilçesinde akşam saatlerinde çıkan yangın paniğe neden oldu. Kısa sürede tüm daireyi saran yangın binanın diğer katlarına sıçramadan söndürüldü. Yangın çıkan binada oturanlar ekipler tarafından tahliye edilirken, yangında yaralanma ve can kaybı yaşanmaması teselli kaynağı oldu."
Summary generated with BERTurk	**Summary generated with BERTurk**
"Kötü şans getiren şeylere ilişkin kontrol et. İyi şansın kötü şans getirdiğine inanılmaya çalış., Bir batıl inanç hakkında okumak ya da yazmaktan kaçın. Olumsuzluklara karşı uyanık ol. Başkalarına yardım etmekten hoşlandığını söyle. Hoşlandığın birini seç."	"Aydın'ın Efeler **ilçesinde** bir apartmanın mutfağında çıkan yangın itfaiye ekipleri tarafından söndürüldü. Alevlerin büyümesi üzerine dairede oturanlar canlarını kurtarıp, evlerinden çıktı. Yangında ölen ya da yaralanan olmazken, olay yerine çok sayıda itfaiye ekibi sevk edildi."

Summary generated with mT5	**Summary generated with mT5**
"Kötü şanstan kaçmak için işaretleri tanı. Kendine "uğursuzluk getirme" sesini söyle. Karga görmek için masaya ya da herhangi bir zemine 3 kez vur., İçeride şemsiye aç. Bir kedi var mı diye düşün. İyi bir inançla bağlantıyı ortaya koy."	"**Aydın'ın** Efeler **ilçesindeki** Sağlık Evleri sitesi G Blok 2. Katta yangın meydana geldi. İtfaiye ekipleri yangını söndürmeye çalışırken, diğer yandan da binanın doğalgazlı olması nedeniyle bölgeye ilgili ekipler çağrıldı."
Summary generated with mBART	**Summary generated with mBART**
"Kötü şans getiren şeyleri tekrarlamaktan kaçın. Kendine meydan oku. Başka kötü niyetli davranışları görmezsen batıl inançlara karşı dikkatli ol., Güneş Tanrısı'na hakaret etme. Bir şemsiye açmanın kötü şans getirdiğini unutma. Kıskançlık ve öfkelilik arasındaki çizgiyi öğren. Dikkatin dağılırsa bir kart oyunu oynamayı dene. Herşeyden haberdar olup olmadığını bil. Hayatına devam et. Başarısız olduğun şeyin altında yatan nedenlerin neler olduğuna karar ver. Arada bir kendini suçlu hissettir. Kişisel hataların üstesinden gel."	"Bursa'nın Efeler **Mahallesinde** evde kiracı olarak oturan aile mutfakta çıkan yangına itfaiye ekipleri müdahale etti. Yangını **söndürme çalışmaları devam ederken**, diğer yandan da binanın doğalgaz ve bir anda yanmaya başladı. Kısa sürede yangının büyümesi üzerine dairede oturanlar evden çıkarak canlarını kurtardı."

When the results in Table 6 are examined for the TR-News dataset, it is observed that the summary generated by the BERTurk is more informative than the other models in terms of the news content and the subject of the meeting mentioned in the original summary. However, since it does not include the name of the person who will attend the meeting, the sentence is semantically incomplete. It has been observed that the summaries produced by mT5 and mBART are inferential; that is, they can combine the sentences in the original text without making many changes.

The sample news content of the MLSUM dataset was summarized by all models as meaningful and compatible sentence structures. The word "one person" is in the summary of the BERTurk model; the word "young" in the mT5 summary is generated by the model. In the summary produced by the mBART model, the "young man" is entirely accurate information produced by the model. The model produced the word "man" by inferring that the person in the news text was male from the name. Similarly, it produced the word "young" from the knowledge of his age.

As the WikiLingua dataset is translated from another language, the sentence structures are influenced by the language structure of the original text. This situation caused the translation effects to be reflected in the flow of the text by different grammar rules or word order between some languages. When the texts in the WikiLingua database were analyzed, it was observed that most of the sentences ended with words with imperative or infinitive endings. While the texts in other news content datasets use more objective and informative language regarding grammar and sentence structure, WikiLingua contains more colorful and narrative elements. These differences directly affected the summarization performance of the models. As a result, although all three models could capture some concepts from the original text accurately, each contained significant misunderstandings, inconsistencies, and information outside the original text. The quality of the summaries is relatively lower due to word errors and misunderstandings.

The BERTurk model for the Firat_DS dataset produced the closest result to the original summary but omitted some crucial details. The mT5 and mBART models produced summaries with some incorrect information. It should be remembered that summarization models often fail to convey the details in the text entirely and accurately and even create false information. The BERTurk summary has retained the essential elements of the original summary. The information that the fire broke out in an apartment, the firefighters intervened, and the residents were evacuated was accurately summarized. The mT5 summary accurately stated the location of the building where the fire occurred. In addition, there is information that firefighters are trying to extinguish the fire and that the relevant teams are called because it is natural gas. However, the mT5 summary does not explain why the fire broke out or the residents' condition. The mBART summary accurately summarized where the fire started, where firefighters intervened, and where extinguishing efforts are ongoing. However, this summary contains incorrect information. For example, the phrase "Efeler Mahallesi of Aydın" in the original text has been changed to "Efeler Mahallesi of Bursa" in the mBART summary. In addition, the phrase "natural gas and suddenly it started to burn" was added to the mBART summary, providing a detail that was not included in the original text.

After all evaluation metrics were calculated and analyzed separately, the scores required to calculate the MDS (Multi-Dimensional Score) were collected. The pre-normalization scores prepared using the F1 values calculated according to the 1 and 2-gram values of all metrics used are presented in Table 7. Normalization is necessary to ensure that metrics of different scales contribute fairly, and to maintain consistency in the results when combined in MDS calculations. The formula used in the normalization calculation of the values in our study is presented in equation (9). " x " represents the original metric value, "x_{min}" represents the lowest value observed for the relevant metric, "x_{max}" represents the highest value observed for the relevant metric, and "x_n" represents the normalized value between 0 and 1. The final versions of the values presented in Table

7, normalized between 0 and 1, are presented in Table 8. Additionally, in Table 7 and Table 8, "R" stands for Rouge score, "B" stands for BERTScore, "F" stands for FastText-based Cosine Similarity, and "N" stands for Novelty Rate.

$$x_n = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{9}$$

Table 7. Scores of Evaluation Metrics Before Normalization

Datasets	Models	1 GRAM				2 GRAM			
		R	B	F	N	R	B	F	N
TR-News	BERTurk	0.490	0.716	0.706	0,118	0.372	0.716	0.706	0,267
	mT5	0.438	0.680	0.748	0,088	0.303	0.680	0.748	0,185
	mBART	0,343	0.589	0.750	0,121	0,217	0.589	0.750	0,240
MLSUM	BERTurk	0.410	0.646	0.733	0,123	0.252	0.646	0.733	0,285
	mT5	0.392	0.661	0.734	0,095	0.228	0.661	0.734	0,211
	mBART	0.378	0.627	0.718	0,121	0.232	0.627	0.718	0,245
WikiLingua	BERTurk	0.271	0.592	0.875	0,071	0.091	0.592	0.875	0,144
	mT5	0.293	0.580	0.841	0,060	0.101	0.580	0.841	0,117
	mBART	0.286	0.579	0.871	0,089	0.089	0.579	0.871	0,155
Firat_DS	BERTurk	0.407	0.645	0.889	0,120	0.282	0.645	0.889	0,275
	mT5	0.412	0.638	0.860	0,205	0.247	0.638	0.860	0,300
	mBART	0.387	0.645	0.908	0,299	0.265	0.645	0.908	0,387

Table 8. Scores of Evaluation Metrics After Normalization

Datasets	Models	MDS	1 GRAM					2 GRAM				
			R	B	F	N	MDS1	R	B	F	N	MDS2
TR-News	BERTurk	0,600	1,000	1,000	0,000	0,243	0,561	1,000	1,000	0,000	0,556	0,639
	mT5	0,472	0,763	0,737	0,208	0,117	0,456	0,756	0,737	0,208	0,252	0,488
	mBART	0,259	0,329	0,073	0,218	0,255	0,219	0,452	0,073	0,218	0,456	0,300
MLSUM	BERTurk	0,418	0,635	0,489	0,134	0,264	0,380	0,576	0,489	0,134	0,622	0,455
	mT5	0,377	0,553	0,599	0,139	0,146	0,359	0,491	0,599	0,139	0,348	0,394
	mBART	0,318	0,489	0,350	0,059	0,255	0,288	0,505	0,350	0,059	0,474	0,347
WikiLingua	BERTurk	0,252	0,000	0,095	0,837	0,046	0,244	0,007	0,095	0,837	0,100	0,260
	mT5	0,187	0,100	0,007	0,668	0,000	0,194	0,042	0,007	0,668	0,000	0,180
	mBART	0,246	0,068	0,000	0,817	0,121	0,252	0,000	0,000	0,817	0,141	0,239
Firat_DS	BERTurk	0,614	0,621	0,482	0,906	0,251	0,565	0,682	0,482	0,906	0,585	0,664
	mT5	0,609	0,644	0,431	0,762	0,607	0,611	0,558	0,431	0,762	0,678	0,607
	mBART	0,764	0,530	0,482	1,000	1,000	0,753	0,622	0,482	1,000	1,000	0,776

As presented in Table 8, MDS scores for 1-gram and 2-gram were first calculated, and then these values were averaged. When the MDS results are examined, there are significant parallels between the human evaluation results. While these parallels and similarities indicate a positive outcome for our study, it should be noted that further validation, including human annotations and correlation measurements, is required to conclusively determine the reliability of MDS as an evaluation metric for text summarization. This alignment between MDS scores and human evaluation suggests that MDS may serve as an effective complementary metric for assessing model performance. For example, for the TR-News Dataset, the BERTurk model showed the highest performance according to both MDS and human evaluation results. MDS scores reflect that BERTurk is strong in superficial similarity and semantic accuracy, confirming the superiority of the model. Both mT5 and mBART models showed poor performance in terms of MDS and human evaluation. MDS evaluation reflected the performance difference between the models well. In terms of MLSUM Dataset, mBART was the best in both MDS and human evaluation. MDS scores provided a fair estimate of the model's ability to gather information and form ideas, as well as the overall human evaluation. The existing BERTurk and mT5 models again ranked in the average range in terms of MDS and human evaluation results. In summary, the results showed that it is reasonable to claim that MDS scores adequately represent the summary performance. In the WikiLingua Dataset, the mBART model had the highest MDS scores and also performed relatively better in human evaluation results. MDS supported the relative superiority of this model. Both mBART and mT5 showed low performance in both MDS and human evaluation. MDS was also observed to be useful in identifying poor performance. In the Firat_DS Dataset, mBART showed the best performance in both MDS and human evaluation results. This means that MDS is not only able to capture semantic information with high accuracy but also to identify information that is opposite to that high accuracy. BERTurk and mT5 performed well against both evaluation methods but were outperformed by mBART. This showed that MDS accurately depicted the distinction between the performances of the models.

6. Conclusions

In this study, we explored the potential of models BERTurk, mT5, and mBART in performing the task of abstractive text summarization in Turkish using TR-NEWS, MLSUM, WikiLingua, and Firat_DS as datasets. The summaries and scores produced were evaluated comprehensively, and the scores we obtained from ROUGE, BERTScore, Novelty Rate, and FastText Based Cosine Similarity metrics were normalized and compared with the manual evaluation results by creating a new score that we called MDS. This comprehensive approach allowed us to gain a multifaceted understanding of summarization quality.

In ATS studies, it is essential to evaluate Rouge values and Novelty Rates, especially to reveal the effectiveness of summarization models. In addition, instead of a summarization model that summarizes the text by copying the sentences in the original text one-to-one, a model that produces the summary with new sentences is preferred. According to the ROUGE values obtained, BERTurk gave more accurate results in summarizing the automatic abstraction text in Turkish compared to other models. mT5 produced summaries with higher Novelty Rates compared to BERTurk and mBART. Except for the MLSUM dataset, BERTurk obtained the highest BERTScore values. mBART and mT5 also have BERTScore performance values close to the BERTurk model. Considering the comprehensive evaluations, it was concluded that although the summarization performances of BERTurk, mBART, and mT5 are close, each model has shortcomings and strengths.

Assessing a model's summaries in conjunction with MDS proved helpful in understanding the subtleties of a model's performance, emphasizing the positives and negatives in detail. Through this extensive evaluation, we found that while the summarized performances of the BERTurk, mBART, and mT5 models were quite close, each had distinct benefits and drawbacks. This approach to evaluation proved that different metrics are indeed helpful and necessary for a thorough assessment of various models for the task of human evaluations summarization in Turkish. In addition, the correlation between the results of human evaluations and the results of MDS was a good reason to claim the efficiency of MDS in case of replacing human evaluation.

In future work, we plan to develop innovative approaches for Turkish abstractive text summarization. Performing the data summarization task in layers can enable more effective capture of different forms of information and data. In this process, the summaries produced by the model can be divided into specific layers of information and provide more prosperous and multidimensional content. Furthermore, developing personalized summarization models in the future can be an up-and-coming area. In particular such models will be especially useful in the age of Information Overload, where users will be able to indicate the total number of words they prefer to read from the summaries and the modalities. At this juncture, it is possible to construct adaptive summary models that correspond with individual user preferences and reading practices leveraging on learning devices. In particular, the multidimensional evaluation approach we propose in this study can provide a solid foundation for personalized and multidimensional summarization models in the future and more consistent results by evaluating the performance of the models from different perspectives.

References

[1] M. Zhang, G. Zhou, W. Yu, N. Huang, & W. Liu. (2022). A comprehensive survey of abstractive text summarization based on deep learning. Computational intelligence and neuroscience, 2022(1), 7132226.] [Akhmetov, I.,

Nurlybayeva, S., Ualiyeva, I., Pak, A., & Gelbukh, A. (2023). A Comprehensive Review on ATS. Computación y Sistemas, 27(4), 1203-1240.
[2] I. Mani, & M. T. Maybury (Eds.). (1999). Advances in ATS. MIT Press.
[3] D. Jain, M. D. Borah, & A. Biswas (2021). Summarization of legal documents: Where are we now and the way forward. Computer Science Review, 40, 100388.
[4] D. Suleiman and A. Awajan (2020). Deep learning based abstractive text summarization: Approaches, datasets, evaluation measures, and challenges. Mathematical Problems in Engineering, 2020, 1-29. https://doi.org/10.1155/2020/9365340.
[5] M. Allahyari, S. Pouriyeh, M. Assefi, S. Safaei, E. Trippe, J. Gutiérrez … & K. Kochut, (2017). Text summarization techniques: A brief survey. https://doi.org/10.48550/arxiv.1707.02268.
[6] S. Gehrmann, Z. Ziegler, & G. Rushton, (2019). Generating abstractive summaries with fine-tuned language models. https://doi.org/10.18653/v1/w19-8665.
[7] A. See, P. J. Liu & C. D. Manning (2017). Get to the point: Summarization with pointer-generator networks. arXiv preprint arXiv:1704.04368.
[8] D. Bahdanau, K. Cho &Y. Bengio (2014). Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473.
[9] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit,, L. Jones, A. N. Gomez, ... & I. Polosukhin. (2017). Attention is all you need. Advances in neural information processing systems, 30.
[10] J. Devlin, M. W. Chang, K. Lee, & K. Toutanova (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.
[11] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, ... & D. Amodei. (2020). Language models are few-shot learners. Advances in neural information processing systems, 33, 1877-1901.
[12] W. S. El-Kassas, C. R. Salama, A. A. Rafea, & H. K. Mohamed (2021). ATS: A comprehensive survey. Expert systems with applications, 165, 113679.
[13] B. Baykara & T. Güngör (2023). Turkish abstractive text summarization using pre-trained sequence-to-sequence models. Natural Language Engineering, 29(5), 1275-1304.
[14] M. Ülker, & A.B. Özer (2021, June). TTSD: A novel dataset for Turkish Text Summarization. In 2021 9th International Symposium on Digital Forensics and Security (ISDFS) (pp. 1-6). IEEE.
[15] F. B. Fikri, K. Oflazer, & B. Yanikoglu (2021, August). Semantic similarity based evaluation for abstractive news summarization. In *Proceedings of the 1st workshop on natural language generation, evaluation, and metrics (GEM 2021)* (pp. 24-33).
[16] B. Baykara & T. Güngör (2022). Abstractive text summarization and new large-scale datasets for agglutinative languages Turkish and Hungarian. Language Resources and Evaluation, 56(3), 973-1007.
[17] A. Safaya, E. Kurtuluş, A. Göktoğan,, & D. Yuret (2022). Mukayese: Turkish NLP strikes back. arXiv preprint arXiv:2203.01215.
[18] R. Bech, F. Sahin, & M. F. Amasyali (2022, September). Improving Abstractive Summarization for the Turkish Language. In 2022 Innovations in Intelligent Systems and Applications Conference (ASYU) (pp. 1-6). IEEE.
[19] B. Ay, F. Ertam, G. Fidan, & G. Aydin (2023). Turkish abstractive text document summarization using text-to-text transfer transformer. Alexandria Engineering Journal, 68, 1-13.
[20] B. Baykara & T. Güngör (2023, June). Morphosyntactic Evaluation for Text Summarization in Morphologically Rich Languages: A Case Study for Turkish. In *International Conference on Applications of Natural Language to Information Systems* (pp. 201-214). Cham: Springer Nature Switzerland.
[21] Y. Yüksel., & Y. Çebi (2021, October). TR-SUM: An ATS Tool for Turkish. In the International Conference on Artificial Intelligence and Applied Mathematics in Engineering (pp. 271-284). Cham: Springer International Publishing.
[22] S. Hochreiter, & J. Schmidhuber. (1997). Long short-term memory. Neural Computation, 9(8), 1735-1780.
[23] Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., ... & Zettlemoyer, L. (2019). Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. arXiv preprint arXiv:1910.13461.
[24] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, ... & P. J. Liu, (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of Machine Learning Research, 21(140), 1-67.
[25] S. Rothe, S. Narayan, & A. Severyn (2020). Leveraging pre-trained checkpoints for sequence generation tasks. Transactions of the Association for Computational Linguistics, 8, 264-280.
[26] S. Schweter, (2020). Berturk-bert models for Turkish, April 2020. URL https://doi. org/10.5281/zenodo, 3770924.
[27] L. Xue, N. Constant, A. Roberts, M. Kale, R. Al-Rfou, A. Siddhant, ... & C. Raffel, (2020). mT5: A massively multilingual pre-trained text-to-text transformer. arXiv preprint arXiv:2010.11934.
[28] Y. Liu. (2020). Multilingual denoising pre-training for neural machine translation. *arXiv preprint arXiv:2001.08210*.
[29] T. Scialom, P. A. Dray, S. Lamprier, B. Piwowarski, & J. Staiano, (2020). MLSUM: The multilingual summarization corpus. arXiv preprint arXiv:2004.14900.
[30] F. Ladhak, E. Durmus, C. Cardie, & K. McKeown, (2020). WikiLingua: A new benchmark dataset for cross-lingual abstractive summarization. arXiv preprint arXiv:2010.03093.

[31] C. Y. Lin (2004, July). Rouge: A package for automatic evaluation of summaries. In Text summarization branches out (pp. 74-81).
[32] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, & Y. Artzi, (2019). Bertscore: Evaluating text generation with Bert. arXiv preprint arXiv:1904.09675.
[33] P. Bojanowski, E. Grave, A. Joulin, & T. Mikolov, (2017). Enriching word vectors with subword information. Transactions of the association for computational linguistics, 5, 135-146.

Article Information Form

Authors Contributions: All authors contributed equally to the writing of this paper. All authors read and approved the final manuscript.

Conflict of Interest Notice
The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Ethical Approval and Informed Consent
It is declared that during the preparation process of this study, scientific and ethical principles were followed, and all the studies benefitted from are stated in the bibliography.

Plagiarism Statement: This article has been scanned by iThenticate ™.

SAKARYA UNIVERSITY JOURNAL OF COMPUTER AND INFORMATION SCIENCES
http://saucis.sakarya.edu.tr/
e-ISSN: 2636-8129
Publisher: Sakarya University
Vol. 7, No. 3, 2024, 361-377
DOI: 10.35377/saucis...1495856
SAKARYA UNIVERSITY
RESEARCH ARTICLE

A Hybrid Approach for Color Face Recognition Based on Image Quality Using Multiple Color Spaces

Mohammad Mehdi Pazouki[1], Önsen Toygar[1*], Mahdi Hosseinzadeh[2]

[1] Eastern Mediterranean University, Faculty of Engineering, Department of Computer Engineering, Gazimağusa, Turkish Republic of Northern Cyprus
[2] Tarbiat Modares University, Faculty of Electrical and Computer Engineering, Department of Computer Engineering, Tehran, Iran

Corresponding author:
Önsen Toygar, Eastern Mediterranean University, Faculty of Engineering, Department of Computer Engineering, Gazimağusa, Turkish Republic of Northern Cyprus
onsen.toygar@emu.edu.tr

Article History:
Received: 04.06.2024
Accepted: 18.09.2024
Published Online: 29.11.2024

ABSTRACT
In this paper, the color face recognition problem is investigated using image quality assessment techniques and multiple color spaces. Image quality is measured using No-Reference Image Quality Assessment (NRIQA) techniques. Color face images are categorized into low, medium, and high-quality face images through the High Low Frequency Index (HLFI) measure. Based on the categorized face images, three feature extraction and classification methods as Scale Invariant Feature Transform (SIFT), Speeded Up Robust Features (SURF), and Convolutional Neural Networks (CNN) are applied to face images using RGB, YCbCr, and HSV color spaces to extract the features and then classify the images for face recognition. To enhance color face recognition systems' robustness, a hybrid approach that integrates the aforementioned methods is proposed. Additionally, the proposed system is designed to serve as a secure anti-spoofing mechanism, tested against different attack scenarios, including print attacks, mobile attacks, and high-definition attacks. A comparative analysis that assesses the proposed approach with the state-of-the-art systems using Faces94, ColorFERET, and Replay Attack datasets is presented. The proposed method achieves 96.26%, 100%, and 100% accuracies on ColorFERET, Replay Attack, and Faces94 datasets, respectively. The results of this analysis show that the proposed method outperforms existing methods. The proposed method showcases the potential for more reliable and secure recognition systems.

Keywords: Face recognition, Image quality assessment measures, Color spaces, Feature extraction, Deep learning

1. Introduction

Exploring the recent technology for face recognition has shown that it is an important application in the world, especially for passport control, entrance to secure systems, and authorization for mobile phones and other technical devices. A digital image or a single frame extracted from a video recording can be used for the identification or verification of faces [1]. The technology analyzes specific facial features in an image and compares them with those in a database of faces stored in a central repository. In particular, applications based on color face images perform significantly better than those using grayscale facial images, largely because grayscale conversion separates luminance from chrominance, and luminance is more efficient for detecting visual features [2]. Successful recognition of face images requires the inclusion of multiple attributes of the face, namely orientation, position, expression, color, and scale [3]. However, color alone does not provide a substantial advantage in face recognition beyond luminance information [4]. Many face recognition methods rely solely on luminance data, converting color face images into grayscale images to streamline processing, as color adds complexity and slows down computation [2].

In the literature, Olayede et al., in 2020, presented the recent challenges and approaches for face recognition systems [3]. The authors indicated that under the illumination challenge, the best result was 90.38% on the ORL database using the Convolutional Neural Networks (CNN) approach. In the presence of pose variations, the highest performance was obtained as 93.5% on the PIE dataset using a 3D reconstruction procedure based on facial landmarks and sparse regression. Face recognition under various expressions was also discussed and the best recognition rate was achieved as 89.76% on the JAFFE dataset using the CNN approach. On the other hand, in the presence of occlusions, the highest recognition rate was reported as 92.5% on the AR dataset using a fuzzy max-pooling approach based on CNN [3].

Recently, in 2023, Rusia and Singh [4] presented a comprehensive survey on face recognition approaches and challenges. The authors reviewed the fusion of feature-based and texture-based methods that represent 91% accuracy under plastic

Cite as: M. M. Pazouki, Ö. Toygar, M. Hosseinzadeh, "A Hybrid Approach for Color Face Recognition Based on Image Quality Using Multiple Color Spaces," *Sakarya University Journal of Computer and Information Sciences*, vol.7, no. 3, pp. 361-377, 2024. Doi: 10.35377/saucis…1495856

surgery challenges. Besides, for detecting facial makeup, a 93.5% detection rate was mentioned which was achieved using shape, texture, and skin color analysis-based methods with RGB, Lab, and HSV color spaces. Additionally, face-based indirect spoofing was discussed with photo attack, video attack, and mask attack categories. The best error rate was reported as 0.024 Half Total Error Rate (HTER) on the Replay Attack dataset using an image quality assessment based fast non-intrusive method with Linear Discriminant Analysis (LDA) and Support Vector Machine (SVM) methodologies [4].

Moreover, several color spaces have been studied for performing color face recognition. For instance, Yang et al. [5] introduced a global Eigen scheme that handles color components independently, showing potential improvements in face recognition with color information. Yip and Sinha [6] developed a non-negative matrix factorization (NMF) approach that outperforms grayscale methods in recognizing color face images. While some important investigations of the researchers emphasize the role of color cues in recognition [7], the rest of the researchers indicate that color does not significantly alter face recognition processes since it does not affect shape-from-shading mechanisms [8]. Nevertheless, it can be stated that color is important in face recognition technology, especially in high-resolution images.

The most widely used color spaces, namely YCbCr, RGB, and HSV are utilized in this study to assess image quality under various conditions. The aforementioned color spaces are analyzed for each channel individually, afterwards, the concatenation of outputs is employed. Several feature extraction and/or classification techniques such as Principal Component Analysis (PCA), Local Binary Patterns (LBP), Color Local Binary Patterns (ColorLBP), Scale Invariant Feature Transform (SIFT), Speeded Up Robust Features (SURF), Oriented FAST and rotated BRIEF (ORB), and Convolutional Neural Networks (CNN), are employed to extract or classify the facial features in face recognition experiments. Additionally, Image Quality Measures (IQM) and No-Reference Image Quality Measure (NR-IQM) metrics are used to measure the image quality and distinguish between real and fake face images [9], [10]. All the aforementioned techniques for feature extraction and classification are integrated in this study and a hybrid color face recognition algorithm for color face images with different image qualities is proposed. Consequently, the experimental results using multiple feature extraction methods and color spaces on publicly available datasets are presented.

The novelty of the paper can be summarized with the following contributions:

- This study combines the usage of image quality by utilizing No-Reference Image Quality Assessment techniques and employing various hand-crafted feature extraction methods, including ColorLBP, LBP, SIFT, SURF, ORB, and PCA to extract features from both color and grayscale images.
- Hand-crafted feature extraction methods and deep learning-based Convolutional Neural Networks are used together for color face recognition in a hybrid approach.
- Three different color spaces (YCbCr, HSV, and RGB) across three face databases (ColorFERET, Faces94, and Replay Attack) are applied.
- High-Low Frequency Index (HLFI) quality measure is employed in determining image quality, surpassing other no-reference quality measures.
- For medium-quality images, the texture-based SIFT feature extraction method combined with RGB color space is found to be the most suitable. Deep-learning-based CNN demonstrates high performance on low-quality images, particularly in the HSV color space. The YCbCr color space, in conjunction with the SURF feature extraction method, proves to be the triumphant approach for high-quality images.
- The proposed approach is a hybrid algorithm in which the outcomes are promising, outperforming state-of-the-art face recognition approaches on the same datasets.

The remaining sections of the paper cover fundamental concepts, literature review, the proposed method, experimental results, and concluding remarks, all contributing to the understanding and advancement of face recognition technology.

2. Materials and Methods

The databases used in the experiments; materials and methodologies employed in this study such as image quality assessment techniques, color spaces, feature extractors, and classification methods; and the details of the proposed method are explained in the following subsections.

2.1. Databases

Three color face databases are used to measure the quality level of face images and evaluate the performance of color face recognition methods. The databases used are Replay-Attack (Kumar et al. [11]), Faces94 (Wang et al. [12]), and ColorFeret (Rowley et al. [13]).

The Replay-Attack database, sourced from Idiap, is a valuable resource for studying face spoofing. It comprises 1300 video clips capturing attempts at image and movie attacks on 50 subjects, conducted under varying illumination conditions. These color videos, captured by a webcam at 320x240 pixels resolution, provide rich data for analysis. In our study, we carefully selected 100 real images, with 50 designated for testing and the remaining 50 for training purposes. These images encompass

three distinct attack types, namely High-Def, Mobile, and Print. The High-Def attack involves displaying high-quality images on a tablet, while the Mobile attack showcases images on a mobile screen and presents them to the photo lens. The Print attack involves printing the image and presenting it to the photo lens. For each attack type, we curated 100 images from the Replay-Attack database, allocating 50 for training and 50 for testing.

Conversely, the Faces94 database boasts a collection of 153 images, each rendered at a resolution of 180x200 pixels. This database thoughtfully segregates its images into distinct categories, delineating between male and female individuals. Specifically, it comprises 20 images of females, 113 of males, and 20 of male staff members. The majority of subjects represented in this database are first-year undergraduate students, predominantly falling within the age range of 18 to 20 years. The imagery within Faces94 exhibits a deliberate artificial illumination setting, with some subjects captured wearing glasses. Furthermore, the photos exhibit a blend of both tungsten and fluorescent lighting conditions, adding diversity to the dataset for comprehensive analysis.

The expansive ColorFERET database boasts a staggering collection of 11,338 color images, each generously sized at 512 x 768 pixels. These images were meticulously captured against a semi-controlled backdrop, featuring subjects adopting 13 distinct poses. Impressively, the database encompasses imagery from a diverse pool of 994 subjects. The training stage of the machine relies on the standard frontal image set (denoted as Fa), where one frontal image per subject is employed. Subsequently, the system's prowess is assessed on a different frontal pose drawn from the complementary set (referred to as Fb). In the context of the experiments conducted in this study, a subset comprising 268 training images and 268 test images is judiciously utilized to ensure robust evaluation.

Table 1 provides a concise overview of the image distribution utilized for both training and testing across the three pivotal databases featured in our experimental analysis. Furthermore, a visual glimpse into these datasets is given in Figure 1. A compelling showcase of color face images sourced from the Replay Attack, Faces94, and ColorFERET datasets are presented in Figure 1 (a), (b), and (c), respectively. In particular, Figure 1a thoughtfully presents an assortment of sample images from the Replay Attack database, illustrating high-definition attacks, mobile attacks, and print attacks.

A series of comprehensive experiments have been conducted in the realm of face recognition utilizing color-rich images. These experiments leverage the power of No-Reference Image Quality Measurement (NR-IQM) methods to ascertain the quality levels of the images under scrutiny. Four distinct IQM methods were meticulously implemented to accomplish this. Through rigorous evaluation, the most suitable IQM method was judiciously chosen for integration into the image quality determination process.

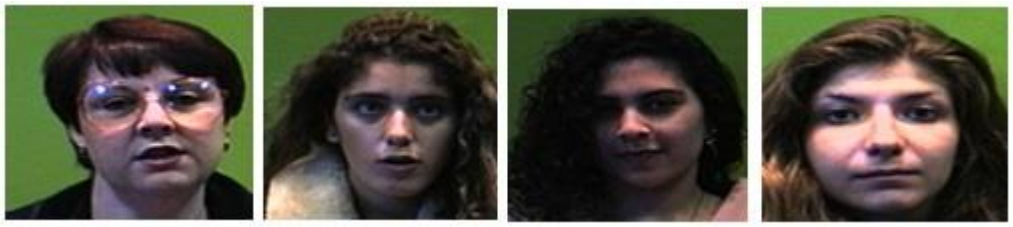
Furthermore, these experiments ventured into the exploration of three distinctive color spaces: RGB, HSV, and YCbCr, each offering a unique perspective on image representation. Additionally, a robust array of seven feature extraction methods, including PCA, LBP, ColorLBP, SIFT, SURF, ORB, and CNN, were meticulously implemented to capture the diverse facets of image characteristics.

The experiments are conducted on a Windows 10 Professional OS, Skylake 6700k Core i7 CPU operating at 4.3 GHz and 16 GB of 2400 Hz dual channel RAM. The version of Python is 3.6 and the OpenCV is 3.4. The following subsections provide an in-depth presentation of these experiments.

Table 1. Summary of Datasets and Experimental Setup Used in the Experiments

Dataset	Test images			Train images		
Faces94	1520 real			1520 real		
ColorFERET	268 real			268 real		
Replay Attack	150 fake			150 fake		
	50 real			50 real		
	50 Print Attack	50 Mobile Attack	50 High-def Attack	50 Print Attack	50 Mobile Attack	50 High-def Attack

(a)



(b)

(c)

Figure 1. Sample Color Face Images from (a) Replay Attack (b) Faces94 and (c) ColorFERET

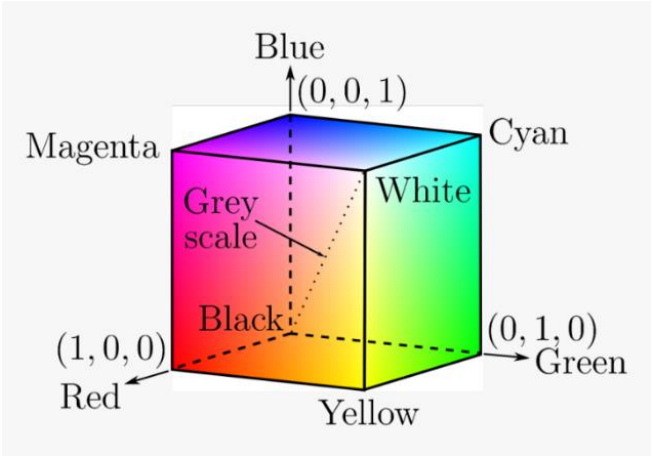
2.2. Materials

Image processing investigates several color channels in which the most widely used is RGB color space. Investigation of color face images in the RGB domain showed some limitations due to the close relationship between the channels of red (R), green (G), and blue (B). In this research, our primary goal is to use a color space with reduced inter-element correlation to enhance the performance of face classification. To achieve this objective, a combination of diverse color spaces is employed in the proposed color face recognition method, including RGB color space, HSV, and YCbCr color spaces. Color space models are shown in Figure 2 [14]. HSV is renowned for its alignment with human perception and encompasses hue, saturation, and value components. In contrast, the YCbCr color domain which was initially designed for image compression, separates the luminance (Y) channel from chrominance channels (Cb, Cr). On the other hand, as another advantage, YCbCr also serves as a functional space for the segmentation of skin color applications [4].

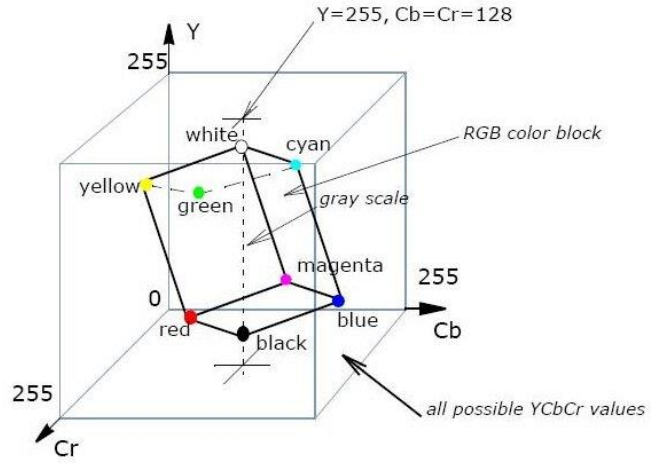
Image quality assessment is a significant field that is employed for many image processing applications. Within this field, Full Reference Image Quality Measurement (FR-IQM) and Reduced Reference Image Quality Measurement (RR-IQM) methods can be used on complete or partial images to evaluate visual quality effectively [9]. On the other hand, it is also possible not to use a reference image and measure the quality. Specifically, No-Reference Image Quality Measures provide a quality measurement of the images without using reference images [10]. Image quality can be predicted accurately by the majority of NR-IQM metrics for a single type of distortion. There are four notable NR-IQMs employed in this study: Blind Image Quality Index (BIQI), JPEG Quality Index (JQI), Naturalness Image Quality Estimator (NIQE) and High-Low Frequency Index (HLFI).

The first quality metric employed in this study is JQI which is successful in estimating image quality when block-based algorithms, which are common in compression methods like Joint Photographic Experts Group (JPEG) [2], introduce distortion at low bit rates. Meanwhile, the second quality metric is HLFI which demonstrates sensitivity to image sharpness by calculating the difference in power between lower frequency values and upper frequency values in the Fourier Spectrum. Blind Image Quality Index is the third metric employed which capitalizes on natural scene statistics (NSS) and prior knowledge of unaltered natural scene images to train its primary model. Lastly, the fourth metric is NIQE, which is a comprehensive blind image quality analyzer that constructs an informed quality profile through a Gaussian multivariate model of natural scenes. The assessment of all image quality measures is performed collectively focusing on improving the accuracy and reliability. Methods that fail for a specific application to deliver appropriate and detailed evaluations are excluded from consideration.

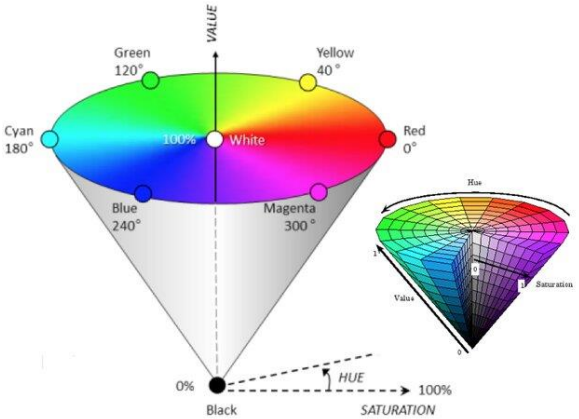
Conversely, the most prominent step for face recognition is the feature extraction step. Facial feature extraction specifically involves the isolation of essential facial components including the nose, mouth, and eyes. This process stands as a fundamental prerequisite for initiating various sophisticated techniques, such as facial tracking, face recognition, and facial expression recognition. Feature extraction can be classified into several categories. One of the most widely used categories is appearance-based feature extraction methods. As an example of this category, the Principal Component Analysis (PCA) method performs data summarization by condensing a comprehensive set of variables into a reduced set through specific transformations. These transformations aim to convert linearly correlated variables into uncorrelated ones, effectively simplifying the data.



(a)



(b)



(c)

Figure 2. Color Models (a) RGB Color Space, (b) YCbCr Color Space, (c) HSV Color Space [14]

On the contrary, texture-based feature extraction methods involve several methods such as Local Binary Patterns (LBP) and Scale Invariant Feature Transform (SIFT) that are based on the texture of images and known as local descriptors. The fundamental concept behind LBP revolves around the replacement of pixel values within an image with decimal characters to produce a code that extracts the local structural information surrounding each pixel [15]. In this process, a central pixel undergoes analysis in conjunction with its eight neighboring pixels. Pixels with values lower than the center pixel are assigned bit 0, while the others that are equal to or greater than the center pixel receive bit 1. Consequently, for every center pixel, a binary number is created by concatenating these binary digits in a clockwise fashion, starting from one of its top-left neighbors.

Color Local Binary Patterns (ColorLBP) method is another texture-based feature extraction method that is a variant of the LBP method. ColorLBP method is tailored for the analysis of color images, which comprise three distinct channels in RGB color space, representing the blue, green, and red color components. In the realm of color space analysis, various methods offer distinct advantages and applications. RGB images, characterized by their red, green, and blue channels, are known for their clarity and adaptability under varying conditions. This is achieved by normalizing these color elements within the color space of RGB. Contrarily, different color spaces can also be considered for extracting features from images within the ColorLBP method. For example, the color space of HSV is designed to emulate the visual system's perception of color with three channels as hue, which defines color; saturation, representing chrominance; and value, specifying luminance. Similarly, the other color space, namely YCbCr, divides the image into three channels. The first channel is the luminance (Y) while the remaining channels are chrominance blue (Cb) and chrominance red (Cr). In general, the processing steps for these color spaces follow a similar pattern. Condensation is applied to each channel's histogram independently and then concatenation of histograms is conducted to extract the whole feature vector.

Another powerful texture-based feature extraction method is the Scale-Invariant Feature Transform (SIFT) [16]. SIFT is renowned for its important ability to identify and characterize features in different scales, making it invaluable for tasks like image recognition, classification, image registration, and 3D reconstruction. Similarly, Speeded-Up Robust Features (SURF) [17] is a significant variant of SIFT which emerges as an efficient alternative to SIFT. It represents a faster iteration of SIFT with high speed and robustness against image transformations. The SURF method is a powerful and fast feature detector and descriptor that is adaptable for several tasks such as image classification, object recognition, and 3D reconstruction.

Oriented FAST and rotated BRIEF (ORB) method was proposed in 2011 [18] as another alternative for local descriptors. ORB offers a rapid and robust solution for tasks like object recognition and 3D reconstruction. The method builds upon the FAST keypoint detector and employs a modified version of the visual descriptor known as Binary Robust Independent Elementary Features (BRIEF). ORB method is aimed at providing a reliable and efficient alternative to the SIFT algorithm.

Recently, the development of deep learning techniques attracted many researchers in many fields such as face recognition. In computer vision and biometrics fields, deep learning techniques such as Convolutional Neural Networks (CNN) are universalized [3, 4]. Object detection, tracking, surveillance, and object recognition tasks have all harnessed the power of CNN. In the realm of surveillance applications, CNN finds widespread use, particularly in face recognition. In this respect, numerous university automation systems and intelligent entry management solutions have successfully incorporated face recognition technology [19] into their operations.

Deep learning approaches such as Convolutional Neural Networks have shown superior results for detecting and recognizing faces across diverse conditions [3, 4]. These conditions may include scenarios where intra-class and inter-class variations exist in the facial images. These variations can be listed as facial expressions, partial occlusions, illumination variations, the similarity between identical twins' faces [20], and so on.

2.3. Proposed Method

This study proposes a novel and hybrid approach for the recognition of facial images across various quality levels, encompassing low, medium, and high-quality color images. To achieve this, an extensive series of preliminary experiments were conducted, exploring different color spaces and diverse techniques for extracting features in face recognition. The outcome of these experiments led to the development of a versatile color-based face recognition system employing a hybrid methodology. To determine the most suitable quality assessment measure for this study, various image quality assessment metrics were applied, ultimately selecting HLFI for this purpose. Furthermore, numerous methods to extract facial features including LBP, SIFT, SURF, ORB, and CNN, were independently implemented for both grayscale and color images within the RGB, HSV, and YCbCr color spaces.

This study adopts a tailored approach based on image quality: low-quality images are processed using the CNN approach within the HSV color space, medium-quality images employ the SIFT feature extractor using RGB color space, and high-quality images utilize the SURF method in the YCbCr color space. Each of these scenarios was thoroughly explored in preliminary experiments to determine the most effective feature extractor and color space for the respective image quality category. In terms of matching and classification, Manhattan Distance measurement is employed, and the face classification is carried out using the Nearest Neighbor classifier or a CNN model. This matching and classification process is conducted using a comparison with the test image features with the corresponding using the same color space to train image feature sets,

ultimately yielding the system's decision (ID). A detailed and step-by-step algorithm of the proposed method is presented in Algorithm 1. The flowchart of the proposed method is shown in Figure 3. Every phase of the suggested approach shown in the flowchart is explained in detail in the following subsections.

Algorithm 1. Proposed Method for Color Face Recognition

```
1   Start
2   Read the training images and calculate HLFI value for
    each of them and apply preprocessing.
3   Check the average HLFI value and using the threshold
    values, determine whether an image is low, high, or
    medium-quality image. Store the quality type of
    training images.
4   Use CNN approach to extract features from low-quality
    images in HSV color space. Go to step 7.
5   If the image is medium quality image, then apply
    feature extraction on RGB color space using SIFT
    technique. Go to step 7.
6   If the image is high quality image, then apply feature
    extraction on YCbCr color space using SURF technique.
    Go to step 7.
7   Perform matching and classification after applying
    preprocessing, HLFI calculation and image quality type
    decision on test images according to the same threshold
    values as in training.
8   Report the decision of the classifier.
9   Stop
```

2.3.1. Reading Train Images, Applying HLFI Quality Metrics and Preprocessing

First of all, the initial stage of the proposed method helps to read the database images and convert them into the selected image format. Afterward, the images are saved in specific directories so that it is easier to read them. Since the size of each image is different, resizing the images to the same size is performed. Image quality is then studied by calculating the HLFI image quality measure [21] on each training image as follows:

$$HLFI(I) = \frac{\sum_{i=1}^{i_l}\sum_{j=1}^{j_l}|F_{i,j}| - \sum_{i=i_h+1}^{N}\sum_{j=j_h+1}^{M}|F_{i,j}|}{\sum_{i=1}^{N}\sum_{j=1}^{M}|F_{i,j}|} \quad (1)$$

The High-Low Frequency Index (HLFI) feature is sensitive to image sharpness and this is performed by computing the difference between the power in the lower and upper frequencies of the Fourier Spectrum. In HLFI calculation as shown in (1), jl, jh, il, ih are the indices corresponding to the lower and upper-frequency thresholds, M and N demonstrate the number of rows and columns of the images, Fij is the Fourier transform of image matrix I. In this study, the values of the parameters used are jl = jh = 0.15M and il = ih = 0.15N.

2.3.2. Classifying Image Quality by Specifying Threshold

Using the HLFI image quality measure, the individual images are examined and the results are saved. Using the average HLFI method, a threshold is found to recognize the quality of each image and then the quality type is stored as high-quality, medium-quality, and low-quality. Threshold values for each quality type are specified as described in the next section using the average HLFI values based on preliminary experiments on RGB, HSV, and YCbCr color spaces.

2.3.3. Feature Extraction on Different Color Spaces

SIFT and SURF methods are used as feature extractors in this study on different channels of color spaces. Additionally, deep learning-based CNN is used as a feature extractor and classifier in the proposed method.

Scale Invariant Feature Transform (SIFT) approach [16] detects the points of interest which are called keypoints and produce features using four steps as follows: (1) Scale-space Extrema Detection which identifies keypoints

at different scales; (2) Keypoint Localization which refines the location and scale of keypoints; (3) Orientation Assignment that assigns orientations to keypoints; and (4) Keypoint Descriptor that generates descriptors for keypoints. Detecting the keypoints in the SIFT framework starts with convolving the image with Gaussian filters at different scales, and then the difference of successive Gaussian-blurred images is calculated. Keypoints are then obtained as maxima/minima of the Difference of Gaussians (DoG) that occur at multiple scales.

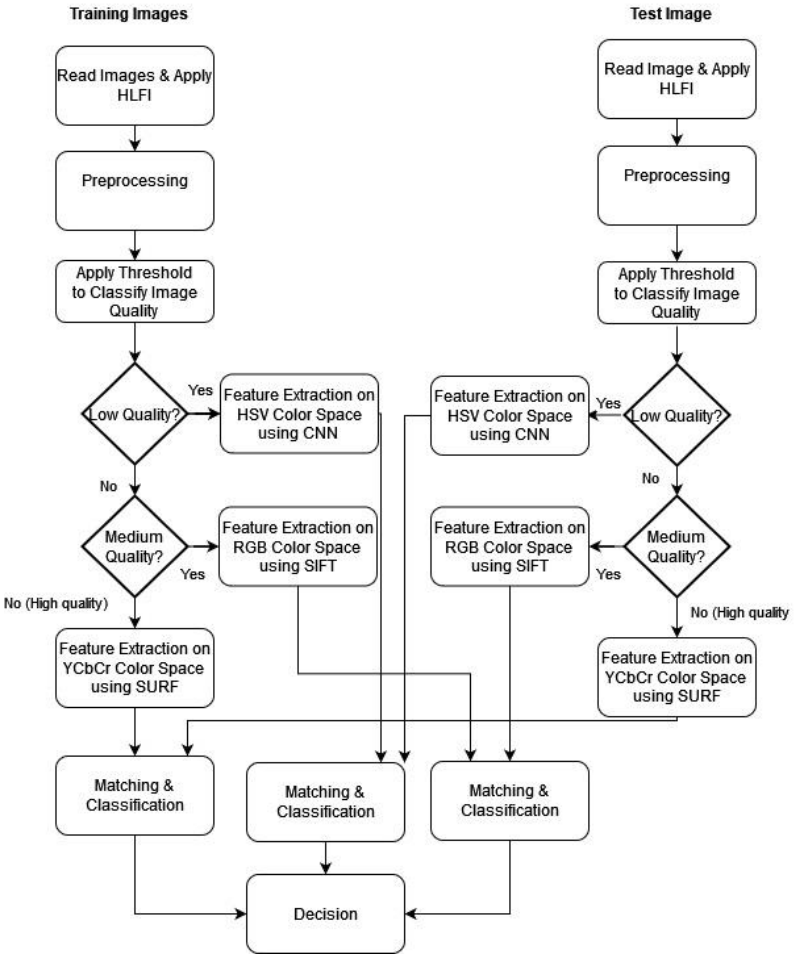


Figure 3. Proposed method flowchart

Gaussian function is the only possible scale-space kernel under a variety of reasonable assumptions. Consequently, the scale space of an image is defined as a function, $L(x, y, \sigma)$, that is produced from the convolution of a variable-scale Gaussian, $G(x, y, \sigma)$, with an input image, $I(x, y)$ as follows:

$$L(x, y, \sigma) = G(x.y, \sigma) * I(x, y), \tag{2}$$

where * is the convolution operation in x and y, and

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}. \tag{3}$$

To detect stable keypoint locations in scale space, scale-space extrema in the difference-of-Gaussian function convolved with the image, $D(x, y, \sigma)$, is proposed [16] that can be computed from the difference of two nearby scales separated by a constant multiplicative factor k as follows:

$$\begin{aligned} D(x, y, \sigma) &= \big(G(x, y, k\sigma) - G(x, y, \sigma)\big) * I(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma). \end{aligned} \tag{4}$$

The function in (4) is a particularly efficient function to compute since a simple image subtraction is performed and the smoothed image, L, can be computed in any case for scale space feature description.

Speeded Up Robust Features (SURF) [17] is a local feature detector and descriptor which is a faster variant of the SIFT

approach. The algorithm has three main parts: (1) Interest point detection; (2) Local neighborhood description; (3) Matching. SURF approach detects the interest points using an integer approximation of the determinant of the Hessian blob detector that can be computed with 3 integer operations using a precomputed integral image. The feature descriptor of SIFT is based on the sum of the Haar wavelet response around the point of interest which can also be computed with the integral image. Using the multi-resolution pyramid technique, the image is transformed into coordinates to copy the original image with a Pyramidal Gaussian or Laplacian Pyramid shape. This operation aids in obtaining an image of the same size but with reduced bandwidth. Consequently, a special blurring effect on the original image is achieved which is called Scale-Space. The process helps to produce points of interest that are scale invariant.

Specifically, SURF uses square-shaped filters as an approximation of Gaussian smoothing [17]. Therefore, filtering the image with a square is much faster if the integral image is used as follows:

$$S(x,y) = \sum_{i=0}^{x} \sum_{j=0}^{y} I(i,j) \quad (5)$$

The integral image in (5) helps to quickly evaluate the sum of the original image within a rectangle that requires evaluations at the rectangle's four corners.

To find points of interest with SURF, a blob detector based on the Hessian matrix is used. The determinant of the Hessian matrix is calculated. Afterward, it is used as a measure of local change around the point. The points are selected where the determinant is maximal. Moreover, SURF uses the determinant of the Hessian for choosing the scale. Given a point $p = (x,y)$ in an image I, the Hessian matrix $H(p,\sigma)$ at point p and scale σ, is as follows:

$$H(p,\sigma) = \begin{pmatrix} L_{xx}(p,\sigma) & L_{xy}(p,\sigma) \\ L_{xy}(p,\sigma) & L_{yy}(p,\sigma) \end{pmatrix} \quad (6)$$

where $L_{xx}(p,\sigma)$ is the convolution of the second-order derivative of Gaussian with the image $I(x,y)$ at the point p.

Specifically, the box filter of size 9×9 is an approximate of a Gaussian with $\sigma = 1.2$ which represents the lowest level with the highest spatial resolution for blob-response maps.

On the other hand, Convolutional Neural Network (CNN) [22] is a deep learning-based method and an improved version of feedforward neural networks that extracts features from input data using convolution structures. The inspiration for CNN architecture comes from visual perception. Four specific components are needed to build a CNN model as follows: (1) Padding, which enlarges the input with zero values in order not to lose information on the borders of the input image; (2) Convolution, which is a pivotal step for feature extraction using activation functions or convolution kernel; (3) Pooling (or downsampling), that is used to prevent redundancy with max pooling and average pooling; (4) Classification with Fully Connected (FC) Layers, that classifies the input and produces the output. An example of a CNN procedure is shown in Figure 4.

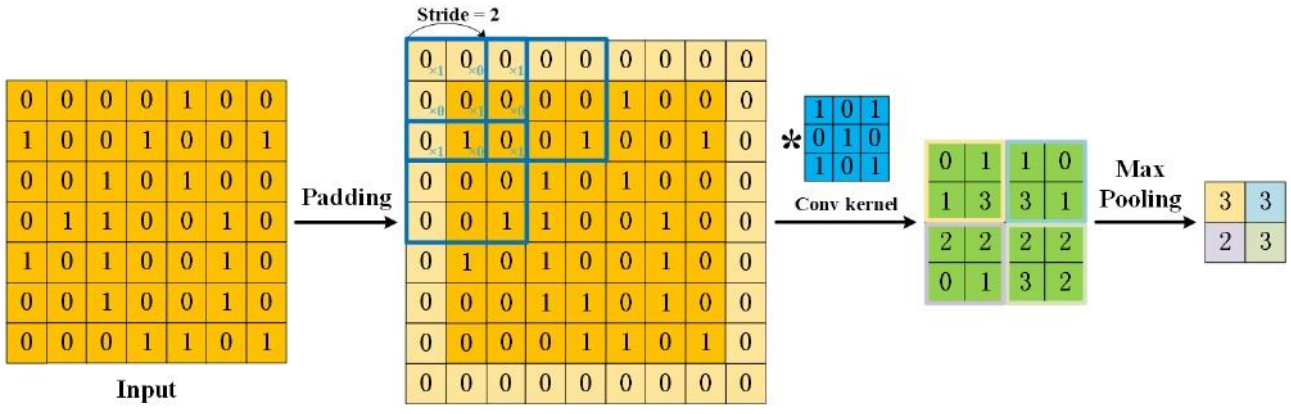


Figure 4. A 2D Convolutional Neural Network Procedure [22]

In the proposed method, SIFT, SURF, and CNN feature extraction and classification approaches are used as follows. The preprocessed images that are classified as low, medium, or high-quality images in the previous steps go through the feature extraction process with different feature extraction methods on different color spaces. According to the preliminary experiments which tried all possibilities based on the implemented feature extractors in this study, the most appropriate

feature extractor on a specific color space is selected to be used in the feature extraction stage. Consequently, each color channel from the selected color space is processed by the feature extractor and the average of all channels is considered as the final feature set. In the proposed method, for low-quality images, a deep-learning-based CNN approach is applied to HSV color space, and the resulting feature vectors are used as the output of the feature extraction stage for low-quality images. On the other hand, medium-quality images are processed with a texture-based SIFT method to extract the features on RGB color space. High-quality images are handled using the fast texture-based SURF feature extractor on YcbCr color space. The extracted feature vectors for all train images are stored to be used in the matching and classification stages.

2.3.4. Reading and Processing Test Images

Images in the test phase are read, HLFI is applied to them, and they are preprocessed similarly to training images. HLFI quality measure values are used to specify the condition level of the test images based on the specified thresholds. Consequently, the condition level of the test images is determined as high, moderated, or low condition. Afterward, the same feature extraction methods and color spaces are used for each test image according to the quality level in the same way as applied for train images. Specifically, features of low-quality test images are extracted using the CNN approach on HSV color space; for medium-quality test images, the SIFT method is applied on RGB color space; for high-quality test images, SURF features on YCbCr color space are extracted. Afterward, the extracted feature vectors for each test image are used for the matching and classification processes.

2.3.5. Matching & Decision

In the context of this study, face image classification is facilitated through the application of the Nearest Neighbor classifier [23]. The classification process hinges on a voting mechanism where a face image's classification is determined by the cumulative votes of its neighboring images. Consequently, the face image is allocated to the nearest neighbor that exhibits the highest resemblance.

As a result, the system employs a criterion of selecting the most akin face identification from among the pool of training images as its final decision. The accuracy of this decision is contingent on the concordance between the identification of the test image and the identification made by the system. The decision is considered accurate when the test image's ID aligns with the ID arrived at by the system. The Nearest Neighbor classifier excels in achieving robust performance through the utilization of multiple statistical patterns. Typically, this classifier necessitates comprehensive training with both genuine and counterfeit cases. When classifying a new sample, it calculates the distance to the nearest training case, and the label of the sample is determined by the class of that nearest training case. Expanding on this principle, the Nearest Neighbor classifier goes further by considering the k closest points and assigning the class label that predominates among them. The validation process, involving the input test set, enables the computation of an n-dimensional pattern vector based on the training samples, ultimately leading to classification based on minimum distance.

In the training phase, the examples consist of vectors annotated with class labels. These vectors are stored in the training stage of the nearest neighbor classification, and class labels are assigned accordingly. Notably, in this study, the Manhattan distance measure is employed for matching purposes since it is effective for face matching [20]. Subsequently, the Nearest Neighbor classifier is applied to execute the classification of the test images.

3. Results

In this section, we delve into a series of experiments conducted with the newly proposed method and offer a comparison analysis of its outcomes alongside state-of-the-art methods. The subsequent subsections delve into the finer details of our experimental setup, providing insights into database particulars. Furthermore, we explore the preliminary results of initial experiments that leverage several no-reference image quality assessment techniques to ascertain the image quality. Finally, we present the results of color face recognition experiments using our proposed method.

3.1. Experimental Results

This study places significant importance on the color aspect as a crucial criterion for face image recognition. Color-rich images, consisting of three channels, are stored in diverse formats, each with its unique advantages and drawbacks. The investigation delves into the merits of three distinct color spaces: RGB, HSV, and YcbCr, exploring their potential contributions to the recognition process.

Additionally, the study encompasses three different tiers of image quality levels. This diversity in quality levels allows for the consideration of an index that strikes a balance between ensuring a minimum image quality threshold while optimizing recognition performance. To enhance processing speed and address the challenges posed by growing image datasets, the study tackles the critical issues of program runtime and available storage space. These considerations become pivotal as the number of images escalates, and the study provides a comprehensive discussion on strategies for optimization in this context. The subsequent sections offer a detailed exploration of these aspects.

The objective of this study is to determine the similarity in the face patterns observed after extracting the features using PCA, LBP, SIFT, SURF, and ORB feature extraction methods. Measurement of Manhattan distance using two images' extracted

features for face recognition is not efficient on single-channel images or low-quality images. However, it works well with multi-channel images and extracts more useful information. They must also be tested in different color spaces.

3.1.1. Preliminary Experiments using NR-IQM Results

To classify the image quality type as high, low, or medium quality, four No-Reference Image Quality Measurement (NR-IQM) methods are employed on each dataset. The results obtained through these methods, including the (JQI), (HLFI), (BIQI), and (NIQE), are presented in Tables 2 and 3.

Table 2 showcases the experimental outcomes on the Replay Attack datasets, specifically addressing subsets such as Print Attack, Mobile Attack, and High-Def Attack. These subsets contain medium-quality images. Notably, the results reveal that RGB color-based images yield superior outcomes when utilizing the HLFI method. The HLFI method exhibits a wider range of effectiveness across different image types, making it a suitable choice for estimating image quality in this context. Similarly, the HSV color space is investigated for image quality assessment on the Replay Attack dataset, and the results indicate that the HLFI method excels in this color space as well. The extensive range of HLFI's effectiveness in the HSV color space further solidifies its appropriateness for making image quality determinations.

Additionally, the YCbCr color space is explored in the experiments with the Replay Attack dataset, and once again, the HLFI method outperforms other methods across various image types. This robust performance across image types underscores the suitability of the HLFI method for assessing image quality in the YCbCr color space within this specific scenario.

In conclusion, the HLFI method is chosen as the primary measure for assessing image quality due to its consistently low probability of misclassifying images. Table 3 provides an overview of the average No-Reference Image Quality Measurement (NR-IQM) results for images from the Faces94 and ColorFERET datasets, each analyzed in different color spaces. The range of results presented in Table 3 aids in identifying the most suitable color space for each method. Notably, HLFI emerges as the most appropriate measure for evaluating image quality across the considered datasets. This determination is based on the comprehensive evaluation of results encompassing JQI, HLFI, BIQI, and NIQE metrics.

Table 4 provides valuable insights into the no-reference image quality assessment metric HLFI, highlighting minimum, average, and maximum values. These statistics help establish thresholds for categorizing images into different quality types. Here are the key findings from Table 4. High-Quality Images: In high-quality images, the YCbCr color space exhibits a more consistent alternation pattern in HLFI values. Low-Quality Images: For low-quality images, the HSV color space demonstrates greater stability and consistency in HLFI values compared to other color spaces. Medium-Quality Images: When it comes to medium-quality images, the RGB color space outperforms other color spaces based on HLFI values. These observations underscore the importance of selecting the appropriate color space for assessing image quality, depending on the quality type (high, low, or medium) under consideration.

Table 2. Average of NR-IQM Results for Real and Fake Images on Replay Attack Dataset

Dataset	Color space	JQI	BIQI	NIQE	HLFI
Real	RGB	7.70	27.23	2.44	8271132.84
	HSV	17.57	27.35	3.94	35398.87
	YCbCr	7.74	36.77	4.21	5993760.03
Highdef Attack	RGB	7.64	26.74	2.61	9191549.95
	HSV	17.36	26.92	4.29	32722.45
	YCbCr	8.20	37.01	4.52	6101433.80
Mobile Attack	RGB	7.69	38.14	3.00	10565951.74
	HSV	17.82	30.11	4.17	36521.13
	YCbCr	8.29	41.83	4.32	6660354.29
Print Attack	RGB	8.33	32.62	2.25	8966469.80
	HSV	18.50	28.80	3.44	30806.30
	YCbCr	8.33	43.32	4.08	6087265.60
Average of All Images	RGB	7.84	31.18	2.57	9248776.08
	HSV	17.81	28.29	3.96	33862.18
	YCbCr	8.14	39.73	4.28	6210703.43

Table 3. Average of NR-IQM Results for Real and Fake Images for Faces94 and ColorFERET Databases

Dataset	Color space	JQI	BIQI	NIQE	HLFI
Faces94	YCbCr	8.87	47.97	4.73	2483901.57
	HSV	19.15	36.76	4.25	13629.71
	RGB	8.76	32.54	3.27	2982913.14
ColorFERET	YCbCr	12.61	49.35	3.71	33056379.03
	HSV	11.37	30.12	3.71	17199617.13
	RGB	10.97	29.69	3.36	49998026.51

Table 4. HLFI Values for Specifying Thresholds

Color space	High quality			Medium quality			Low quality		
	Max	Average	Min	Max	Average	Min	Max	Average	Min
YCbCr	65M	33M	2M	9-11M	5-6M	3-4M	3M	2M	1M
HSV	26M	17M	3M	45-58K	30-36K	7-14K	21K	13K	4K
RGB	73M	49M	16M	11-13M	8-10M	2-6M	73M	2M	765k

3.1.2. Face Recognition Results

The face recognition experiments conducted on three datasets using seven feature extraction methods have yielded results summarized in Table 5. Here is an overview of the experimental outcomes for each feature extraction method on each dataset:

Replay Attack Dataset: This dataset utilized 2 training images (1 real and 1 fake) and 2 testing images (1 real and 1 fake) for the experiments. Faces94 Dataset: Experiments on the Faces94 dataset involved 10 training and 10 testing images. ColorFERET Dataset: The ColorFERET dataset experiments were performed with 1 training and 1 testing scenario. The results are presented in terms of recognition rate, calculated as specified in the document. These recognition rates provide insights into the performance of various feature extraction methods on different datasets, facilitating the evaluation of their effectiveness in face recognition tasks:

$$Recognition\ rate(\%) = \frac{Number\ of\ correctly\ recognized\ test\ images}{Total\ number\ of\ test\ images} * 100 \quad (7)$$

According to the presented results in Table 5, SIFT achieves the best recognition rate (93%) on the Replay Attack dataset while SURF performed well (96.26%) on the ColorFERET dataset. CNN works very well on the Faces94 dataset with a 100% recognition rate. The proposed method combines the advantages of all these superior achievements and produces the best results on all datasets. Therefore, the recognition rates of the proposed method are superior in general on all datasets. The graphical representation of the results is also demonstrated in Figure 5.

The results of face recognition experiments using a deep-learning-based CNN model are shown in Table 6. The identification rate is low on the Replay Attack dataset because the number of images used in that dataset is low, it includes both real and fake images, and it needs to perform data augmentation to increase the number of images to achieve better results with this model. However, with Faces94 and ColorFERET datasets, the accuracies (recognition rates) are very high and the model can be used for face recognition.

3.2. Comparison with the State-of-the-art

The performance of the suggested face recognition method is compared to state-of-the-art studies on three different datasets, as presented in Tables 7, 8, and 9. On Replay Attack Dataset, as shown in Table 7, the proposed method achieves a remarkable accuracy of 100%, making it comparable to the top-performing studies by Yu et al. [24] and Benlamoudi et al. [25], which also achieved 100% accuracy. This demonstrates the excellence of the proposed approach in recognizing faces under spoofing attacks.

On Faces94 Dataset which is presented in Table 8, the proposed method excels with an accuracy of 100%, outperforming other studies. The closest result to the proposed method is from the study by Karanwal and Diwakar [26], which achieved an accuracy of 98.89%. This highlights the effectiveness of the proposed method for face recognition in the Faces94 dataset. The proposed method exhibits superior performance on the ColorFERET dataset, as shown in Table 9, with an accuracy of 96.26%. It outperforms state-of-the-art studies, with the closest performance achieved by Terhörst et al. [27] at 95.9%. This underscores the capability of the proposed approach in recognizing faces in color images. Overall, the proposed method consistently achieves high accuracy levels and demonstrates competitiveness with state-of-the-art studies across different datasets, showcasing its effectiveness in face recognition tasks. In general, three different databases are used in this study and

the results show that the proposed method is superior to the other methods across all datasets with 100%, 100%, and 96.26% on Replay Attack, Faces94, and ColorFERET datasets, respectively.

Table 5. Face Recognition Results

Dataset	Number of Face Images	Recognition Rate (%)							
		CNN	LBP	Color LBP	SIFT	SURF	ORB	PCA	Proposed Method
Faces94	3040 real	100	99.86	99.86	43.10	42.03	40.72	99.53	100
Color Feret	536 real	92.50	57.46	92.91	89.55	96.26	88.80	61.19	96.26
Replay Attack	300 fake 100 real	12.90	30.00	63.00	93.00	85.66	77.33	19.00	100

Table 6. CNN Accuracy and Loss Results

Dataset	Train		Test	
	Loss (%)	Accuracy (%)	Loss (%)	Accuracy (%)
Replay Attack	0.48	84.5	9.15	12.99
Faces94	~0	100	~0	100
Color Feret	0.04	100	0.42	92.53

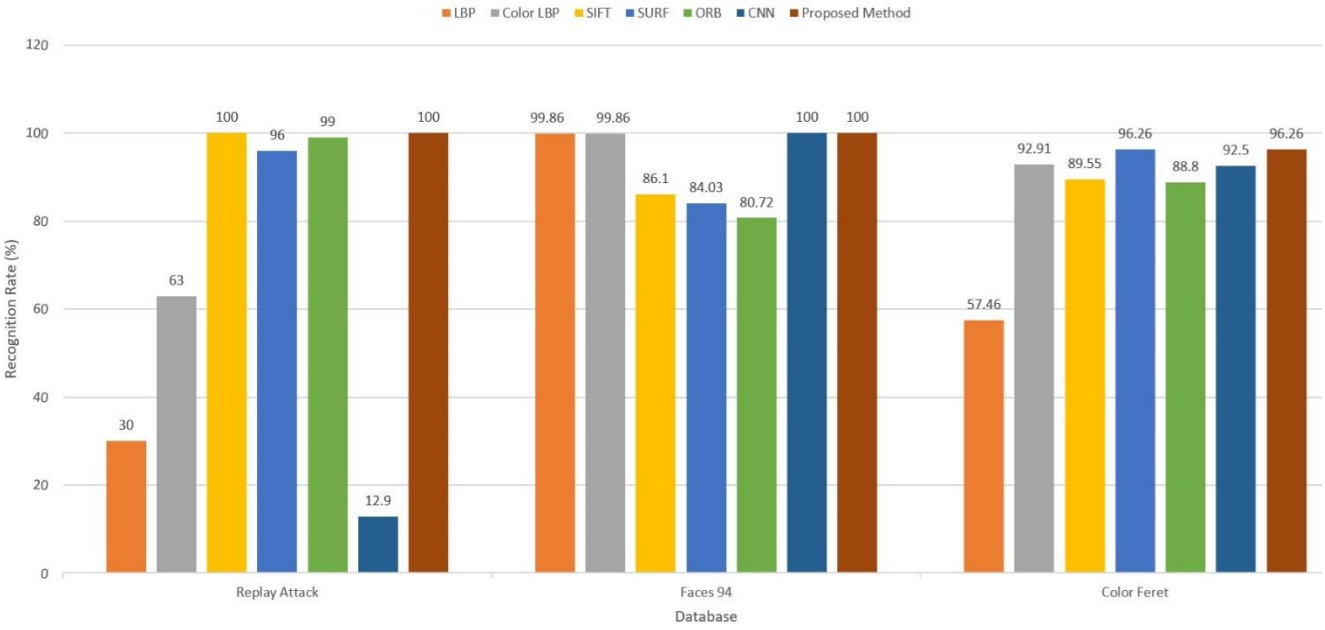


Figure 5. Face Recognition Rates on Replay Attack, Faces94, and ColorFERET Datasets

Table 7. Comparison with the State-of-the-art Face Recognition on Replay Attack Dataset

Reference	Year	Experimental Setup	Methods	Accuracy (%)
Zhang et al. [28]	2020	Train:800, Test:800	CNN	96.6%
Boulkenafet et al. [29]	2017	Train:240, Test:240	SVM+LBP	98.17%
Yu et al. [24]	2020	Train:1300, Test:500	CDCN++	99.89%
Yu et al. [24]	2020	Train:1300, Test:500	CDCN	100%
Benlamoudi et al. [25]	2022	Train:650, Test:650	BS-CNN+MV	100%
Proposed Method	2023	Train:200, Test:200	SIFT	100%

Table 8. Comparison with the State-of-the-art Face Recognition on the Faces94 Dataset

Reference	Year	Experimental Setup	Methods	Accuracy (%)
Chen et al. [31]	1998	Train:768, Test:768	LDA	79.39%
Karanwal [33]	2021	Train:1530, Test:510	CLBP	89.14%
Vinay et al. [30]	2018	Train:2295, Test:765	LBP	93.6%
Sikder et al. [32]	2021	Train:572, Test:52	PCA	94.4%
Tran et al. [34]	2017	Train:608, Test:2432	LIAD	98.16%
Karanwal and Diwakar [26]	2021	Train:1360, Test:680	CMBZZBP	98.89%
Proposed Method	2023	Train:1520, Test:1520	CNN	100%

Table 9. Comparison with the State-of-the-art Face Recognition on the ColorFERET Dataset

Reference	Year	Experimental Setup	Methods	Accuracy (%)
Chou et al. [35]	2019	Train:675, Test:675	Deep CNN	83.79%
Zhao et al. [36]	1998	Train:1316, Test:298	PCA	95%
Geng and Jiang [37]	2009	Train:1340, Test:1340	LBP	95%
Du et al. [38]	2009	Train:804, Test:268	SURF	95%
Terhörst et al. [27]	2020	Train:9125, Test:5000	kNN	95.9%
Proposed Method	2023	Train:268, Test:268	SURF	96.26%

4. Conclusion

Securing sensitive data has become a challenging task leading to significant research in biometrics. Within the realm of biometrics, robust and secure face recognition plays a pivotal role. According to this study, image quality is assessed by utilizing No-Reference Image Quality Assessment Measures (NR-IQMs). Four NR-IQMs are employed in this study, namely the Blind Image Quality Index (BIQI), JPEG Quality Index (JQI), Naturalness Image Quality Estimator (NIQE) and High-Low Frequency Index (HLFI). Besides, various feature extraction methods are employed including ColorLBP, LBP, SIFT, SURF, ORB, and PCA to extract features from both color and grayscale images. Additionally, a deep learning approach using CNN architecture is applied when a substantial volume of image data is available. The experiments encompass three different color spaces (YCbCr, HSV, and RGB) across three face databases (ColorFERET, Faces94, and Replay Attack).

The key findings from the study are as follows: The High-Low Frequency Index (HLFI) quality measure excels in determining image quality, surpassing other no-reference measures. For medium-quality images, the texture-based SIFT feature extraction method combined with RGB color space is found to be the most suitable. Deep-learning-based CNN demonstrates high performance on low-quality images, particularly in the HSV color space. The YCbCr color space, in conjunction with the SURF feature extraction method, proves to be the triumphant approach for high-quality images. As a result, the proposed approach leverages these insights and combines them into a hybrid algorithm that capitalizes on these findings.

Specifically, the proposed hybrid face recognition algorithm starts with reading the color face images and calculating the HLFI quality measure to decide on the quality of the face image as low-quality, medium-quality, or high-quality image. For low-quality images, the proposed method employs the CNN approach on the images in HSV color space. Mediaum-quality images are processed in RGB color space with the SIFT feature extractor. High-quality images are used in YcbCr color space and their features are extracted with the SURF method.

The proposed hybrid color face recognition method achieves 100% accuracy on the Replay Attack dataset with the SIFT feature extractor employed for medium-quality images of the dataset. Additionally, the accuracy of the proposed method on low-quality images of the Faces94 dataset is 100% by employing a deep learning-based CNN approach. Moreover, high-quality images in the ColorFERET dataset are recognized by the proposed method which employs a SURF feature extractor with 96.26% accuracy. The outcomes of the proposed method are promising, outperforming extant state-of-the-art face recognition approaches on the same datasets. Future research could explore further enhancements in face recognition algorithms by incorporating diverse and robust techniques for extracting features in other color spaces and investigating the potential of texture-based and deep learning-based techniques for an even more advanced face recognition system.

References

[1] A. K. Jain, A. A. Ross, and K. Nandakumar, “Introduction to Biometrics”, Springer Publishing Company, Incorporated, 2011.

[2] R. Szeliski, “Computer Vision: Algorithms and Applications”, 1st. ed., Springer-Verlag, Berlin, Heidelberg, 2010.

[3] M. O. Oloyede, G. P. Hancke, H. C. Myburgh, “A review on face recognition systems: recent approaches and challenges”, Multimed Tools Appl, 79, pp. 27891–27922, 2020. https://doi.org/10.1007/s11042-020-09261-2

[4] M. K. Rusia, D. K. Singh, “A comprehensive survey on techniques to handle face identity threats: challenges and opportunities”, Multimed Tools Appl, 82, pp. 1669–1748, 2023. https://doi.org/10.1007/s11042-022-13248-6

[5] J. Yang, D. Zhang, Y. Xu, and J. Y. Yang, “Recognize color face images using complex Eigenfaces”, Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 3832 LNCS, pp. 64–68, 2006, doi: 10.1007/11608288_9.

[6] A. W. Yip and P. Sinha, “Contribution of color to face recognition”, Perception, vol. 31, no. 8, pp. 995–1003, 2002, doi: 10.1068/p3376.

[7] Q. B. Sun, W. M. Huang, and J. K. Wu, “Face detection based on color and local symmetry information”, Proc. - 3rd IEEE Int. Conf. Autom. Face Gesture Recognition (FG1998), pp. 130–135, 1998, doi: 10.1109/AFGR.1998.670937.

[8] D. H. Brainard and B. A. Wandell, “Asymmetric color matching: how color appearance depends on the illuminant”, J. Opt. Soc. Am. A, vol. 9, no. 9, p. 1433, 1992, doi: 10.1364/josaa.9.001433.

[9] M. Pedersen and J. Y. Hardeberg, “Full-Reference Image Quality Metrics: Classification and Evaluation”, Foundations and Trends in Computer Graphics and Vision, vol. 7, no. 1, pp. 1–80, 2012, doi: 10.1561/0600000037.

[10] L. Liu, B. Liu, H. Huang and A. C. Bovik, “No-reference image quality assessment based on spatial and spectral entropies”, Signal Process. Image Commun., vol. 29, no. 8, pp. 856–863, 2014, doi: 10.1016/j.image.2014.06.006.

[11] I. Chingovska, A. Anjos, S. Marcel, "On the Effectiveness of Local Binary Patterns in Face Anti-spoofing"; IEEE BIOSIG, 2012, https://ieeexplore.ieee.org/document/6313548

[12] Facial Images: Faces94, Computer Vision Science Research Projects website, Designed and maintained by Dr Libor Spacek on 13th June 2009, https://cmp.felk.cvut.cz/~spacelib/faces/faces94.html

[13] Color FERET Database, National Institute of Standards and Technology (NIST) website, Designed by P. Jonathon Phillips, Created January 31, 2011, Updated December 3, 2019, https://www.nist.gov/itl/products-and-services/color-feret-database

[14] N. Khediri, M. Ammar and M. Kherallah, “Comparison of Image Segmentation using Different Color Spaces”. 2021 IEEE 21st International Conference on Communication Technology (ICCT), pp. 1188-1192, 2021, doi: 10.1109/ICCT52962.2021.9658094.

[15] S. Banerji, A. Verma, and C. Liu, “Novel color LBP descriptors for scene and image texture classification”, Proc. 2011 Int. Conf. Image Process. Comput. Vision, Pattern Recognition, IPCV 2011, vol. 2, pp. 537–543, 2011.

[16] D.G. Lowe, “Distinctive Image Features from Scale-Invariant Keypoints”. International Journal of Computer Vision, vol. 60, pp. 91–110, 2004, https://doi.org/10.1023/B:VISI.0000029664.99615.94

[17] H. Bay, A. Ess, T. Tuytelaars, L.V. Gool, "SURF: Speeded Up Robust Features", Computer Vision and Image Understanding (CVIU), vol. 110, no. 3, pp. 346--359, 2008.

[18] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, “ORB: An efficient alternative to SIFT or SURF”, Proc. IEEE Int. Conf. Comput. Vis., pp. 2564–2571, 2011, doi: 10.1109/ICCV.2011.6126544.

[19] S. Kakarla, P. Gangula, M. S. Rahul, C. S. C. Singh, and T. H. Sarma, "Smart Attendance Management System Based on Face Recognition Using CNN", 2020 IEEE-HYDCON, Hyderabad, India, 2020, pp. 1-5, doi: 10.1109/HYDCON48903.2020.9242847.

[20] A. Afaneh, F. Noroozi and Ö. Toygar, “Recognition of Identical Twins Using Fusion of Various Facial Feature Extractors”, EURASIP Journal on Image and Video Processing, vol. 2017:81, pp.1-14, Dec. 2017.

[21] J. Galbally, S. Marcel, and J. Fierrez, "Image Quality Assessment for Fake Biometric Detection: Application to Iris, Fingerprint and Face Recognition", IEEE Transactions on Image Processing, vol. 23, no. 2, pp. 710.724, Feb. 2014, doi: 10.1109/TIP.2013.2292332.
[22] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou, "A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects," in IEEE Transactions on Neural Networks and Learning Systems, vol. 33, no. 12, pp. 6999-7019, Dec. 2022, doi: 10.1109/TNNLS.2021.3084827.
[23] T.K. Ho, "Nearest Neighbors in Random Subspaces", In: A. Amin, D. Dori, P. Pudil, H. Freeman (eds), Lecture Notes in Computer Science, Springer, Germany, pp 640–648, 1998.
[24] Z. Yu et al., "Searching central difference convolutional networks for face anti-spoofing", Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., pp. 5294–5304, 2020, doi: 10.1109/CVPR42600.2020.00534.
[25] A. Benlamoudi et al., "Face Presentation Attack Detection Using Deep Background Subtraction", Sensors, vol. 22, no. 10, 2022, doi: 10.3390/s22103760.
[26] S. Karanwal and M. Diwakar, "Two novel color local descriptors for face recognition", Optik (Stuttg)., vol. 226, 2021, doi: 10.1016/j.ijleo.2020.166007.
[27] P. Terhörst, M. Huber, N. Damer, F. Kirchbuchner, and A. Kuijper, "Unsupervised Enhancement of Soft-biometric Privacy with Negative Face Recognition", arXiv:2002.09181v1 [cs.CV], 2020, [Online]. Available: http://arxiv.org/abs/2002.09181.
[28] B. Zhang, B. Tondi, and M. Barni, "Adversarial examples for replay attacks against CNN-based face recognition with anti-spoofing capability", Comput. Vis. Image Underst., vol. 197–198, 2020, doi: 10.1016/j.cviu.2020.102988.
[29] Z. Boulkenafet, J. Komulainen, L. Li, X. Feng, and A. Hadid, "OULU-NPU: A Mobile Face Presentation Attack Database with Real-World Variations", 2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017), Washington, DC, USA, 2017, pp. 612-618, doi: 10.1109/FG.2017.77.
[30] A. Vinay, A. Gupta, A. Bharadwaj, A. Srinivasan, K. N. B. Murthy, and S. Natarajan, "Deep Learning on Binary Patterns for Face Recognition", Procedia Comput. Sci., vol. 132, pp. 76–83, 2018, doi: 10.1016/j.procs.2018.05.164.
[31] L. F. Chen, H. Y. M. Liao, M. T. Ko, J. C. Lin, and G. J. Yu, "A New LDA-based Face Recognition System Which Can Solve the Small Sample Size Problem", Proc. Jt. Conf. Inf. Sci., vol. 4, pp. 282–286, 1998.
[32] J. Sikder, R. Chakma, R. J. Chakma, and U. K. Das, "Intelligent Face Detection and Recognition System", 2021 Int. Conf. Intell. Technol. CONIT 2021, 2021, doi: 10.1109/CONIT51480.2021.9498291.
[33] S. Karanwal, "A comparative study of 14 state of art descriptors for face recognition", Multimed. Tools Appl., vol. 80, no. 8, pp. 12195–12234, 2021, doi: 10.1007/s11042-020-09833-2.
[34] C.-K. Tran et al., "Local intensity area descriptor for facial recognition in ideal and noise conditions", J. Electron. Imaging, vol. 26, no. 2, p. 023011, 2017, doi: 10.1117/1.jei.26.2.023011.
[35] H. R. Chou, J. H. Lee, Y. M. Chan, and C. S. Chen, "Data-Specific Adaptive Threshold for Face Recognition and Authentication", Proc. - 2nd Int. Conf. Multimed. Inf. Process. Retrieval, MIPR 2019, pp. 153–156, 2019, doi: 10.1109/MIPR.2019.00034.
[36] W. Zhao, A. Krishnaswamy, R. Chellappa, D. L. Swets, and J. Weng, "Discriminant Analysis of Principal Components for Face Recognition", In: Wechsler, H., Phillips, P.J., Bruce, V., Soulié, F.F., Huang, T.S. (eds) Face Recognition. NATO ASI Series, vol 163. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-72201-1_4
[37] C. Geng and X. Jiang, "Face recognition using SIFT features", Proc. - Int. Conf. Image Process. ICIP, pp. 3313–3316, 2009, doi: 10.1109/ICIP.2009.5413956.
[38] G. Du, F. Su, and A. Cai, "Face recognition using SURF features", MIPPR 2009 Pattern Recognit. Comput. Vis., vol. 7496, p. 749628, 2009, doi: 10.1117/12.832636.

Acknowledgments
The authors would like to thank Idiap Research Institute in Switzerland for providing the Replay Attack database; Dr. Libor Spacek for providing the Faces94 database; and Dr. P. Jonathon Phillips from the National Institute of Standards and Technology for providing the ColorFERET database.

Authors Contributions
Mohammad Mehdi PAZOUKI: Performed experimental analysis and wrote the paper.
Önsen TOYGAR: Developed and designed the analysis and reviewed the paper.
Mahdi HOSSEINZADEH: Collected and prepared data and wrote the paper.

Conflicts of Interest
The authors declare no conflict of interest.

Ethical Approval and Informed Consent
It is declared that during the preparation process of this study, scientific and ethical principles were followed, and all the studies benefited from are stated in the bibliography.

Availability of data and material

Not applicable.

Plagiarism Statement
This article has been scanned by iThenticate ™.

SAKARYA UNIVERSITY JOURNAL OF COMPUTER AND INFORMATION SCIENCES
http://saucis.sakarya.edu.tr/
e-ISSN: 2636-8129
Publisher: Sakarya University
Vol. 7, No. 3, 2024, 378-388
DOI: 10.35377/saucis...1509150
SAKARYA UNIVERSITY
RESEARCH ARTICLE

Extraction of Cattle Retinal Vascular Patterns with Different Segmentation Methods

Pınar Cihan[1], Ahmet Saygılı[1*], Muhammed Akyüzlü[1], Nihat Eren Özmen[1], Celal Şahin Ermutlu[2], Uğur Aydın[2], Alican Yılmaz[2], Özgür Aksoy[2]

[1]Tekirdağ Namık Kemal University, Çorlu Engineering Faculty, Department of Computer Engineering, Tekirdağ, Türkiye
[2]Kafkas University, Faculty of Veterinary Medicine, Department of Surgery, Kars, Türkiye

Corresponding author:
Ahmet Saygılı, Tekirdağ Namık Kemal University, Çorlu Engineering Faculty, Department of Computer Engineering, Tekirdağ, Türkiye,
asaygili@nku.edu.tr

Article History:
Received: 2.07.2024
Accepted: 04.11.2024
Published Online: 29.11.2024

ABSTRACT
In the field of animal husbandry, the process of animal identification and recognition is challenging, time-consuming, and costly. In Türkiye, the ear tagging method is widely used for animal identification. However, this traditional method has many significant disadvantages such as lost tags, the ability to copy and replicate tags, and negative impacts on animal welfare. Therefore, in some countries, biometric identification methods are being developed and used as alternatives to overcome the disadvantages of traditional methods. Retina vessel patterns are a biometric identifier with potential in biometric identification studies. Preprocessing steps and vessel segmentation emerge as crucial steps in image processing-based identification and recognition systems. In this study, conducted in the Kars region of Türkiye, a series of preprocessing steps were applied to retinal images collected from cattle. Fuzzy c-means, k-means, and level-set methods were utilized for vessel segmentation. The segmented vascular structures obtained with these methods were comparatively analyzed. As a result of the comparison, it was observed that all models successfully performed retinal main vessel structure segmentation, fine vessels were successfully identified with fuzzy c-means, and spots in retinal images were detected only by the level-set method. Evaluating the success of these methods in identification, recognition, or disease detection will facilitate the development of successful systems.

Keywords: Animal retina segmentation, CLAHE, Fuzzy c-means, K-means, Level-set

1. Introduction

Segmentation and classification of retinal blood vessels in medical imaging have been extensively utilized in various fields of research [1]. Retinal blood vessels offer valuable insights for disease detection and identification studies [2]. By detecting diseases, it becomes possible to initiate the treatment process and facilitate the identification of existing ailments in animals. Moreover, early diagnosis allows for prompt intervention before the disease progresses further. Animal identification methods have yielded successful results in areas such as traceability and fraud prevention [3]. Traditional methods like earring, marking, and forging, which have been employed for cattle identification over the years, are associated with security vulnerabilities, susceptibility to fraudulent practices, and a negative impact on animal welfare, making them less preferred today. Advancements in technology have led to the increased adoption of identification processes utilizing biometric features, which are gaining popularity day by day.

The production and trade of animals and animal products are witnessing a steady increase, making it challenging to effectively monitor and track animals and their associated products. Animal tracking is particularly vital for ensuring safety and preventing the spread of diseases. Transparent and accurate animal identification processes must be implemented throughout an animal's lifespan to facilitate effective tracking. Biometric-based identification processes have gained prominence due to their superior success rates and enhanced security measures. The field of animal identification is witnessing a growing number of research studies as time progresses. Figure 1 illustrates the upward trend in the quantity of studies conducted in this domain across various years [4].

Cite as: P. Cihan, A. Saygılı, M. Akyüzlü, N. E. Özmen, C. Ş. Ermutlu, U. Aydın, A. Yılmaz, Ö. Aksoy "Extraction of Cattle Retinal Vascular Patterns with Different Segmentation Methods," *Sakarya University Journal of Computer and Information Sciences*, vol.7, no. 3, pp. 378-388, 2024. Doi: 10.35377/saucis…1509150

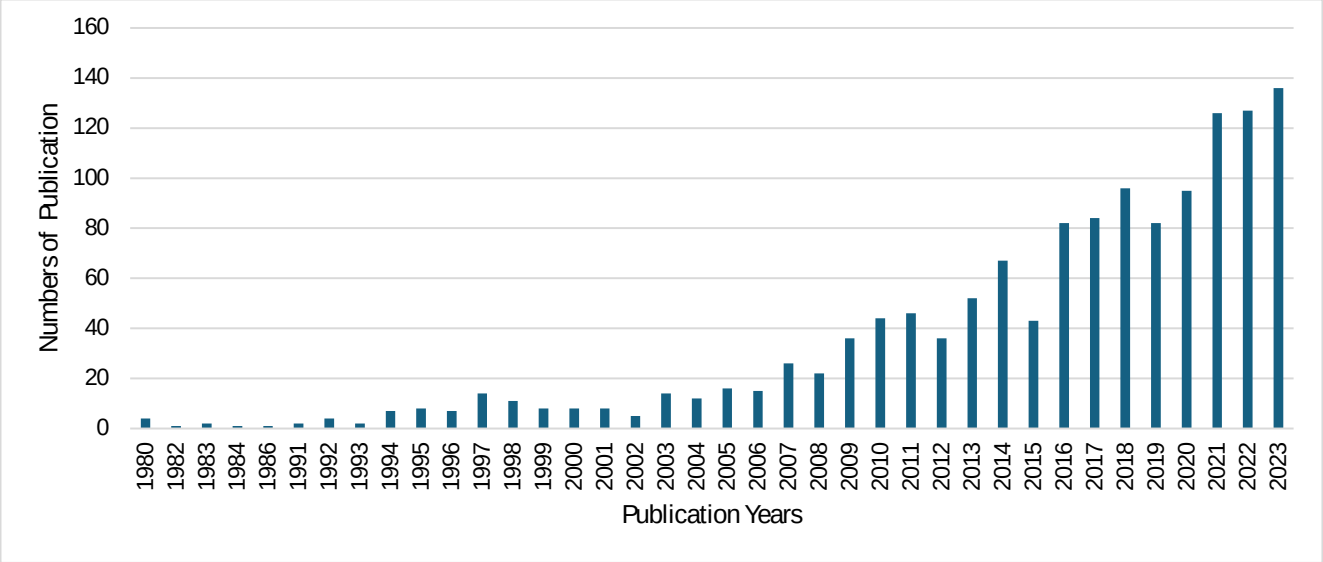


Figure 1. Representation of the Number of Studies Carried Out in the Field of Animal Identification by Years (1991-2023)

Biometric identification serves as the foundation of an individual's physical identity. Biometric features typically remain constant unless altered through medical procedures or injury, and they are inherently difficult to replicate exactly [5]. These methods, widely employed in human identification, are increasingly finding applications in animal identification as well. Biometric features in animals encompass characteristics such as retinal vessel patterns, nose prints, and iris patterns. During the selection process of a biometric feature to be implemented, it is essential to assess it against various criteria and make a decision based on the evaluation results. Here are the key points outlining the selection process for a biometric feature.

- Universality: The chosen biometric identifier should apply to all animals without compromising their confidentiality or health.
- Uniqueness: The biometric identifier must possess distinctive features that enable differentiation between two animals with the same characteristics.
- Performance: The identification criteria, such as accuracy, speed, and robustness, should be achievable. The required resources for attaining acceptable identification performance should also be manageable.
- Circumvention: The system's response to fraudulent techniques should be considered. The selected biometric identifier and the associated application should meet the requirements in terms of preventing fraudulent activities [6].

In this research, retinal vessel patterns, a biometric feature, were employed. Retinal vessel patterns exhibit a remarkable level of uniqueness and distinctiveness in both livestock and humans, allowing differentiation even among monozygotic (identical) twins [7]. Similar to medical image analysis, this study incorporated various preprocessing steps to analyze the retinal image. Subsequently, segmentation techniques were applied to achieve improved outcomes in the analysis.

Image segmentation plays a pivotal and indispensable role in the realm of medical image analysis, serving as a fundamental technique. Segmentation involves dividing an image into distinct parts or sections. The primary aim of image segmentation is to simplify the information in an image and make it readily analyzable and comprehensible. [8]. This simplification aids in analysis by assigning labels to individual pixels, enabling the identification and localization of objects within an image. Once the segmentation process is completed, the output consists of a collection of segments representing the entire image [9].

Extensive research has been conducted in the literature on the extraction of retinal blood vessels in both human and animal subjects. This section focuses on reviewing studies specifically related to the extraction of blood vessels in animals.

Mustafi et al. conducted a study introducing a system named RetIS for identification using goat retina images. The application of the suggested system involved the utilization of the Contrast Limited Adaptive Histogram Equalization (CLAHE) technique for image enhancement. The segmentation process employed the active contour algorithm, specifically the snake algorithm. Following segmentation, matching was performed utilizing the Hamming distance as a similarity measure [10].

The study focused on analyzing the temporal changes in retinal vessels using images obtained from dogs. Retinal images captured at specific time intervals were utilized to observe these changes. Two programs were employed to examine the variations in the vessels. The first program utilized the intersection points of two concentric circles, where a line was drawn from the midpoint of the retina, to observe temporal changes over time. In the second program, three major vessels were selected, and the changes in the angles between them were observed in subsequent images [11].

Several studies [1,12-14] have investigated retinal images of cattle and sheep. In all of these studies, the OptiReader device, patented in the United States in 2004, was employed to capture retinal images. This device enables automatic acquisition and storage of the vascular structure from retinal images, providing researchers with a convenient tool for their analysis.

In this study, after applying preprocessing steps to the collected cattle retinal images, three different methods were used to segment vessel patterns. These methods are Fuzzy c-means, K-means, and Level-set methods. The results of vessel segmentation using these methods were compared comparatively. It is noteworthy that detailed information about preprocessing and segmentation stages is not provided in retinal identification/recognition studies in the literature. Therefore, this study fills this gap by shedding comprehensive light on the preprocessing and segmentation steps, providing valuable information to researchers in this field.

2. Materials and Method

The retinal images used in the study were collected from cattle in the Kars region of Türkiye. Optomed Smartscope digital fundus camera was used to capture the retinal images. A total of 2430 retinal images were collected from the right and left eyes of 300 cattle in the study. In this study, retinal images collected from cattle were shared with the public via Kaggle (https://www.kaggle.com/datasets/animalbiometry/cattle-retinal-fundus-images). However, not all collected images were of sufficient quality to be used in image processing, so some were removed from the dataset. The images were removed from the dataset due to low quality, distortion caused by external factors such as lighting, and the inability to capture the retina clearly during the imaging process. The insufficient number of images (Animals with less than 2 retinal images were excluded from the study) obtained from the animals is also one of the reasons for the exclusion of the images. Sample retinal images that are of such poor quality that they cannot be used in the study are presented in Figure 2.

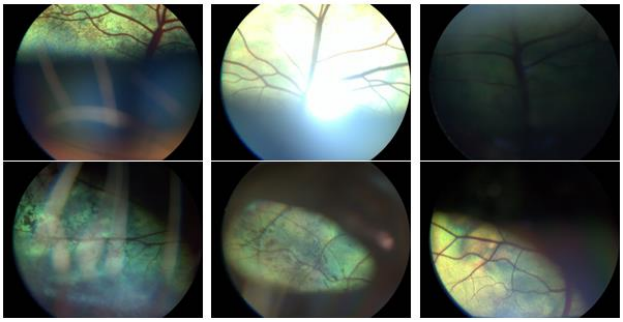


Figure 2. Sample Retina Images Removed from Dataset

As a result, a total of 1206 images belonging to 234 animals were used in the study. Figure 3 shows representative examples of high-quality retinal images used in this dataset. Notably, the images are characterized by a resolution of 1536x1152 pixels and are stored in the JPG format, a commonly employed file format for image compression and storage.

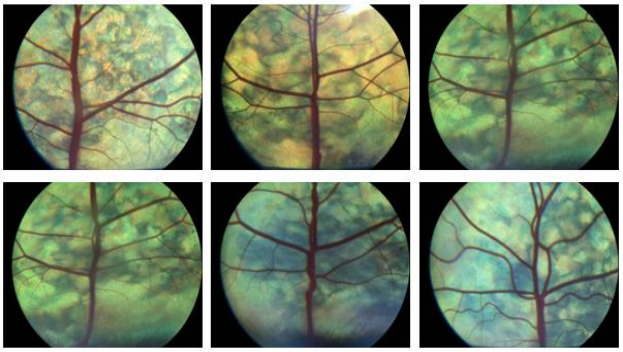


Figure 3. Sample Retina Images from Used Dataset

2.1. Preprocessing Step

Preprocessing steps play a critical role in achieving successful results in image processing methods. These steps help to improve the performance of the model by cleaning and optimizing the raw data. Operations such as reducing noise in images, improving contrast, and highlighting edges provide results with higher accuracy. In addition, these steps reduce the complexity of the model, providing a faster and more efficient processing process. For example, techniques such as resizing, graying, or normalizing images allow algorithms to work more effectively on the image. Therefore, accurate and effective preprocessing steps are a critical requirement in a successful image processing project.
Within our study, we organized the image preprocessing stage according to the flowchart depicted in Figure 9. This crucial phase encompassed several discrete procedures, including channel separation, extraction of the green channel, grayscale transformations, contrast limited adaptive histogram equalization (CLAHE), morphological operations, image extraction, and

noise filtering. For visual clarity, we handpicked three distinct images from our dataset to serve as illustrative examples. These selected images were subjected to the designated preprocessing steps, and the outcomes were meticulously recorded. To streamline computational efficiency while preserving image fidelity, we resized all images to a standardized resolution of 480x480 pixels before initiating the chosen preprocessing steps. By incorporating this resizing operation into our workflow, we were able to expedite the overall processing time while maintaining the integrity of the images.

2.1.1. Green Channel Extraction

As a pre-processing step to obtain retinal vessel patterns, one of the procedures involved separating the retinal image into its constituent color channels and subsequently extracting the green channel. The original image consisted of three color channels: red, green, and blue. By performing green channel subtraction, we focused on extracting the green channel from the image. This approach was chosen due to the observation that blood vessels exhibit higher contrast in the green channel compared to the other two channels. Additionally, the green channel offered a balanced representation without being under-illuminated or oversaturated, as opposed to the red and blue channels [15]. Figure 4 presents the resulting images after the extraction of the green channel.

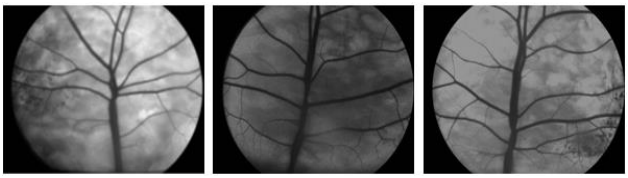


Figure 4. Three Different Sample Images Results of Applying Green Channel Extraction

2.1.2. Grayscale Conversions

Grayscale transformation is a commonly employed preprocessing step in various image processing techniques. Its primary purpose is to simplify algorithms and alleviate computational complexities. By converting an image to grayscale, the resulting representation offers easier visualization and interpretation compared to the original color image. Grayscale transformation effectively reduces the complexity associated with color information, allowing algorithms to focus on essential image features and patterns. This simplification facilitates the subsequent stages of image analysis and enhances the efficiency of processing algorithms.

2.1.3. Contrast Limited Adaptive Histogram Equalization (CLAHE)

Contrast Limited Adaptive Histogram Equalization (CLAHE) is a technique commonly employed to address the low contrast issues encountered in digital images, particularly in medical imaging [16]. In the context of medical imaging, CLAHE has demonstrated superior performance compared to other histogram equalization methods such as Adaptive Histogram Equalization (AHE) and standard Histogram Equalization (HE). CLAHE enhances contrast by imposing limitations on the contrast enhancement process, thereby mitigating the generation of unwanted noise. CLAHE achieves contrast limitation by defining a clip limit, which determines the maximum height of the histogram. This clip limit is adjusted based on the desired contrast requirements. By applying CLAHE following the gray-level transformations, the retinal vessels in the images become more discernible from the background. This enhancement, in contrast, enables improved visibility and differentiation of the vessels. Figure 5 showcases sample images where the CLAHE method has been applied, highlighting the effectiveness of this technique in enhancing image quality and vessel visibility [17].

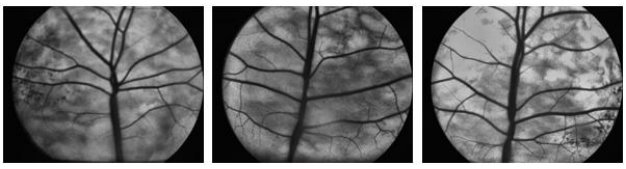


Figure 5. Three Different Sample Images with CLAHE Applied

2.1.4. Morphological Operations

When working with binary images, morphological operations are commonly employed to differentiate and extract specific components of interest [18]. Two primary processes involved in morphological operations are erosion and dilation. Erosion diminishes the size of objects present in the image, while dilation expands the area occupied by these objects. By combining these two operations, more intricate filtering operations can be achieved.

The process of dilation followed by erosion is known as opening, while erosion followed by dilation is referred to as closing. Opening operation smoothens object contours and eliminates thin protrusions. It has a similar contour-smoothing effect as closing, but it is particularly effective in removing small holes and filling gaps in the image [19]. These specific morphological operations, referred to as opening and closing, offer valuable tools for refining binary images, enhancing object shapes, and improving the overall quality of segmentation results.

To discern retinal vessels, morphological opening and closing operations were employed. These procedures aid in refining the segmentation of vessels in binary images. Figure 6 visually presents sample images where morphological opening and closing have been applied.

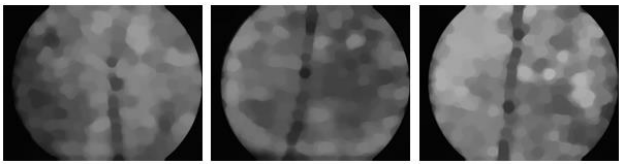


Figure 6. Example Images with Morphological Processes Applied

2.1.5. Image Subtraction

Image subtraction is a technique employed to isolate the objects of interest within an image from the background. In our study, an image extraction process was implemented to separate the vascular patterns from the surrounding tissues. Following the application of morphological operations, images enhanced with CLAHE were obtained. Subsequently, the image extraction procedure was executed to isolate the desired vascular patterns.

Figure 7 showcases sample images where the image extraction process has been successfully applied. These images demonstrate the effectiveness of the extraction method in isolating and highlighting the vascular patterns, thus facilitating further analysis and examination of the retinal vessels in the presence of minimal background interference.

Figure 7. Three Different Sample Images with Image Subtraction Applied

2.1.6. Noise Filtering

During image processing, filtering operations are commonly employed to yield valuable results. These operations serve to eliminate noise from images while preserving important details. In our study, filtering operations were performed to enhance the quality of the retinal vessel patterns.

Initially, the bitwise_not operation was applied, causing the image to be inverted at the bit level. This operation ensured that the retinal vessel patterns appeared as black, while the background was represented as white. This inversion facilitated subsequent analysis and visualization of the vessels against a clear background.

To further improve the image quality, the application of a median filter helped in noise elimination. The median filter effectively reduces noise by replacing each pixel's value with the median value within a defined neighborhood. This operation helps to maintain the important features and details of the vessel patterns while reducing unwanted noise.

Figure 8 depicts sample images where the noise filtering process has been successfully applied. These images demonstrate the efficacy of the filtering techniques in enhancing clarity and removing unwanted noise, allowing for more accurate analysis of the retinal vessel patterns.

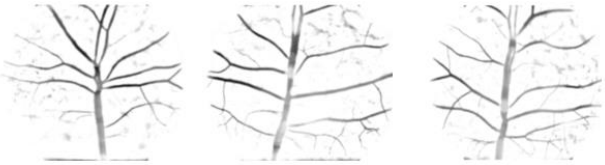


Figure 8. Three Different Sample Images with Noise Filtering Applied

2.2. Segmentation Step

In this section of the study, three distinct methods for segmentation are discussed: Level-set, Fuzzy c-means, and k-means methods. Each of these techniques offers a unique approach to segmenting retinal images. By applying these three segmentation methods to the selected sample images, the results are presented, showcasing the effectiveness of each approach in delineating the retinal vessels from the background.

2.2.1. Fuzzy C-means Method

The fuzzy c-means method is an unsupervised technique widely applied in various fields, including medical imaging, target recognition, and shape analysis [20]. In the domain of retinal image analysis, numerous studies have explored the segmentation of retinal images and the detection of blood vessels using the fuzzy c-means method [21-24].

Clustering is a commonly utilized approach for image segmentation, which aims to categorize patterns so that samples within the same group exhibit greater similarity compared to those belonging to different groups. The fuzzy c-means method represents one of the fuzzy clustering techniques that retain more information compared to traditional fixed clustering methods. This algorithm provides additional flexibility by allowing pixels to belong to multiple classes with varying membership degrees.

The objective of the fuzzy c-means algorithm is to partition a given set of pixels into a collection of "C" fuzzy sets based on a specific criterion. This algorithm operates by minimizing the objective function as depicted in Equation 1 [25]. The fuzzy c-means method leverages the mathematical optimization process to iteratively refine the assignment of pixels to different classes, resulting in the segmentation of the retinal image into distinct regions with varying degrees of membership to each class.

$$J_m = \sum_{i=1}^{N} \sum u_{i,j}^{m} \left| x_i - c_j \right| \left\| x_i - c_j \right|^2 \quad 1 \leq m < \infty \tag{1}$$

N represents the total pixel count, m is a real number greater than 1, $u_{i,j}$ is the membership degree of x_i in cluster j, x_i is the ith pixel in the cluster, the cluster's center is represented by c_j, while $\|*\|$ denoting the similarity between the center and the measured data. After taking the derivative of Equation 2, the equation obtained by using the Lagrange method is equal to zero, and Equation 3 is obtained. The M value indicates the coefficient determined according to the characteristics of the image. As the M value approaches infinity, the blurring increases and reaches full turbidity.

$$u_{ij} = \frac{1}{\sum_{k=1}^{c} \left(\frac{\left\| x_i - c_j \right\|}{\left\| x_i - c_k \right\|} \right)^{\frac{2}{m-1}}} \tag{2}$$

$$c_j = \frac{\sum_{i=1}^{N} u_{i.j}^{m} X_i}{\sum_{i=1}^{N} u_{ij}^{m}} \tag{3}$$

2.2.2. K-means Method

The k-means method is a straightforward unsupervised technique commonly used in clustering problems. This method enables the classification of a given dataset into a predetermined number of clusters [26]. In the context of retinal image segmentation and blood vessel detection, the k-means method has been widely applied in numerous studies [27-28-29-30].

The k-means algorithm follows a set of steps:

1. The initial step involves defining the center points for each of the k clusters. It is crucial to position the center points as far apart from each other as possible.

2. Each data point's distance to the cluster centers is computed based on the dataset. According to this distance metric, each data point is assigned to the cluster that is closest to it.

3. After assigning data points to clusters, the average values of the cluster centers are recalculated, and the second step is repeated for all data points.

4. The iterations described in steps 2 and 3 are repeated until a specified number of iterations or a predetermined threshold level is reached.

Finally, this method aims to minimize an objective function. The objective function is shown in Equation 4. c_k denotes the object in the C cluster, y_i the center point of the cluster, K the number of clusters, S is the clustering partition of the set of entities, and the expression $\|.\|$ the distance metric.

$$W(S, C) = \sum_{k=1}^{K} \sum_{i \in S_k} \|y_i - c_k\|^2 \tag{4}$$

2.2.3. Level-Set Method

The level-set method, initially introduced by Osher and Sethian, is founded on the formulation of an appropriate equation of motion for evolving a boundary based on curvature effects [31]. While originally developed for solving physics and fluid dynamics problems, the level-set method has found substantial utility in the field of image processing in recent years.

The fundamental principle of the level-set method is to iteratively determine the boundaries of objects within an image gradually. At each stage of the method, contour curves are aligned with the object boundaries and adjusted accordingly. The objective is to minimize discrepancies between the contour curves and the actual object boundaries by employing smoothing techniques.

By gradually evolving the contour curves and minimizing differences between the curves and the object boundaries, the level-set method ensures the identification of the most accurate and appropriate object boundaries. This progressive refinement process facilitates robust and accurate segmentation results, allowing for precise analysis and interpretation of objects within the image.

Zero set of the function ϕ, active contour

$$C = \{(x, y) \mid \phi(x, y) = 0\} \quad (5)$$

is used to represent the contour and points inside/outside the contour are indicated as greater or less than ϕ. In Figure 9, the variation of the contour curves with the level-set method is given.

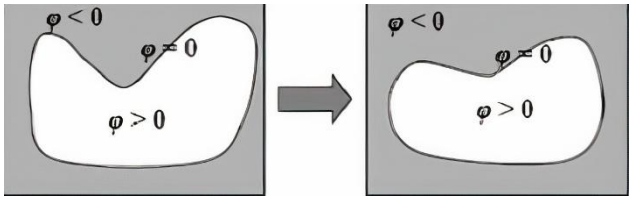


Figure 9. Contour curves change and representation of ϕ values [32]

3. Experimental Results and Discussions

To provide a comprehensive overview of the experimental workflow, Figure 10 depicts a flow chart diagram showcasing the sequence of preprocessing steps, segmentation methods, and subsequent operations applied to the images. This diagram serves as a visual representation of the entire process, illustrating the order and interconnectedness of the implemented techniques. A study was executed on a computer system equipped with an Intel(R) Core(TM) i7-11800H processor and 16 gigabytes of RAM. The coding phase was carried out using the Python programming language and the Anaconda Navigator platform.

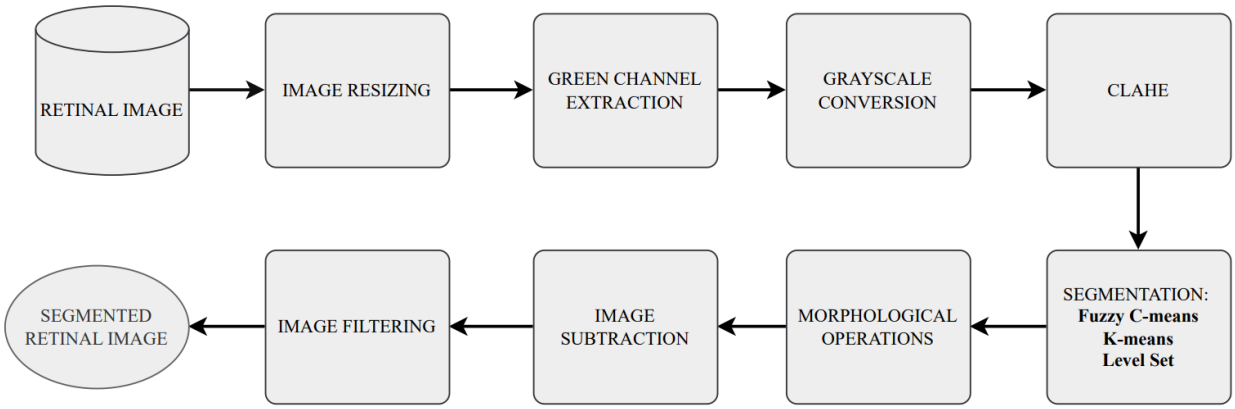


Figure 10. Flow Chart of This Study

In the conducted study, a series of preprocessing steps and distinct segmentation methods were applied to the acquired images. Table 1 presents the parameter names along with their respective values used in the employed methods. These parameters play a crucial role in determining the specific characteristics and behavior of each method during the segmentation process.

Table 1. Parameters and Values of the Methods

Method	Parameter	Value
CLAHE	clipLimit, tileGridSize	2.0, (9,9)
medianBlur	ksize	5
Morphological operations	kernel_size	(2,2) / (7,7) / (2,2)
fuzzy_c_mean	clusters, max_iter	2 / 4
k_mean	k, max_iter	2 / 4
level_set	mu, max_iter	0.1 / 200

The CLAHE method in this study employed two parameters: "clipLimit" and "tileGridSize". The "clipLimit" parameter defined the maximum allowed contrast enhancement, while "tileGridSize" determined the number of cells in the grid used for contrast equalization. The "ksize" parameter in the medianBlur method represented the size of the kernel used for blurring or smoothing the image. For morphological operations, the "kernel_size" parameter indicates the selected size of the kernel employed in these operations. In the fuzzy_c_mean method, the parameter "clusters" denoted the desired number of clusters, while "max_iter" represented the maximum number of iterations to be performed during the algorithm's execution. In the k_mean method, "k" refers to the number of clusters to be generated, and "max_iter" represents the maximum number of iterations to be executed. In the level_set method, "mu" denoted the determination rate of the contours, influencing their evolution, and "max_iter" parameter sets the upper limit for the number of iterations performed by the level-set algorithm.

The segmentation results obtained are given in Figure 11. In Figure 11, the segmentation results created by the original image, fuzzy c-means, k-means, and level-set methods are given on the same line. In different rows are the results of images taken from different animals.

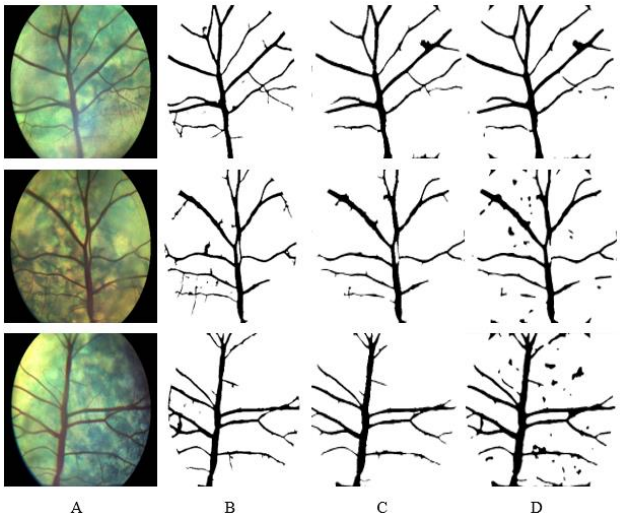
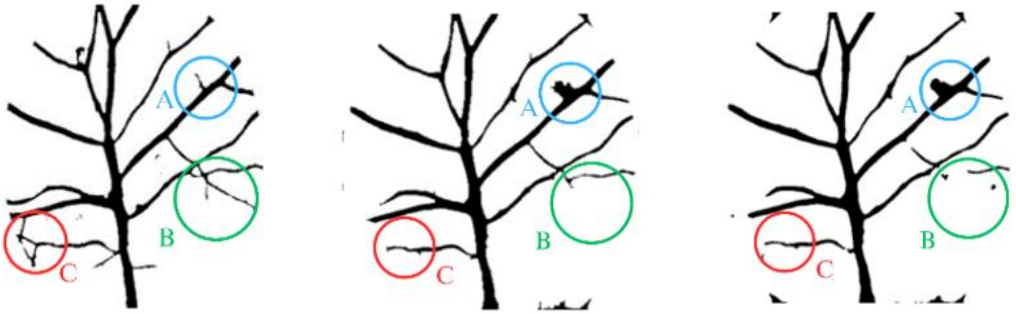


Figure 11. Retinal images for (A) Original and Segmented from (B) Fuzzy c-means, (C) K-means, and (D) Level-set method

Differences between segmented images obtained with three different methods (Fuzzy c-means, K-means, and Level-set) have been identified in the study. Marking these differences will help the segmentation results to be interpreted more successfully. In Figure 12, there are marked representations of some differences in the images obtained as a result of the methods. In the Figure, parts belonging to the same region are encoded with the same-colored circle.

Sample 1



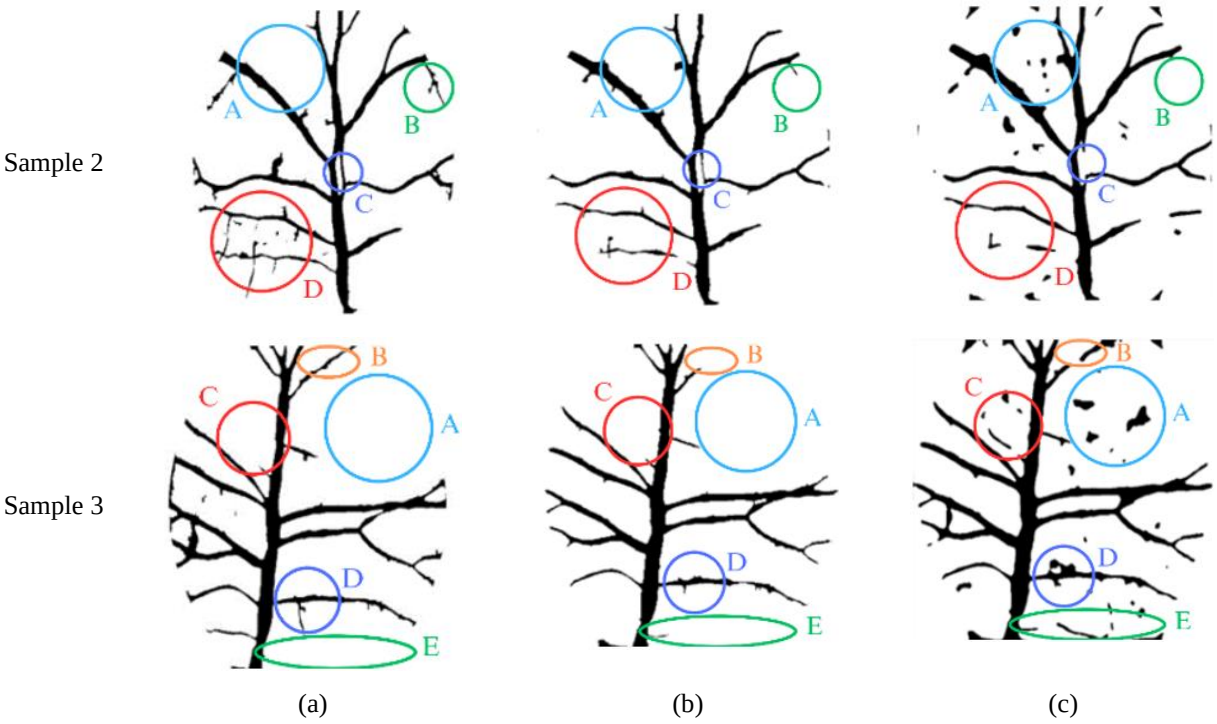


Figure 12. Comparison of Segmentation Results for (a) Fuzzy c-means, (b) K-means, (c) Level-set Method

Figure 12 - Sample 1: When examining the vessel regions marked with three different colors, it is observed that fine vessels are better detected with the fuzzy c-means method in region A enclosed by the blue circle, while it is observed that the k-means and level-set methods could not fully capture the structure of fine vessels. Similarly, when examining the vessels marked as B (green circle) and C (red circle), it is seen that fine vessels are better detected with fuzzy c-means, while other methods could not fully extract these vessels.

Figure 12 - Sample 2: In this retinal image, four regions have been marked; it is observed that the level-set method detects the spots in the background in region A. Upon examination of the original image in Figure 10, it appears that there are spots in the background of the retinal image. It is important to segment these spots if they are associated with a disease. When areas B, C, and D are examined, it is again observed that fine vessels are more clearly obtained by the fuzzy c-means method.

Figure 12 - Sample 3: In the last example, five different areas have been marked with circles on the retinal image. Upon examination of the marked areas, it is evident that the level-set method successfully segmented the background spots in areas A, C, and E. While the entire fine vessel in region B was obtained using the fuzzy c-means method, only a part of the vessel could be extracted with the k-means and level-set methods. In area D, the fine vessel was well detected with the fuzzy c-means method; however, it is observed that the spots were more effectively removed with the level-set method.

As a result, although there is no significant difference in main vessel segmentation among the methods, fine vessels have been better extracted using the fuzzy c-means method, and the spots in retinal images have been detectable with the level-set method. It is also worth investigating which method could be more successful in identification, recognition, or disease diagnosis, and this should be evaluated as a research topic.

In existing literature studies with animal retinal images, the segmentation process is often performed automatically using specific pre-built software. However, in our study, the results obtained from various segmentation methods were compared and evaluated. This approach allowed for a more comprehensive analysis and assessment of the different techniques employed in the segmentation process. However, it is known that applications such as artificial intelligence and image processing are limited in the field of animal science [33, 34]. Nonetheless, there is a noticeable trend of increasing popularity in these areas with each passing day. In this study, fundamental information that will increase interdisciplinary studies has been presented to researchers.

4. Conclusion

In this study, the aim was to present researchers with a detailed exploration of preprocessing and segmentation steps for the development of identification, recognition, or disease diagnosis systems based on cattle retina patterns. Particularly, these processes are crucial for the transition from traditional animal identification methods, such as ear tagging, to biometric-based systems.

Cattle retina images collected from the Kars region of Türkiye were used in this study. In the preprocessing step, the green channel extraction method was applied to enhance the visibility of veins, followed by applying the grayscale transformation

to the image, addressing low contrast issues with CLAHE, applying morphological operations, obtaining clear vessel structures with image subtraction, and finally, applying noise filtering for noise removal and background elimination. Following preprocessing, the vascular patterns in these images were segmented using fuzzy c-means, k-means, and level-set methods. Upon comparative examination of the segmented vessels, it was observed that all models effectively segmented retinal main vascular structures, while the fuzzy c-means method was more successful in extracting fine vessels and the level-set method was more successful in removing speckles from retina images.

This study contributes to the existing literature by providing a detailed implementation of preprocessing steps for animal retina images and offering a comparative analysis of segmentation techniques. In contrast to many existing studies based on automatic software, our approach includes a systematic comparison and evaluation of various methods. In the future, the next step will be to evaluate the effectiveness of these methods in identification, recognition, and disease diagnosis. Additionally, by highlighting the importance of interdisciplinary research in the context of artificial intelligence applications in animal sciences, this study provides a valuable perspective for future endeavors in this developing field.

References

[1] A. Allen, B. Golden, M. Taylor, D. Patterson, D. Henriksen, R. Skuce, "Evaluation of retinal imaging technology for the biometric identification of bovine animals in Northern Ireland," Livestock Science, 116(1-3), 42-52,2008.

[2] E. L. Fletcher, A. I. Jobling, K. A. Vessey, C. Luu, R. H. Guymer, P. N. Baird, "Animal models of retinal disease," Progress in Molecular Biology and Translational Science, Elsevier, 2011, pp. 211-286.

[3] D. L. Pendell, G. W. Brester, T. C. Schroeder, K. C. Dhuyvetter, G. T. Tonsor, "Animal identification and tracing in the United States," American Journal of Agricultural Economics, 92(4), 927-940,2010.

[4] Animal Biometric. (2024). Result of animal biometric in web of science [Online]. Available: https://www.webofscience.com/wos/woscc/summary/fce1ad3f-45dd-49d2-864a-32a7eea75200-9145fcaa/relevance/1.

[5] B. Shavers, J. Bair, "Hiding behind the keyboard: uncovering covert communication methods with forensic analysis," Syngress, 2016.

[6] A. I. Awad, "From classical methods to animal biometrics: A review on cattle identification and tracking," Computers and Electronics in Agriculture, 123, 423-435, 2016.

[7] G. U. Barron, G. Corkery, B. Barry, F. Butler, K. McDonnell, S. Ward, "Assessment of retinal recognition technology as a biometric method for sheep identification," Computers and Electronics in Agriculture, 60(2), 156-166, 2008.

[8] H. D. Cheng, X. H. Jiang, Y. Sun, J. Wang," Color image segmentation: advances and prospects," Pattern Recognition, 34(12), 2259-2281, 2001.

[9] S. Wali, C. Li, M. Imran, A. Shakoor, A. Basit, "Level-set Evolution for Medical Image Segmentation with Alternating Direction Method of Multipliers," Signal Processing, 109105, 2023.

[10] S. Mustafi, P. Ghosh, S. N. Mandal, "RetIS: Unique identification system of goats through retinal analysis," Computers and Electronics in Agriculture, 185, 106127, 2021.

[11] J. R. Gionfriddo, A. C. Lee, T. A. Precht, C. C. Powell, K.K Marren, S. V. Radecki, "Evaluation of retinal images for identifying individual dogs," American Journal of Veterinary Research, 67(12), 2042-2045, 2006.

[12] B. Barry, G. U. Barron, F. Butler, S. Ward, K. McDonnell, "Verification of sheep identity by means of a retinal recognition system," Transactions of the ASABE. 2011; 54:1161-7.

[13] G. Alturk, F. Karakus, "Assessment of Retinal Recognition Technology as A Biometric Identification Method in Norduz Sheep," in International Animal Science Conference, Turkey, 2019, pp. 404.

[14] M. A. Rojas-Olivares, G. Caja, S. Carné, A. A. K. Salama, N. Adell, P. Puig, "Determining the optimal age for recording the retinal vascular pattern image of lambs," Journal of Animal Science, 90(3), 1040-1046,2012.

[15] V. V. Kumari, N. Suriyaharayananm, C. T. Saranya, "Feature extraction for early detection of diabetic retinopathy," in International Conference on Recent Trends in Information, Telecommunication and Computing, India, 2010, pp. 359-361.

[16] S. M. Pizer, E. P. Amburn, J. D. Austin, R. Cromartie, A. Geselowitz, T. Greer, B. H. Romeny, J. B. Zimmerman, K. Zuiderveld, "Adaptive histogram equalization and its variations," Computer Vision, Graphics, and Image Processing, 39(3), 355-368, 1987.

[17] S. Sahu, A. K. Singh, S. P. Ghrera, M. Elhoseny, "An approach for de-noising and contrast enhancement of retinal fundus image using CLAHE," Optics and Laser Technology, 110, 87-98,2019.

[18] M. L. Comer, E. J. Delp III, "Morphological operations for color image processing," Journal of Electronic Imaging, 8(3), 279-289,1999.

[19] J. E. Arco, J. M. Górriz, J. Ramírez, I. Álvarez, C. G. Puntonet, "Digital image analysis for automatic enumeration of malaria parasites using morphological operations," Expert Systems with Applications, 42(6), 3041-3047,2015.

[20] S. Ghosh, S. K. Dubey, "Comparative analysis of k-means and fuzzy c-means algorithms," International Journal of Advanced Computer Science and Applications, 4(4),2013.

[21] B. Sindhusaranya, M.R. Geetha, "Retinal blood vessel segmentation using root Guided decision tree assisted enhanced Fuzzy C-mean clustering for disease identification," Biomedical Signal Processing and Control, 82, 104525,2023.

[22] A. E. Hassanien, E, Emary, H. M. Zawbaa, "Retinal blood vessel localization approach based on bee colony swarm

optimization, fuzzy c-means and pattern search," Journal of Visual Communication and Image Representation, 31, 186-196,2015.
[23] Y. Kumar, B. Gupta, "Retinal image blood vessel classification using hybrid deep learning in cataract diseased fundus images," Biomedical Signal Processing and Control, 84, 104776,2023.
[24] B. D. Barkana, I. Saricicek, B. Yildirim, "Performance analysis of descriptive statistical features in retinal vessel segmentation via fuzzy logic, ANN, SVM, and classifier fusion," Knowledge-Based Systems, 118, 165-176,2017.
[25] K. Padmanaban, R. J. Kannan, "Localization of optic disc using Fuzzy C Means clustering," in International Conference on Current Trends in Engineering and Technology (ICCTET), India,2013, pp. 184-186.
[26] T. M. Kodinariya, P.R. Makwana, "Review on determining number of Cluster in K-Means Clustering," International Journal, 1(6), 90-95,2013.
[27] G. Sun, X. Liu, S. Wang, L. Gao, M. Liu, "Width measurement for pathological vessels in retinal images using centerline correction and k-means clustering," Measurement, 139, 185-195,2019.
[28] T. Zhou, S. Ruan, S. Canu, "A review: Deep learning for medical image segmentation using multi-modality fusion," Array, 3, 100004,2019.
[29] Y. Li, Q. Lao, Q. Kang, Z. Jiang, S. Du, S. Zhang, K. Li, "Self-supervised anomaly detection, staging and segmentation for retinal images," Medical Image Analysis, 87, 102805,2023.
[30] T. Nazir, A. Irtaza, A. Javed, H. Malik, D. Hussain, R. A. Naqvi, "Retinal image analysis for diabetes-based eye disease detection using deep learning," Applied Sciences, 10(18), 6185,2020.
[31] S. Osher, J. A. Sethian, "Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations," Journal of Computational Physics, 79(1), 12-49,1988.
[32] M. Khare, R. K. Srivastava, "Medical image segmentation using level set method without reinitialization," in International Conference on Signal, Image and Video Processing, India, 2012, pp. 619.
[33] P. Cihan, E. Gokce, O. Kalipsiz. "A review of machine learning applications in veterinary field." Kafkas Universitesi Veteriner Fakultesi Dergisi, 23(4), 2017. DOI: 10.9775/kvfd.2016.17281
[34] P. Cihan, A. Saygili, N.E. Ozmen, M. Akyuzlu. "Identification and Recognition of Animals from Biometric Markers Using Computer Vision Approaches: A Review." Kafkas Universitesi Veteriner Fakultesi Dergisi, 29(6), 2023. DOI: 10.9775/kvfd.2023.30265

Funding

This work was supported by the Turkish Scientific and Technical Research Council-TÜBİTAK (Project Number: 121E349).

Authors' Contribution

The authors contributed equally to the study.

The Declaration of Conflict of Interest/Common Interest

No conflict of interest or common interest has been declared by the authors.

Data Availability

The collected Cattle Retinal Fundus Image dataset is available on Kaggle, https://www.kaggle.com/datasets/animalbiometry/cattle-retinal-fundus-images

The Declaration of Ethics Committee Approval

The study was approved by the Kafkas University Animal Experiments Local Ethics Committee (Protocol number: KAÜ-HADYEK/2024-123).

SAKARYA UNIVERSITY JOURNAL OF COMPUTER AND INFORMATION SCIENCES
http://saucis.sakarya.edu.tr/
e-ISSN: 2636-8129
Publisher: Sakarya University
Vol. 7, No. 3, 2024, 389-403
DOI: 10.35377/saucis...1483300
SAKARYA UNIVERSITY
RESEARCH ARTICLE

Application of Classical and Genomic Cryptography on Textual Dataset

Alev Kaya[1*], İbrahim Türkoğlu[1]

[1] Fırat University, Department of Software Engineering, Elazığ, Türkiye

Corresponding author:
Alev Kaya, Fırat University,
Department of Software Engineering,
Tekirdağ, Türkiye,
alev.kaya@firat.edu.tr

Article History:
Received: 14.05.2024
Accepted: 30.07.2024
Published Online: 10.12.2024

ABSTRACT
Cryptology is one of the methods used when sharing confidential or private data over any communication network that poses a security risk. It is applied to restrict access, minimize or completely prevent dangerous situations. Cryptographic algorithms use a combination of mathematical operations and applications to protect information. It strives to ensure the confidentiality, integrity, availability and non-repudiation of information. In other words, it aims to keep data safe against all kinds of threats. However, the performance of these objectives depends on various factors. These factors include the file format used, the volume and complexity of the data. Additionally, the key system and application platform (software and hardware) also affect performance. These variables determine the effectiveness of cryptographic algorithms. In fact, existing cryptographic algorithms may be inadequate or ineffective in the face of new requirements. Therefore, new techniques need to be designed to meet such needs. This study, one of the new generation cryptographic techniques, includes a symmetric key genomics (DNA)-based application. The aim is to test the suitability of genomic encryption on artificial data sets (100 and 500 KB, 1 and 5 MB) generated from the content named "Siyasetname" in the Turkish textual data type. The usability of the genomic encryption technique, which has not been applied before in Turkish data sets, was tested by comparing it with classical algorithms such as AES (symmetric) and ECDH (asymmetric). Performance criteria are determined as encoding and decoding times (seconds), memory consumption (MB) and processor usage (%), which are accepted in the literature for textual data type. It is supported by different indicators according to dimensions and more successful outcomes compared to similar studies in the literature. These findings suggest that DNA/genomic encryption techniques can be considered as an alternative solution to cryptographic requirements.

Keywords: Genomic (DNA) encryption, Textual data, Siyasetname, AES, ECDH

1. Introduction

Cryptography has changed from past to present with different techniques depending on the conditions. It is one of the methods used to share data over unsecured communication networks [1-4]. Existing cryptographic systems, with their keyed (symmetric/asymmetric) or keyless structures, cannot be fully functional in some cases. Technological advances, the requirements of the digital age and the results of other situations direct the trend towards new solution alternatives [5-13]. One of these new fields is Deoxyribonucleic Acid (DNA) encryption techniques, inspired by its bio-molecular structure and designed with a hybrid of cryptographic algorithms [10-14]. The huge storage capacity of DNA in its natural form and the role it plays in transferring information from generation to generation have been a source of inspiration for cryptosystems. Advantages of DNA encryption techniques include multiple security options, large storage capacity and efficient memory usage. Additionally, extensive key generation and storage facilities are a significant plus. However, limitations such as high experimental costs, complex processing processes and lack of knowledge in biotechnology procedures are the disadvantages of these techniques. Due to these limitations, the actual application of DNA encryption techniques cannot go beyond the laboratory environment. In order to be used in the digital environment, conversion processes are carried out with a hybrid of existing cryptographic methods and mathematical coding techniques. These hybrid approaches aim to provide greater efficiency in practical applications. [10, 12-15].

The aim of this study is to test the suitability of cryptographic use of the genomic-based DNA (gDNA) encryption technique on Turkish text datasets of different sizes. In practice, among the classical/existing cryptographic algorithms, Advanced Encryption Standard (AES) was used symmetrically and Elliptic-Curve Diffie-Hellman (ECDH) was used asymmetrically. As a new generation cryptographic algorithm, the gDNA technique is in symmetric form. The performance of the DNA cryptographic technique in the coding and decoding processes was compared with existing classical algorithms and its use in cryptographic systems was evaluated. The second part of the article provides literature information of related (similar or

Cite as: A. Kaya, İ. Türkoğlu "Application of Classical and Genomic Cryptography on Textual Dataset," *Sakarya University Journal of Computer and Information Sciences*, vol.7, no. 3, pp. 389-403, 2024. Doi: 10.35377/saucis…1483300

close) studies. The third chapter is titled materials and methods; it covers the data set, cryptographic algorithms used and performance metrics. Additionally, this section is presented enriched with visuals. Fourth part; it includes graphical representation of experimental results with tabular content and discussion sections of these results. In addition, in this section, the contribution of the study and the information of similar or close studies are concretized in a different table. Thus, a more striking comparison was aimed. In the last section, the results are analyzed and their pros and cons are discussed. Suggestions are given with the aim of contributing to future studies. The highlights of the study are listed below.

• Coding and decoding of artificially produced textual data sets in different sizes from the content named Siyâsetnâme (Nizamülmülk) [16], which is a Turkish content data set,

• Evaluation of time (s), memory (MB) and CPU (%) consumption rates as performance metrics of symmetric (current: AES and new generation: DNA) and asymmetric (ECDH) algorithms used for the application,

• The detailed content of the DNA encryption algorithm, which is one of the new generation techniques in cryptosystem, is given to the Turkish literature. In addition, it is aimed to present not only research but also practical use.

2. Related Works

To date, significant progress has been made in the development of cryptographic algorithms. In this process, basic encryption methods such as single-alphabet ciphers, multi-alphabet substitution ciphers, transfer ciphers and block ciphers were initially developed [5-15]. Following the early techniques, more complex and powerful cryptographic algorithms such as AES, Data Encryption Standard (DES), Rivest-Shamir-Adleman (RSA) and Secure Hash Algorithm (SHA) have emerged [6-8]. Monoalphabet ciphers replace letters of plaintext with other letters using a specific alphabet. Multi-alphabet substitution ciphers perform a more complex substitution process by using more than one alphabet. Transfer ciphers provide security by reversing the letters of the text. Block ciphers are for higher security by encrypting text in blocks of specific sizes. Among these algorithms, AES has become a prominent standard among symmetric encryption algorithms. It offers superior features in terms of both performance and security. DES has been widely used before. However, due to its insufficient key length, more secure algorithms like AES eventually replaced it. RSA is one of the most well-known asymmetric encryption algorithms. It enabled secure data transmission using public and private key pairs. SHA is among the secure hashing algorithms. It is used to protect the integrity of data. These developments have led to the emergence of more robust and effective encryption methods in the field of information security. It has become one of the alternatives to better meet the security needs of the digital world. The evolution of cryptographic algorithms, together with constantly improving technology, has resulted in the design of more efficient information security solutions [4-15]. The foundation of DNA cryptography, one of these possibilities, was laid [17, 18]. It originated from the quest to find a new computational model to meet the requirements for large amounts of processing power and storage. The biological functioning of the DNA molecule inspired this new computational model. In particular, Adleman's [17] attempt to find a solution to the Hamilton Path Problem with DNA computation in 1994 aroused great interest in the field of DNA computation. Then, in 1995, researchers such as Lipton [18] published about DNA computers and (non-deterministic Polynomial) NP-complete problems. The timing problem of 2-bit complex numbers was solved in the test tube using DNA molecules. Success has been achieved with this method. DNA cryptography is inspired by the fact that data can be biologically encoded in DNA strands. This approach is seen as a new source of hope for unbreakable algorithms. Because the nature of DNA provides a very complex and secure storage environment. Therefore, DNA computation and cryptography may offer a new and effective approach to secure data. Based on the application of the article, some of the similar or close studies [19, 20, 22-29] conducted in the last five years are mentioned below.

Before uploading textual data to the cloud environment, a multi-layered symmetric key DNA cryptography technique has been proposed [19]. This study was proposed in 2018; it was compared with classical symmetric structured AES, DES and Blowfish algorithms. Performance criteria of the experimental application; it was evaluated in terms of efficiency, namely cipher text size, encryption time and transfer speed. It has been observed that the proposed technique is more successful. To ensure the confidentiality and security of the textual dataset, another algorithm based on asymmetric key DNA cryptography, consisting of three steps, was proposed in 2019. The result of the study [20] was compared with existing techniques. It has been tested with statistical tests according to the National Institute of Standards and Technology (NIST) [21] to analyze the security analysis and the randomness of the generated cipher text. Another study [22] in which DNA-based cryptography was applied on textual data type was published in 2020. Cryptography has been applied so that some of it is symmetrical and the other part is asymmetrical. First, the combination of DNA coding with algorithms called One Time Pad (OTP) was taken. Secondly, DNA codes in the RSA algorithm were applied. The efficiencies of DNA coding in OTP, RSA and other algorithms are given. As a result, it was seen that the calculation time of the RSA algorithm combined with DNA coding was longer. To solve this problem, data redundancy is reduced by enabling the GZIP compressed algorithm. Current experimental results show that DNA symmetric cryptography works quite well in both time and dimension analyses. Compressed DNA has shown to be less efficient than asymmetric cryptography. Another study [23] evaluating the performance of genetic (DNA) and classical (DES, AES and RSA) encryption algorithms is published in 2021. The application was made on textual data sets of different sizes (58 KB, 100 KB, 1 MB and 5 MB). It is encoded by generating 64 or 128 bit random keys. Additionally, an interface platform is provided for the coding process. Time, CPU and RAM were evaluated as performance metrics in

encryption and decryption. It has been observed that symmetric algorithms give better results in large-sized data sets, and genetic encryption algorithms give better results in small-sized data sets. Another study [24] on textual datasets of different sizes is in 2021. Hybrid structured models based on symmetric (Blowfish and DES) and DNA-based encryption algorithms were used. Their performances were compared in terms of time, memory and processor complexity. General evaluations have been made for the application, which gives different result indicators according to different platforms. The current non-hybrid Blowfish algorithm has been shown to be more successful. However, it was found that it slowed down even more in terms of time after the hybrid was made with DNA-based techniques. It resulted in significant performance loss in the time display of the Blowfish algorithm on large data sets. However, the DES algorithm gave better performance on big data. In addition, DES and DNA hybrid algorithms have been shown to be better than studies in the literature. Study in 2022 [12]; it was about image, text and video data types. The process of transforming DNA as a carrier and a means of applying modern biological techniques has been carried out. The structure of the proposed DNA cryptography was created by integrating DNA operators into the Feistel network structure. The simulation software developed in the experimental application and the synthesized DNA content were integrated with biotechnical hardware. Evaluation criteria; the capacity is about 100%, the brute force attack is about 12 × 106 years, the key space is 2, and the entropy analysis is close to 2. It was interpreted as effective for cryptographic requirements and the results were confirmed by in vitro experiments. The study in 2023 [25] focused on DNA-based new generation cryptographic techniques. Similar or close studies in the literature are mentioned and the general process of biological structure is given. The process steps of multiple formats in the encoding and decoding processes are explained with detailed explanations. There is plain data (original form) consisting of a string of characters of textual type as words. A small bio-molecular based application has been made for the key index created by a hybrid of words and numbers. The encryption and decryption stages were implemented without any problems. It has been shown that the infrastructure can be adapted to storage studies along with cryptographic requirements. Another study in 2023 [26] is the application of different symmetric encryption algorithms on textual data type. The application again consists of data sets of different sizes. Performance metrics in comparing algorithms; CPU, RAM and time. When the results are analyzed, it has been observed that the performances of Blowfish, Salsa20, 3DES, Cast, AES and DES algorithms are better in all data sets. A similar study [27] was also conducted in 2023. The only difference from the previous source is the cryptographic application with asymmetric algorithms. Again, coding and decoding processes were carried out on textual data sets with different volumetric content. Performance criteria are again RAM, time and CPU. When performance analysis was performed, ECDH, El-Gamal and RSA rankings were obtained in all volumetric data sets. Finally, another different study [28] was conducted in 2023. This study was again carried out on textual data sets of different sizes. In the cryptographic process, symmetric (AES, Blowfish, Cast-128) and asymmetric (ECDH, El-Gamal, RSA) encryption algorithms have been implemented. Performance criteria were evaluated based on encryption and decryption speed (s), memory (MB) and CPU (%) usage rates. Thus, the performances of symmetric and asymmetric encryption algorithms were compared on the same data sets. As a result, it was observed that the performance results of symmetric encryption algorithms were better in all volumetric data sets. Based on the studies and gaps in the literature, the goal of this application is given below.

- As far as researched, DNA or other (RNA, amino acid, etc.) genomic-based cryptographic applications have not been used in converting a Turkish data set. Only character transformations have been mentioned in the studies in the literature. Additionally, it has not been tested on data sets containing Turkish characters of different sizes. For this reason, the study is the first to use Turkish content datasets of different sizes. It is aimed to present the proposed algorithm as one of the alternatives in cryptographic requirements.

3. Material and Method

Information called confidential or private may require additional processing in environments where there is a security risk. These transactions; this may include restricting access from unauthorized persons, minimizing or completely preventing dangerous situations. One of the methods used for this is cryptology. Cryptographic algorithms; the transaction is made with keyed (symmetric, asymmetric), keyless or hybrid components of the original data. Changing, or locking, this original data with the help of mathematical functions is called encryption. Transforming it to its original state, that is, opening the lock, with the help of reverse mathematical operations is decryption. Cryptographic systems, which have changed with different techniques according to conditions from past to present, are not fully functional in some cases. Technological breakthroughs, requirements of the digital age and other situations, etc. The results lead to new solution alternatives. This is one of the new areas; they are genomic-based (DNA, RNA, amino acid ...) encryption techniques inspired by the bio-molecular structure and designed with a hybrid of cryptographic algorithms [1-15]. In this study, which is one of the new generation cryptographic techniques, a DNA-based application with a symmetric key was made. This paper; it includes a comparative application of current and next generation cryptographic systems. The purpose of the application; it is a test of the suitability of the cryptographic use of the gDNA encryption technique in data sets of different sizes (100 and 500 KB, 1 and 5MB) artificially created from the Turkish content named "Siyasetname" in the textual data type. It has been observed that the gDNA encryption technique has not been extensively applied on a Turkish data set before. Artificial data sets with different volumetric contents were created. Comparison with existing/classical (AES (symmetric), ECDH (asymmetric) algorithms supported application results. Performance criteria; accepted criteria in the literature for textual data type were taken into account. These are the time (seconds), memory (Megabytes) and processor (%) usage rates in encoding and decoding. In Figure 1, the course of the study is visualized as the main flow chart. It is explained and detailed in the following sections.

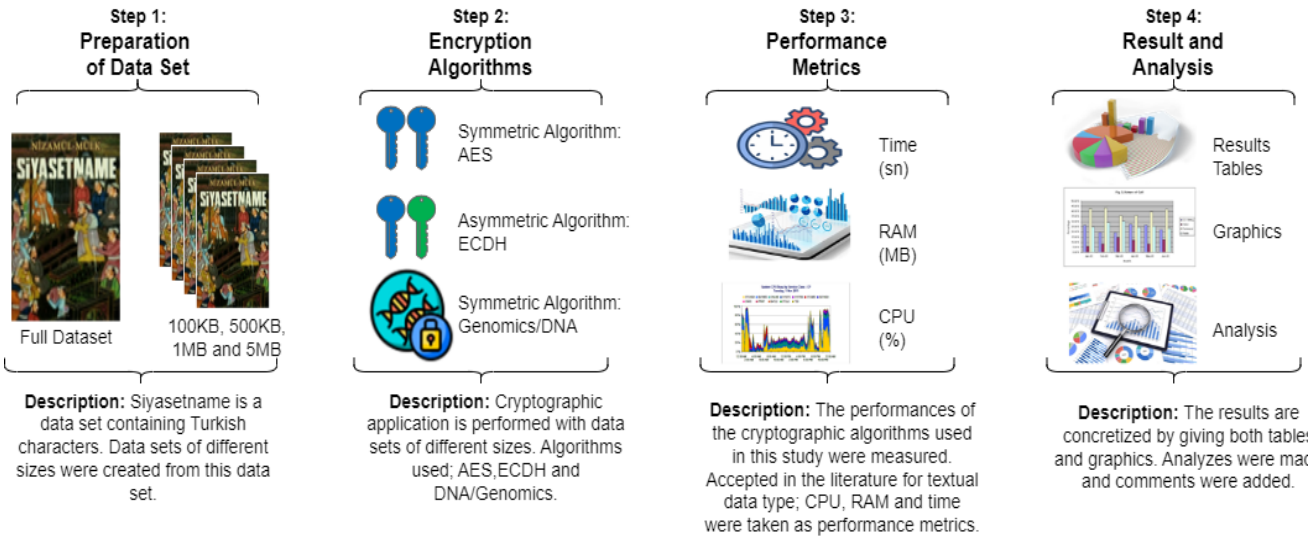


Figure 1. Main Flow Chart of the Study

3.1. Data Set Preparation

In this study, a data set with Turkish content called "Siyasetname" or "Siyeru'l-mülk" [16] written by Nizamülmülk was used. Artificial data sets with four different volumetric contents were created from this original data set. The actual size of the original content is 1.60 MB. Volumes of artificially generated new datasets; it is between 100 KB, 500 KB, 1 MB and 5 MB. Data duplication was done for the 5 MB data set.

3.2. Encryption Algorithms

DNA-based coding and decoding application, one of the new generation cryptographic systems, has a symmetric key structure. The performance of this structure was evaluated by comparing it with existing/classical algorithms. AES was used as the classical existing symmetric encryption algorithm and ECDH was used as the asymmetric encryption algorithm. In particular, the selection of these two algorithms is based on three previous studies [26]-[28]. Because of these three studies, the symmetric and asymmetric encryption algorithms that gave the most successful outputs were included in this application. The purpose of these three studies is to determine the performance of classical encryption algorithms on a Turkish data set of different sizes. Because of the studies, the classical encryption (symmetric and asymmetric) algorithms that give the best performance are determined.

3.2.1. Figures Symmetric Encryption Algorithm: AES

A symmetric encryption system is designed by using the same public key to be secret during the encoding and decoding phases. Since only a single secret key is used in these algorithms, they are easier to break than asymmetric algorithms. However, since there is only one key (secret), the encryption and decryption time is advantageous compared to asymmetric or other hybrid structures. Therefore, it is faster. This also provides superiority over asymmetric structures in encrypting large data. It consumes less system resources (CPU, RAM, etc.). It works compatible with the hardware. The main disadvantage here is that all security depends on the key and its size. The key size can be created in longer directories than asymmetric ones [29]-[32].

AES, one of the popular symmetric encryption algorithms, is used in this application. The working architecture of the AES cryptographic system is visualized in Figure 2. In addition, the working logic of the AES algorithm is given in Algorithm 1, "Pseudo code of the AES algorithm". This pseudo code; it includes key generation, encryption process and decryption processes. AES was designed by NIST in 2001 to be a standard for encrypting digital data. It has a block encryption structure. Key lengths consist of 128 (10 rounds)/192 (12 rounds)/256 (14 rounds) bit options. However, the coding process is done with bytes instead of bits. It takes all data as input, that is, the original contents in 128 bits (16 bytes). It gives the encrypted form as output, again as 128-bit blocks. Each block evaluates as a grid of 4 row bytes x 4 column bytes = 128 (16 byte) bits in main column order. Manipulating and shuffling input data through a series of linked steps is known as the substitution-permutation network principle. Each round consists of 4 steps. The AES instruction set is now integrated into the CPU to increase the speed and security of applications that use AES for encryption and decryption. This offers a transfer speed of several GB/s. Although it has been 20 years since its launch, the AES algorithm, which is not possible even with current technology, has not been cracked. To date, the only vulnerability remains in the implementation of the algorithm [29]-[33].

Key Generation

Input: *Key*
Output: *Round keys*
Method: *Generate the round keys by expanding the initial key.*

Step 1. Check Key Length:
- *If the key length is not 128 bits, the algorithm returns an error.*

Step 2. Initialization:
- *Convert the key into 4-byte words (32-bit).*
- *Determine Nk, Nb, and Nr:*
  - *Nk is the key length divided by 32 (4 words for 128-bit key).*
  - *Nb is the number of columns in the state matrix (4).*
  - *Nr is the number of rounds (10 for 128-bit key).*

Step 3. Place Initial Keys:
- *Place each byte of the key into the round key matrix sequentially.*

Step 4. Key Expansion:
- *Expand the remaining Nb * (Nr + 1) words:*
  - *temp = round_keys[i - 1]*
  - *If i is a multiple of Nk:*
    - *temp = SubWord(RotWord(temp)) ^ Rcon[i / Nk]*
  - *If Nk > 6 and i % Nk == 4:*
    - *temp = SubWord(temp)*
  - *round_keys[i] = round_keys[i - Nk] ^ temp*

Encryption

Input: *Plaintext*
Output: *Ciphertext*
Method: *Encrypt the plaintext using the AES algorithm.*

Step 1. Initialize State:
- *Convert the plaintext into the initial state matrix.*

Step 2. Add Initial Round Key:
- *Add the initial round key to the state matrix.*

Step 3. Rounds (1 to 9):
- ***SubBytes:*** *Transform each byte in the state matrix using the S-Box.*
- ***ShiftRows:*** *Shift each row in the state matrix to the left by the row number.*
- ***MixColumns:*** *Multiply each column in the state matrix by a fixed matrix.*
- ***AddRoundKey:*** *Add the corresponding round key to the state matrix.*

Step 4. Final Round:
- ***SubBytes:*** *Transform each byte in the state matrix using the S-Box.*
- ***ShiftRows:*** *Shift each row in the state matrix to the left by the row number.*
- ***AddRoundKey:*** *Add the final round key to the state matrix.*

Step 5. Output:
- *Extract the state matrix as the ciphertext.*

Decryption

Input: *Ciphertext*
Output: *Decrypted text*
Method: *Decrypt the ciphertext using the AES algorithm.*

Step 1. Initialize State:
- *Convert the ciphertext into the initial state matrix.*

Step 2. Add Final Round Key:
- *Add the final round key to the state matrix.*

Step 3. Rounds (1 to 9):
- ***InvShiftRows:*** *Shift each row in the state matrix to the right by the row number.*
- ***InvSubBytes:*** *Transform each byte in the state matrix using the inverse S-Box.*
- ***AddRoundKey:*** *Add the corresponding round key to the state matrix.*
- ***InvMixColumns:*** *Multiply each column in the state matrix by the inverse fixed matrix.*

Step 4. Final Round:
- ***InvShiftRows:*** *Shift each row in the state matrix to the right by the row number.*
- ***InvSubBytes:*** *Transform each byte in the state matrix using the inverse S-Box.*
- ***AddRoundKey:*** *Add the initial round key to the state matrix.*

Step 5. Output:
- *Extract the state matrix as the decrypted text.*

Algorithm 1. Pseudo code of AES algorithm

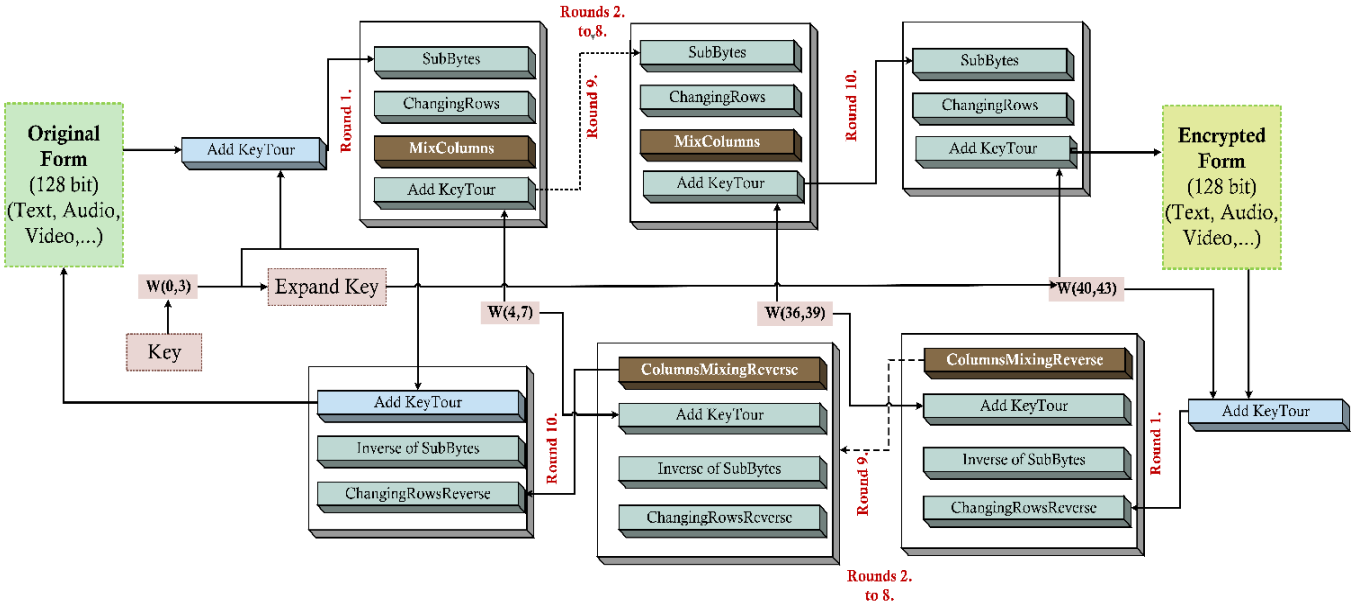


Figure 2. Working Architecture of the AES Cryptographic Algorithm

3.2.2. Asymmetric Encryption Algorithm: ECDH

Asymmetric encryption system forms the infrastructure of digital signatures. While encoding and decoding, two different keys called, secret and open are used. The public key is used in the encryption process. It is common between the parties (message sender-receiver). The secret key is used for decryption. It is special only in decryption. Since there is no single key structure, there is no need for frequent key changes as in symmetric systems. The reason is that the public key has no effect on the decryption phase. This situation also has another disadvantage. Encrypted contents are more difficult to crack than symmetric systems. They can also manage their processes with shorter length key sequences. However, due to the secret key that the parties cannot share, the decryption time is higher (slower) than in the symmetric structure. Additionally, the size of the file to be encrypted increases this situation exponentially. Another disadvantage is that it requires difficult mathematical functions to ensure data/information security. Because the calculation content includes large prime numbers. This consumes the processor and other system resources more than symmetric algorithms. It forces integration with hardware structures. For such reasons, data size is an important constraint in the encryption process.

In other words, it can be applied on lower dimensional data sets compared to symmetrical structures [29]- [32]. Asymmetric crypto systems are aimed at two main uses: asymmetric encryption and digital signatures. Digital signatures are a way to verify that any message comes from the owner of a specific private key and that the information has not been tampered with in transit. In this application, ECDH encryption algorithm, one of the popular asymmetric encryption algorithms, is used. The ECDH cryptographic system is visualized in Figure 3 under the name of its working architecture. In addition, the working logic of the ECDH algorithm is given in Algorithm 2, "Pseudo code of the ECDH algorithm". This pseudo code; it includes key generation, encryption process and decryption processes. It is based on the Diffie-Hellman (DH) algorithm, which dates back to 1976 [34] and is one of the first public key protocols. ECDH algorithm is known as an encryption algorithm in crypto systems. However, it is more commonly known as the agreement protocol for all transactions of the key up to a certain level. These transactions; how to produce, how to share between mutual parties, etc. Contains. The actual applications of the encryption process also depend on how these key operations are used. ECDH itself is a key exchange protocol. It is not designed for direct message encryption operations. ECDH is used to securely generate a shared key between parties. This shared key is then used to encode and decode the actual messages, often using it with a symmetric encryption algorithm (e.g. AES). Encryption process; it covers the protections made to prevent the third party named Oscar from gaining access while communicating between the parties named Alice and Bob. Elliptic algorithms perform operations on a curve. Alice and Bob agree on a curve with a starting point. Let us call this P. Alice has a private key a and a public key A = a * P. Bob has a private key b and public key B = b * P. Combining the equations; a * B = a * b * P = b * A. Thus a * b * P becomes the shared secret [35].

Key Generation

Input: *Private key*
Output: *Public key*
Method: *Generate the public key using the private key and elliptic curve parameters.*

Step 1. Check Private Key Length:

- *Ensure the private key is of the correct bit length according to the elliptic curve used.*

Step 2. Initialization:

- *Select the elliptic curve parameters (curve equation, base point G, order n).*

Step 3. Generate Public Key:

- *Alice and Bob each form a private key (a and b). These keys are usually random numbers and are kept secret during the transaction.*
- *Calculate the public keys:*
  - *Public key of Alice (Qa): $Qa=a \cdot G$*
  - *Public key of Bob (Qb): $Qb=b \cdot G$*
- *The result Qa and Qb are the public keys.*

Key Exchange

Input: *Private key (own), Public key (peer)*
Output: *Shared secret*
Method: *Generate the shared secret using the own private key and peer's public key.*

Step 1. Initialize:

- *Use your private key (a or b) and peer's public key (Qb or Qa).*

Step 2. Calculate Shared Secret:

- *Alice computes the shared private key (K) by using her private key (a) and Bob's public key (Qb):*
  - *$K=a \cdot Qb$*
- *Bob computes the same shared secret key (K) by using his private key (b) and Alice's public key (Qa):*
  - *$K=b \cdot Qa$*
- *The result K is the shared secret.*

Encryption

Input: *Plaintext, Shared secret*
Output: *Ciphertext*
Method: *Encrypt the plaintext using the shared secret.*

Step 1. Derive Symmetric Key:

- *Use a key derivation function (KDF) to derive a symmetric key K from the shared secret K, if necessary.*

Step 2. Encrypt Plaintext:

- *Alice creates the encrypted message (C) using the shared secret key (K) and the message (M):*
  - *$C=M+K$*

Step 3. Output:

- *The result is the ciphertext (C).*

Decryption

Input: *Ciphertext, Shared secret*
Output: *Decrypted text*
Method: *Decrypt the ciphertext using the shared secret.*

1. ***Derive Symmetric Key:***

- *Use the same key derivation function (KDF) to derive the symmetric key K from the shared secret K, if necessary.*

2. ***Decrypt Ciphertext:***
   - *Bob decrypts the encrypted message (M) using the same shared secret key (K) and encrypted message (C):*
     - $M=C-K$
3. ***Output:***
   - *The result is the decrypted text (M).*

Algorithm 2. Pseudo code of ECDH algorithm

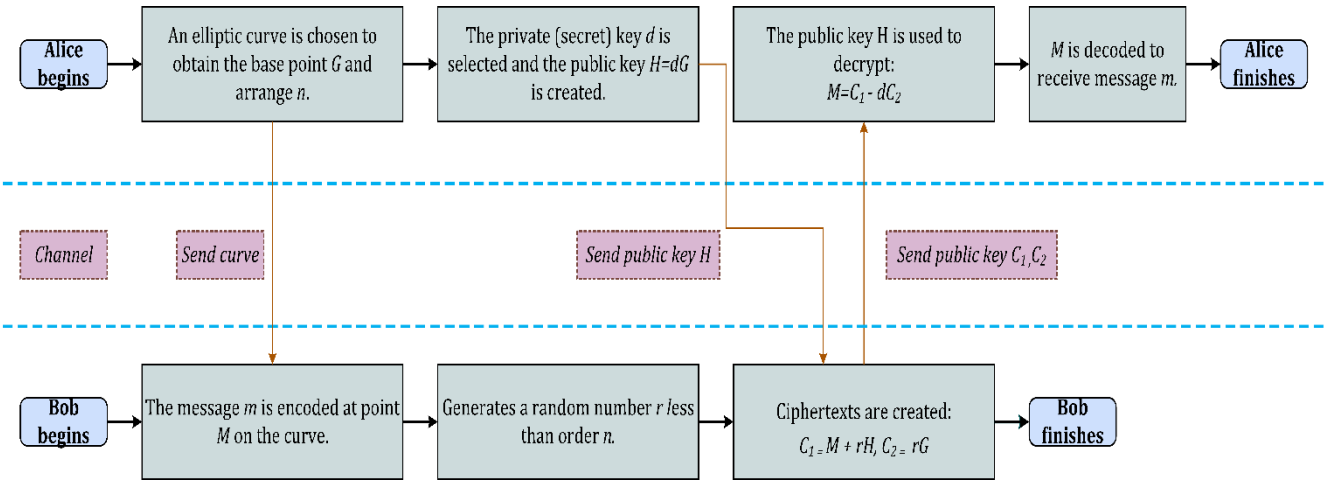


Figure 3. Working Architecture of the ECDH Cryptographic Algorithm

3.2.3. New Generation Encryption Algorithm: DNA

One of the methods used to ensure information security is current/classic cryptographic algorithms. Crypto systems, which have been developed to date with different add-ons according to needs, can sometimes be insufficient or ineffective. One of the new methods proposed to find solutions to such problems is genomic-based crypto systems [1-5]. Genomics-based crypto systems are inspired by bio-molecular structures such as DNA, RNA and amino acids. These systems are inspired by natural functionality such as transcription and translation in living form during the encryption and decryption stages. Then, the inspired form tries to digitize it by transforming it with mathematical techniques. In other words, it plays the role of a hybrid implementation tool of living functionality with mathematical techniques. It carries out its role as a carrier medium through biological processes. Genomics-based algorithms as actors have major roles to play in cryptographic systems. Large-volume data storage and high-end security capabilities are a priority [1-4, 8, 9, 11-13]. However, biotechnology has disadvantages such as limited knowledge of procedures, complex structure of processes and cost of empirical environments. Due to such disadvantages, real applications are limited to the laboratory level. Based on this restriction, the bio-molecular structure is transformed by coding techniques. Thus, it is digitized with a hybrid of existing cryptographic algorithms [10, 12-15, 19].

3.2.3.1. Genomics-based DNA (gDNA) Encryption and Decryption Process

DNA is the structure that undertakes hereditary duties as the information carrier of life. A long biopolymer resembles the interlocking teeth of a zipper. These biopolymers consisting of sequences are called nucleotides. In the mutual chain of each zipper, there are nucleobases Adenine: A, Guanine: G, Cytosine: C and Thymine: T. These nucleobases are ordered by pairing A = T with G = C. These different sequential sequences (combinations) within each other are natural coding. It is in the form that stores and transmits information. This four-character (A, T, G, C) language form of life must be transformed into a language form (0 and 1) that digital systems can understand. This requires a hybrid of existing cryptographic systems and mathematical transformation techniques [10-15, 17-21, 22-25, 29, 30]. Additionally, a pseudo code explaining the encryption and decryption stages of the DNA algorithm is given in Algorithm 3. Its detailed explanation is given below.

Data Set Preparation

Input: *Text data*
Output: *Binary representation of text data*
Method: *Convert text data to its binary representation.*

Step 1. Convert Characters to ASCII:

- *Convert each character of the text data to its ASCII equivalent.*
- *If there are Turkish characters, use a format such as UTF-8 or UTF-16 for wide character set conversion.*

Step 2. Map to Binary Representation:

- *Map ASCII or other numeric conversions to their binary representation counterparts.*

Key Preparation

Input: *None (or a predefined key)*
Output: *Symmetric key*
Method: *Generate a symmetric key for the encryption system.*

Step 1. Generate Symmetric Key:

- *Generate a symmetric key (secret) for the encryption system.*
- *If you are not going to use a key, skip this step.*

Key Expansion

Input: *Symmetric key, Original data*
Output: *Expanded key*
Method: *Expand the binary format of the secret key to the size of the original data.*

Step 1. Expand Key:

- *Expand the binary format of the secret key to match the size of the original data.*

Encryption

Input: *Binary representation of text data, Expanded key*
Output: *Encrypted form*
Method: *Encrypt the binary data using the expanded key and a suitable transformation.*

Step 1. Concatenate Data and Key using XOR:

- *Concatenate the original data in binary form and the expanded key using the XOR technique.*

Step 2. Transform Binary Form:

- *Transform the final combined binary form with A (00), G (01), C (10), and T (11) or another suitable transformation.*

Step 3. Save Encrypted Form:

- *Save the encrypted form.*

Decoding

Input: *Encrypted form*
Output: *Original text data*
Method: *Decrypt the encrypted form to retrieve the original text data.*

Step 1. Convert Encrypted Data to Binary:

- *Convert the recorded encrypted 00 (A), 01 (G), 10 (C), and 11 (T) data to the original binary structure.*

Step 2. Reverse XOR Operation:

- *Obtain the original binary data by performing a reverse XOR operation using the expanded key.*

Step 3. Convert Binary to ASCII:

- *Convert binary data to ASCII characters to get the original text data.*

Algorithm 3 Pseudo code of DNA algorithm

The working steps are visualized in Figure 4 under the name "Working architecture of the DNA cryptographic algorithm". Detailed explanation is given below.

1. Each character in the data type in the original form (text, image, signal … etc.) is converted to its ASCII equivalent. If there are Turkish characters, use UTF-8 or UTF-16 for wider character set conversion, etc. is used.

Recommended in the study: The work named Siyâsetnâme (Nizamülmülk) [16], which is a data set containing Turkish characters, was used. Textual datasets of different sizes (100 KB, 500 KB, 1MB and 5MB) were created.

2. Translated ASCII or other numeric conversions (UTF, etc.) are matched to their binary base counterparts. The aim is to transform the digital form so that the digital form can understand it.

3. If a crypto system with a key (symmetric, asymmetric or hybrid) is designed, all operations applied in the first and second steps are applied exactly within the key. If it is a keyless structure, the key steps are skipped.

Recommended in the Study: A crypto system with a symmetric key was designed. The secret key used is the 32-character (256 bit = 32 byte) "1086_NizamülmülkSiyasetname_1092" index.

4. The key index in binary form is expanded by the size of the original data. The goal is to achieve compliance in the fifth step.

5. The original data in the binary structure and the extended key index in the binary structure are combined with the XOR [36, 37] technique.

6. This final combined binary form is completed by being converted into A (00), G (01), C (10) and T (11) structure, which is the language of life. The order of coding nucleobases can be changed. In the literature, this order is listed as eight basic coding rules [36-38]. This encoding order is irrelevant in the accepted performance metrics (time, RAM and CPU) for textual data type in encryption and decryption processes. The decryption process consists of the exact reverse process of these six steps. In the visual, the fifth work package explains these steps.

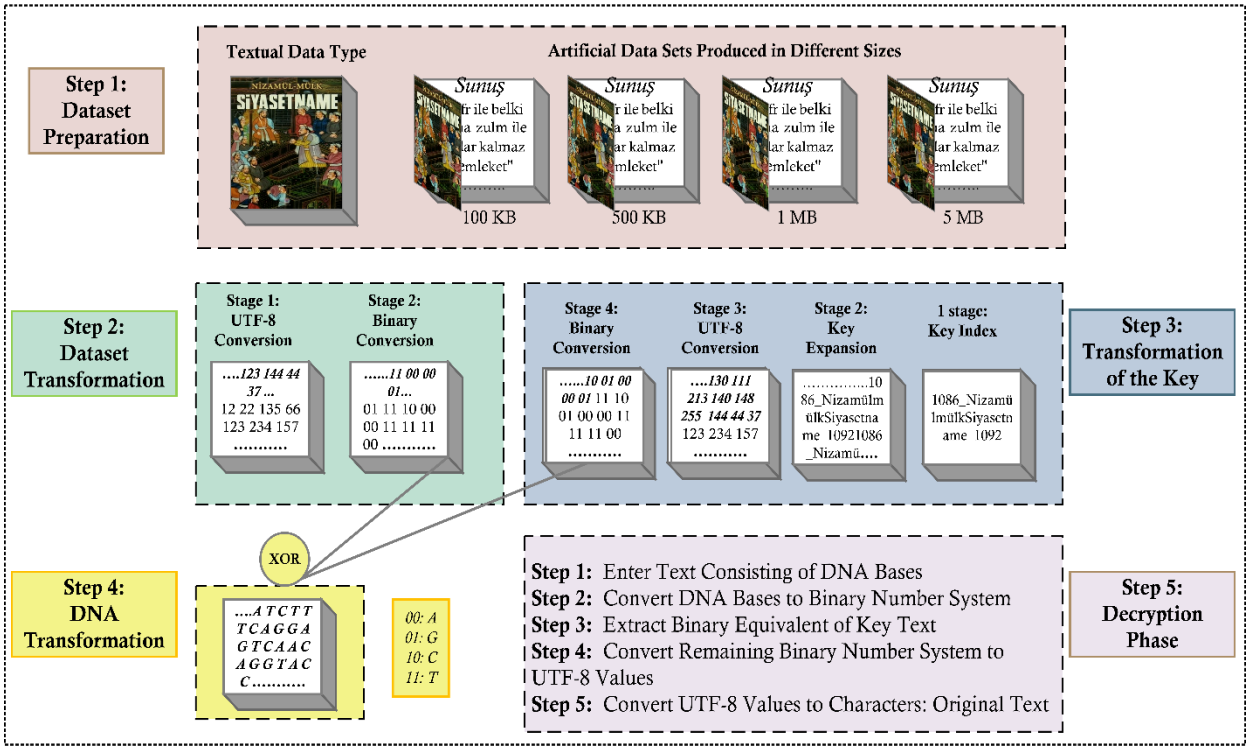


Figure 4. Working Architecture of the DNA Cryptographic Algorithm

3.3. Performance Metrics

This work is for the application of gDNA encryption in symmetric form. It is aimed to evaluate the use of the DNA encryption algorithm in cryptographic requirements. For this reason, two classical algorithms were used. These algorithms; they are AES (symmetric) and ECDH (asymmetric). In the application on textual data type, criteria accepted in the literature were taken into account as performance metrics. These criteria; duration/processing time (seconds), processor (CPU :%) and memory/memory (RAM: Megabyte (MB)) consumption amount [23],[24]. Analysis of performance metrics "Hardware and software features of the computer that uses computing power in practice." It was concluded according to the structure in Table 1. Consumption speeds of algorithms may vary depending on different computer features.

Table 1. Hardware and Software Features of the Computer Whose Computing Power is Used in Practice

Requirements	**Feature**
Processor	Intel(R) Core(TM) i7-10750H CPU @ 2.60GHz (12 CPUs), ~2.6GHz
Memory	16384MB RAM
Operating System	Windows 11 Pro 64-bit
Software	Python 3.9.13, Vs Code 1.78.0.

4. Experimental Results and Discussion

This study involves the application of gDNA cryptography on a textual Turkish content dataset. gDNA cryptography is accomplished with techniques inspired by bio-molecular processes. A (mixed) character sequence consisting of numbers and letters is used as a symmetric keying system. The secret key used is the 32-character (256 bit = 32 byte) index "1086_NizamülmülkSiyasetname_1092". The purpose of the application is to test the suitability of using the gDNA encryption technique in cryptographic requirements. Four new data sets of different sizes were artificially created from the original data set called "Siyasetname". Sizes are between 100 and 500 KB, 1 and 5MB. Comparison with existing/classical (AES (symmetric), ECDH (asymmetric) algorithms support application results of DNA-based encryption techniques. Performance metrics are criteria accepted in the literature for the textual data type. These are time (seconds), memory (Megabytes) and processor (%) usage rates in encoding and decoding. The images from Figure 5 to Figure 12 show the outputs of the application, that is, the encoding and decoding results.

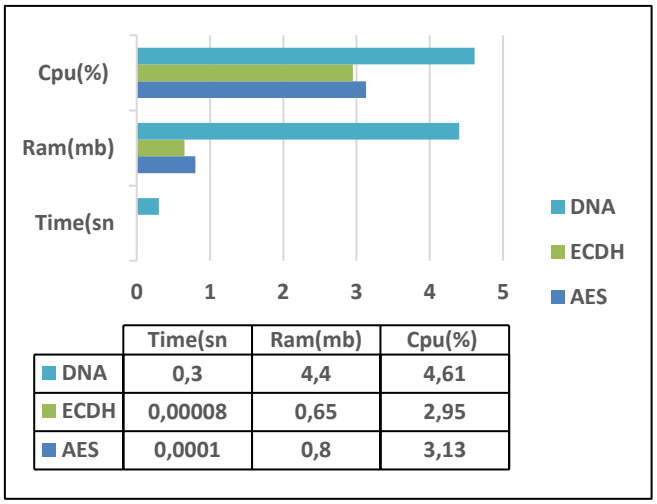


Figure 5. Encryption Performance Results of 100 KB Data Set

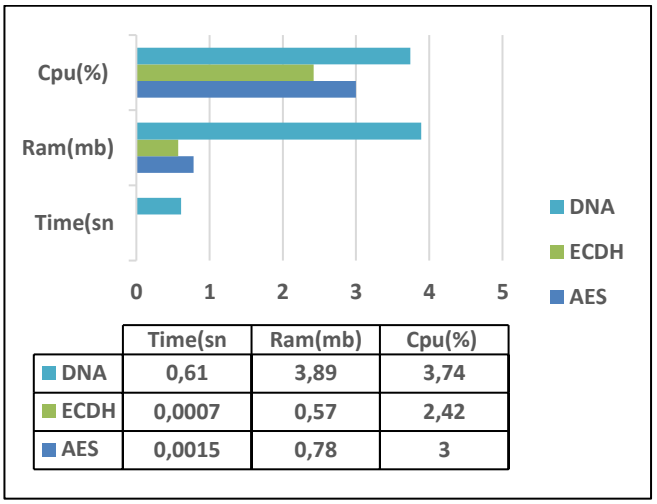
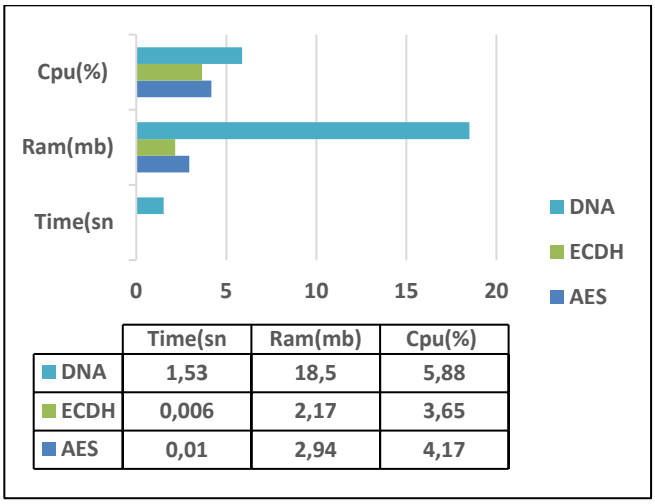


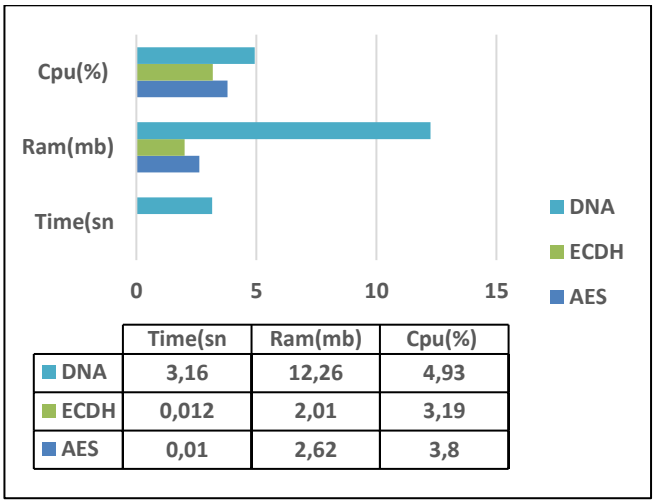
Figure 6. Decryption Performance Results of 100 KB Data Set

Figure 5 and Figure 6 are the encryption and decryption graphs of the 100 KB text dataset, respectively. The results in these two images, which appeared in a small data set, are close to each other. When the conversion of gDNA is calculated, the result of encryption and decryption is at an acceptable level. AES and ECDH algorithms work very fast and efficiently, especially on small data sets. Genomic (DNA) encryption, on the other hand, takes longer in terms of both coding and decoding time. Additionally, RAM and CPU usage is also higher. This is because DNA-based encryption requires more complex algorithms and data structures when processing biological data



	Time(sn	Ram(mb)	Cpu(%)
DNA	1,53	18,5	5,88
ECDH	0,006	2,17	3,65
AES	0,01	2,94	4,17

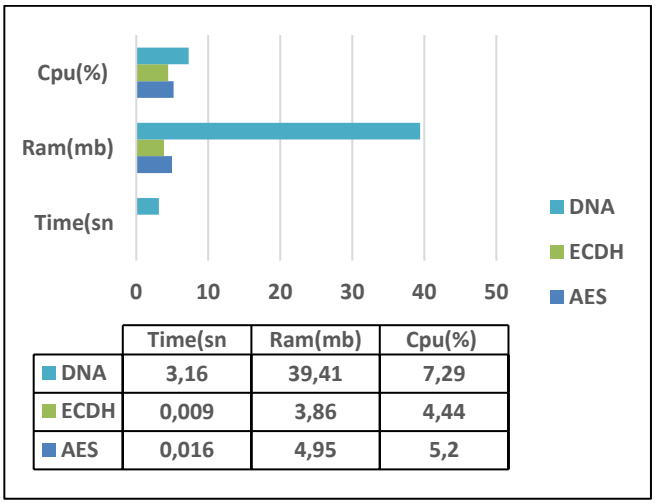
Figure 7. Encryption Performance Results of 500 KB Data Set



	Time(sn	Ram(mb)	Cpu(%)
DNA	3,16	12,26	4,93
ECDH	0,012	2,01	3,19
AES	0,01	2,62	3,8

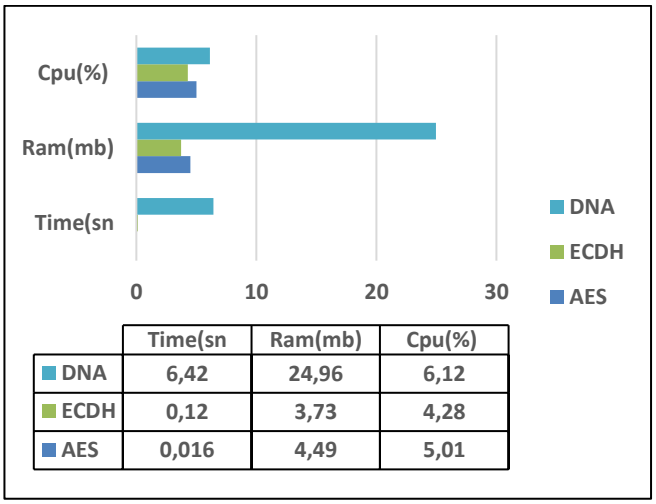
Figure 8. Decryption Performance Results of 500 KB Data Set

Figure 7 and Figure 8 are the encryption and decryption graphs of the 500 KB text dataset, respectively. In these two images, which appeared in a medium volume data set, the results gradually became clearer. Here again, when the conversion of gDNA is calculated, the result obtained in encryption and decryption is at an acceptable level. On medium-sized data sets, AES and ECDH algorithms remain fast and efficient. Genomic (DNA) encryption, on the other hand, takes longer and consumes more resources. DNA-based encryption algorithms cause processing time and resource consumption to increase as data grows.



	Time(sn	Ram(mb)	Cpu(%)
DNA	3,16	39,41	7,29
ECDH	0,009	3,86	4,44
AES	0,016	4,95	5,2

Figure 9. Encryption Performance Results of 1 MB Data Set

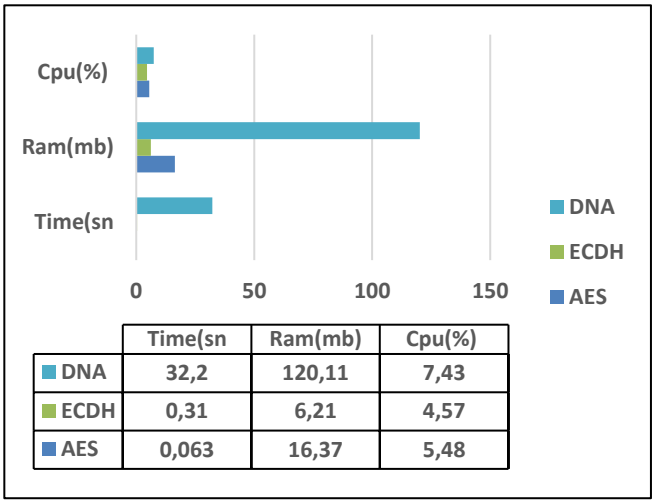


	Time(sn	Ram(mb)	Cpu(%)
DNA	6,42	24,96	6,12
ECDH	0,12	3,73	4,28
AES	0,016	4,49	5,01

Figure 10. Decryption Performance Results of 1 MB Data Set

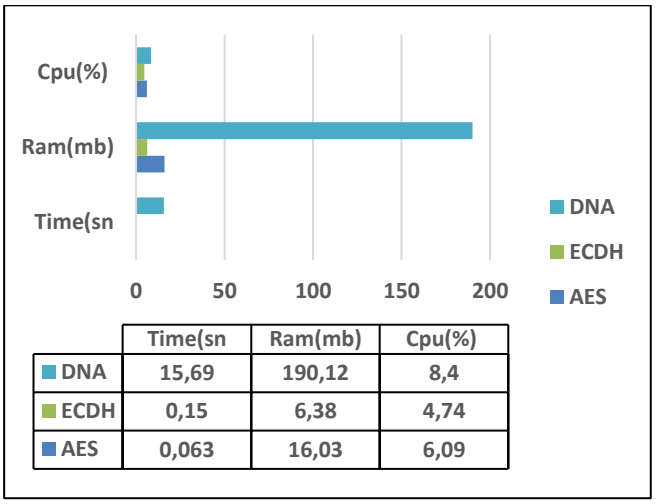
Figure 9 and Figure 10 are the encryption and decryption graphs of the 1 MB text dataset, respectively. In these two images, which appear in a large data set, the range of results is widened. On large data sets, AES and ECDH still work quite efficiently. Genomic (DNA) encryption, on the other hand, requires much more time and resources. As the data size increases, the computational complexity of DNA-based encryption and the resources required to process the data structures further increase.

Finally, Figure 11 and Figure 12 are the encryption and decryption graphs of the 5 MB text dataset, respectively. In these two images, which are another large data set, the range of results is widened. AES and ECDH algorithms also perform with acceptable times and resource usage on large data sets. gDNA encryption, on the other hand, works very slowly and resource intensive on large data sets. This is because DNA-based encryption requires complex biological data structures and algorithms.



	Time(sn	Ram(mb)	Cpu(%)
DNA	32,2	120,11	7,43
ECDH	0,31	6,21	4,57
AES	0,063	16,37	5,48

Figure 11. Encryption Performance Results of 5 MB Data Set



	Time(sn	Ram(mb)	Cpu(%)
DNA	15,69	190,12	8,4
ECDH	0,15	6,38	4,74
AES	0,063	16,03	6,09

Figure 12. Decryption Performance Results of 5 MB Data Set

Based on these results, the advantages of the DNA-based cryptographic algorithm are listed below.

- **High Security:** DNA-based encryption provides high security due to the complexity and uniqueness of biological data. This is an important advantage, especially in protecting biometric data.
- **Resistance:** DNA-based encryption may be more resistant to traditional cryptographic attacks. This can increase data security. It can be used in situations with special security requirements.
- **Originality:** DNA-based encryption can offer unique solutions regarding the processing of biotechnological and biometric data. This increases its usability in specialized areas such as medical data, genetic information or biological research.

In addition, the disadvantages of the DNA-based cryptographic algorithm are given below.

- **High Resource Usage:** DNA-based encryption requires higher resources in terms of both RAM and CPU usage. This difference becomes more evident especially in large data sets.
- **Long Processing Time:** The encoding and decoding time is much longer compared to algorithms such as AES and ECDH. This can be a significant disadvantage in large data sets.
- **Complexity:** DNA-based encryption algorithms require more complex data structures and algorithms when processing biological data. This can create additional challenges in terms of implementation and management.

AES and ECDH algorithms are very advantageous, especially in large data sets, with their high efficiency, low resource usage and fast processing times. However, DNA-based encryption can be preferred in certain cases with its high security, authenticity and advantages in processing biometric data. For small to medium-sized data sets, gDNA encryption may be appropriate. It can be used in situations where security and authenticity requirements are high. In large data sets, performance and resource usage should be taken into consideration. More efficient algorithms such as AES or ECDH may be preferred. The advantages of DNA-based encryption may be important for certain security and biotechnological requirements. However, in general use, careful evaluation should be made in terms of performance and efficiency.

The appearance may differ when this application is run on different platforms. Therefore, it is possible to show different results depending on the factors. Some of these factors; It is the type of file, its volume, complexity, keyed or non-keyed structure, or software/hardware platform.

5. Result and Discussion

In this study, the textual data type called Siyâsetnâme (Nizamülmülk), which is a data set containing Turkish characters, was used. The original size of the dataset is 1.6 MB. Artificial data sets with different volumetric content were created from this data set. Sizes of the created data sets; they are 100 KB, 500 KB, 1 MB and 5 MB. Volumetric replication was performed for the 5 MB data set. Symmetric-key gDNA encryption was implemented on these artificial data sets. The secret key used is the 32-character (256 bit = 32 byte) "1086_NizamülmülkSiyasetname_1092" index in symmetric form. The purpose of the application is to test the suitability of using the gDNA encryption technique in cryptographic processes. For this reason, a gDNA crypto system has been designed. To determine the suitability of the proposed cryptosystem, existing/classical encryption algorithms were compared. AES, one of the symmetric algorithms, and ECDH, were used as the asymmetric encryption algorithm. The selection of these two classical algorithms is a reference to their most successful outcomes in previous studies. Performance metrics used for the application; time (s), memory (MB) and CPU (%) consumption rates. These criteria are widely accepted in the literature for cryptographic processes of textual data. When the results of the

application were analyzed, the results of the gDNA encryption technique were evaluated in 4 different data sets. When performance was compared with existing algorithms, it was seen that there was not much difference in small and medium volume data sets. However, it has been confirmed by the results that the range widens significantly in large volume data sets. In addition, it has been observed that the cryptosystem we propose gives the best results from similar or close studies in the literature. In conclusion, gDNA encryption techniques can be considered as one of the alternatives that can be used in cryptographic requirements. Future research may be directed towards accelerating the algorithms of genomics-based cryptosystems. Studies can be carried out to improve the algorithms of genomic cryptosystems in large volume data sets. Applications can be focused on video, images, signals or multi-format data sets, not just textual data types. Especially symmetrical, asymmetrical or etc. Hybrid studies of coding techniques may be at the forefront. Hybrid artificial intelligence architectures can be added, especially in the optimization part of genomic-based algorithms. It can be used to improve performance not only in cryptosystem applications but also in stages such as receiving, storing, sharing and storing data.

References

[1] Md M.A. Aziz, Md N. Sadat, D. Alhadidi, S. Wang, X. Jiang, C.L. Brown, N. Mohammed, "Privacy-preserving techniques of genomic data—a survey", *Briefings in Bioinformatics, vol. 20, 887–895,* 2019. https://doi.org/10.1093/bib/bbx139

[2] L. Bonomi, Y. Huang &, L. Ohno-Machado, "Privacy challenges and research opportunities for genomic data sharing", *Nat Genet 52, 646–654,* 2020. https://doi.org/10.1038/s41588-020-0651-0

[3] Z. Wan, J.W. Hazel, E.W. Clayton, et al., "Sociotechnical safeguards for genomic data privacy", *Nat Rev Genet 23, 429–445,* 2022. https://doi.org/10.1038/s41576-022-00455-y

[4] Y. Jiang, T. Shang, & J. Liu, "SM algorithms-based encryption scheme for large genomic data files", *Digital Communications and Networks, 7(4), 543-550,* 2021. https://doi.org/10.1016/j.dcan.2020.12.004

[5] H. Feistel, "Cryptography and computer privacy", *Scientific american, 228(5), 15-23,* 1973. https://www.jstor.org/stable/24923044

[6] G.J. Simmons, "Symmetric and asymmetric encryption", *ACM Computing Surveys (CSUR), 11(4), 305-330,* 1979. https://doi.org/10.1145/356789.356793

[7] P. Mahajan & A. Sachdeva, "A study of encryption algorithms AES, DES and RSA for security", *Global journal of computer science and technology, 13(15), 15-22,* 2013.

[8] H. Handschuh, L.R. Knudsen. M.J. Robshaw, 2001, "Analysis of SHA-1 in encryption mode*", In Cryptographers' Track at the RSA Conference, 70-83,* 2001. https://doi.org/10.1007/3-540-45353-9_7

[9] M. Kantarcioglu, W. Jiang, Y. Liu, B. Malin, "A cryptographic approach to securely share and query genomic sequence", *IEEE Transactions on information technology in biomedicine, 12(5), 606-617,* 2008. https://doi.org/10.1109/TITB.2007.908465

[10] S. Kalsi, H. Kaur & V. Chang, "DNA cryptography and deep learning using genetic algorithm with NW algorithm for key generation", *Journal of medical systems, 42, 1-12,* 2018. https://doi.org/10.1007/s10916-017-0851-z

[11] S. Basu, M. Karuppiah, M. Nasipuri, A. K. Halder, N. Radhakrishnan, "Bio-inspired cryptosystem with DNA cryptography and neural networks", *Journal of Systems Architecture, 94, 24-31,* 2019. https://doi.org/10.1016/j.sysarc.2019.02.005

[12] E. Şatir & O. Kendirli, "A symmetric DNA encryption process with a biotechnical hardware", *Journal of King Saud University-Science, 34(3),* 2018. https://doi.org/10.1016/j.jksus.2022.101838

[13] S. Namasudra, D. Devi, S. Kadry, R. Sundarasekar & A. Shanthini, "Towards DNA based data security in the cloud computing environment", *Computer Communications, 151, 539-547,* 2020. https://doi.org/10.1016/j.comcom.2019.12.041

[14] GZ Cui, "New direction of Data storage: DNA molecular storage technology", *Computer Engineering and Applications, vol. 42, 29-32,* 2006.

[15] S. Pramanik, S.K. Setua, "DNA cryptography", *2012 7th International Conference on Electrical and Computer Engineering, Dhaka, Bangladesh, 551-554,* 2012.https://doi.org/10.1109/ICECE.2012.6471609

[16] Türkiye Yazarlar Birliği (TYB). "nizamulmulk.pdf - TYB KİTAP". https://kitap.tyb.org.tr/kitap/nizamulmulk.pdf (Access Date: 29.10.2023).

[17] L M. Adleman, "Molecular computation of solutions to combinatorial problems", *Science, 266(5187), 1021-1024,* 1991. https://doi.org/10.1126/science.7973651

[18] R.J. Lipton, "DNA solution of hard computational problems", S*cience, 268(5210), 542-545,* 1995. https://doi.org/10.1126/science.7725098

[19] M. Sohal, S. Sharma, "BDNA-A DNA inspired symmetric key cryptographic technique to secure cloud computing", *Journal of King Saud University-Computer and Information Sciences, 34(1), 1417-1425,* 2018. https://doi.org/10.1016/j.jksuci.2018.09.024

[20] M.R. Biswas, K.M.R. Alam, S. Tamura, Y. Morimoto, "A technique for DNA cryptography based on dynamic mechanisms", *Journal of Information Security and Applications, 48, 102363,* 2019. https://doi.org/10.1016/j.jisa.2019.102363

[21] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, E. Barker, S. Leigh, S. Vo, "A statistical test suite for random and pseudorandom number generators for cryptographic applications". *US Department of Commerce, Technology*

Administration, National Institute of Standards and Technology, vol(21), 2001. https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=906762
[22] B.T. Hammad, A.M. Sagheer, I.T.Ahmed, N. Jamil,"A comparative review on symmetric and asymmetric DNA-based cryptography", *Bulletin of Electrical Engineering and Informatics, 9(6), 2484-2491,* 2020. https://doi.org/10.11591/eei.v9i6.2470
[23] M. Zengin, "Genetik kod yöntemiyle kriptoloji uygulaması", Yüksek Lisans Tezi (MSc Thesis), Karabük Üniversitesi, Karabük, Türkiye, 2021.
[24] F. Talo, "DNA tabanlı kriptoloji uygulaması", Yüksek Lisans Tezi (MSc Thesis), Düzce Üniversitesi, Düzce, Türkiye, 2021.
[25] A. Kaya, İ. Türkoğlu, "Yeni nesil güvenlik sistemleri: biyo-ilhamlı kriptografi", *2. International Uludağ Scientific Researches Congress, Bursa/Türkiye, 419-430,* 4-5 Kasım 2023.
[26] A. Kaya, İ. Türkoğlu, "Evaluation of symmetric cryptography algorithms in terms of performance analysis", *Cukurova 10th Internatıonal Scıentıfıc Researches Conference, Adana/Türkiye, 4048-462,* 2-4 April 2023.
[27] A. Kaya, İ. Türkoğlu, "Evaluation of asymmetric cryptography algorithms in terms of performance analysis", *4. International Cappadocia Scientific Research Congress, Nevşehir-/Türkiye, 1056-1070,* 16-17 April 2023.
[28] A. Kaya, İ. Türkoğlu, "Simetrik ve asimetrik şifreleme algoritmalarının performans karşılaştırılması", *Fırat Üniversitesi Müh. Bil. Dergisi, 35(2), 891-900,* 2023. https://doi.org/10.35234/fumbd.1296228
[29] R.M. Indrasena, K.A.P. Siva, R.K. Subba, "A secured cryptographic system based on dna and a hybrid key generation approach", *Biosystems, 197: 1-10,* 2020. https://doi.org/10.1016/j.biosystems.2020.104207
[30] G. Rahman, C.C. Wen, "Omega network pseudorandom key generation based on dna cryptography", *Applied Sciences, 12(16), 1-19,* 2022. https://doi.org/10.3390/app12168141
[31] SSL2, "Simetrik ve asimetrik şifreleme – farklar nelerdir?. https://www.ssl2buy.com/wiki/symmetric-vs-asymmetric-encryption-what-are-differences, (Access Date: 10.12.2023).
[32] H. Kodaz, F.M. Botsalı "Simetrik ve asimetrik şifreleme algoritmalarının karşılaştırılması", *Selçuk-Teknik Dergisi 9(1), 10-23,* 2010.
[33] Geeksforgeeks, "Advanced encryption standard (AES)" https://www.geeksforgeeks.org/advanced-encryption-standard-aes/ , (Access Date:10.11.2023).
[34] W. Diffie and M. Hellman, "New directions in cryptography", *in IEEE Transactions on Information Theory, vol. 22, no. 6,644-654,* 1976. https://doi.org/10.1145/3549993.3550007
[35] G.N. Krishnamurthy, V. Ramaswamy, "Encryption quality analysis and security evaluation of cast-128 algorithm and its modified version using digital images", *International Journal of Network Security & Its Applications, 1(1):28-33,* 2009. https://doi.org/10.48550/arXiv.1004.0571
[36] H. Wen, S. Yu & J. Lü, "Breaking an image encryption algorithm based on DNA encoding and spatiotemporal chaos", *Entropy, 21(3), 246,* 2019. https://doi.org/10.3390/e21030246
[37] M. Şahin, "Memristor-based hyperchaotic system and DNA encoding based image encryption application on LabVIEW", *Uluslararası Muhendislik Arastirma ve Gelistirme Dergisi, 15. 269-276,* 2023. https://doi.org/10.29137/umagd.1239725
[38] A. Arı"CDIEA: Chaos and dna based image encryption algorithm", *Turkish Journal of Science & Technology, 18(1), 261-273,* 2023. https://doi.org/10.55525/tjst.1250419

Author(s) Contributions
Alev Kaya: Methodology, software, writing, editing, material preparation, data collection and analysis
İbrahim Türkoğlu: Supervision, writing, reviewing, editing

Acknowledgments
Firat University Scientific Research Projects Coordination Unit (FÜBAP) supported this study with project number ADEP.22.06.

Conflict of Interest Notice
The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Ethical Approval and Informed Consent
It is declared that during the preparation process of this study, scientific and ethical principles were followed, and all the studies benefited from are stated in the bibliography.

Availability of data and material
[16] https://kitap.tyb.org.tr/kitap/nizamulmulk.pdf

Plagiarism Statement
This article has been scanned by iThenticate ™.

SAKARYA UNIVERSITY JOURNAL OF COMPUTER AND INFORMATION SCIENCES
http://saucis.sakarya.edu.tr/
e-ISSN: 2636-8129
Publisher: Sakarya University
Vol. 7, No. 3, 2024, 404-415
DOI: 10.35377/saucis...1544012
SAUCIS
SAKARYA UNIVERSITY
RESEARCH ARTICLE

TurkishLex:Development of a Context-Aware Spell Checker for Detecting and Correcting Spelling Errors in Turkish Texts

Pınar Savcı[1], **Bihter Daş**[2*]

[1] Arcelik A.Ş., Beyoglu, Istanbul, Türkiye
[2] Firat University, Technology Faculty, Department of Software Engineering, Elazığ, Türkiye

Corresponding author:
Bihter Daş, Fırat Üniversitesi, Elazığ, Türkiye,
bihterdas@firat.edu.tr

Article History:
Received: 5.09.2024
Accepted: 10.10.2024
Published Online: 10.12.2024

ABSTRACT
In Turkish, correct spelling correction is crucial for effective communication and preserving the integrity of written text. The challenge lies in the complexity of Turkish morphology and spelling, which can lead to frequent and diverse spelling errors. This study presents a spelling checker adapted for Turkish by creating a new Turkish dataset. The proposed spelling checker model effectively captures both minor and major textual changes and can detect the error. Our findings show that the proposed spelling checker system provides high accuracy and reliability with 98.21% accuracy performance with the Symspell module in correcting Turkish texts. This study provides valuable information about the strengths and weaknesses of existing spelling checkers and contributes to the improvement of spelling correction tools for Turkish.

Keywords: Spell checker, Spelling errors, Natural Language Processing (NLP), Spelling correction, Turkish texts

1. Introduction

A spell checker, also known as a spell corrector, is a tool that identifies and corrects misspellings in written text [1]. Spell checkers are essential components in natural language processing (NLP) applications, ranging from word processors to more complex systems like search engines, chatbots, and text analysis tools. They help maintain the quality and consistency of textual data, which is crucial for downstream NLP tasks such as sentiment analysis, machine translation, and information retrieval [3-4]. However, creating an effective spell checker for a morphologically rich and agglutinative language like Turkish presents unique challenges. In Turkish, words are formed through the extensive use of suffixes, leading to a vast number of possible word forms that are not common in languages like English. This complexity makes the detection and correction of spelling errors more difficult, especially when existing tools fail to capture the context or morphological structure [5].

The development of a robust spell corrector for Turkish is not only important for improving text quality but also for enabling more accurate NLP applications specific to the language [6]. Despite the growing interest in Turkish NLP, there is still a notable gap in high-quality, context-aware spell correction models tailored to the language's unique characteristics. Existing models often struggle with correctly identifying and suggesting replacements for misspelled words, particularly in cases where the errors involve common suffixes or compound word formations. This study aims to fill this gap by introducing a contextually aware spell checker specifically designed for Turkish texts, integrating both traditional linguistic rules and modern machine-learning techniques [7].

This study aims to enhance the accuracy and usability of Turkish spell-checking tools, making them more applicable across various NLP applications, including automated content moderation, document editing, and educational tools. The proposed model not only improves error detection rates but also enhances contextually relevant corrections by addressing both syntactic and semantic aspects of Turkish text. Specifically, the model's ability to capture semantic nuances plays a crucial role in improving the interpretation of idiomatic expressions, context-dependent variations, and multi-word constructions. This contribution significantly impacts tasks such as sentiment analysis and text classification, where understanding the meaning beyond individual words is essential. In cases involving proverbs or idioms, the model performs semantic disambiguation, which contributes to more accurate corrections. This work not only contributes to the development of a practical solution but also advances the broader field of Turkish NLP by introducing methodologies that can be extended to similar languages.

Cite as: P. Savcı, B. Daş "TurkishLex:Development of a Context-Aware Spell Checker for Detecting and Correcting Spelling Errors in Turkish Texts," *Sakarya University Journal of Computer and Information Sciences*, vol.7, no. 3, pp. 404-415, 2024. Doi: 10.35377/saucis…1544012

1.1. Problem Statement and Motivation

In natural language processing (NLP), the accurate representation and correction of textual data are critical challenges, particularly in the context of spelling errors, variant generation, and dictionary-based corrections. Text data often contain inconsistencies, misspellings, and variations that can significantly affect downstream tasks such as text classification, machine translation, and sentiment analysis. Traditional approaches to error detection and correction rely heavily on static dictionaries, which struggle with evolving language use, context-specific variations, and complex word forms. The challenge of developing an effective spell corrector for Turkish lies in the language's agglutinative nature and morphological richness. Existing tools cannot often handle complex word formations and context-specific errors, leading to inaccurate or incomplete corrections.

This study addresses these challenges for the Turkish language by proposing a method combining text data processing, cleaning, and soft and hard matching techniques to generate comprehensive variants. By introducing advanced dictionary validation and correction processes, this research aims to enhance the robustness and accuracy of NLP applications. In addition, a new Turkish dataset is introduced to the literature in the study. The motivation behind this work stems from the increasing need for flexible and adaptive error correction mechanisms in systems that handle large-scale, noisy text data, where traditional methods may fall short.

1.2. Main Contributions

This paper offers several key contributions to the field of text processing and dictionary-based error correction:

- Creating a new Turkish dataset named "TurkishLex" collected from electronic books containing emotions, idioms, proverbs, and local dialects.
- The development of a spell checker tailored specifically for the Turkish language.
- A structured approach for text data cleaning and preparation, addressing common challenges such as misspellings, punctuation inconsistencies, and context-dependent variations.
- The introduction of a hybrid matching system that effectively captures both minor and significant text variations, allowing for improved variant generation and error detection.

1.3. Related Works

The challenge of addressing spelling errors has been extensively studied, leading to the development of various techniques and algorithms. Early studies, such as those by [8], classified spelling errors into two main types: lexical errors and grammatical errors. Lexical errors, also known as usage errors, involve mistakes within words themselves, irrespective of their grammatical context. Grammatical errors, on the other hand, are morphosyntactic errors that include mistakes related to word combinations or grammatical modifications, such as conjugations and declensions [9]. In the studies conducted, in dictionary-based systems, the detector classifies whether a word is incorrect by searching the dictionary. However, since these systems are based on pre-prepared dictionaries, they can only detect words that are not in the dictionary [10,11]. In various studies, more sophisticated models have been introduced to improve error detection [12,13]. During the error correction process, one or more tokens are chosen for the identified errors. In dictionary-driven systems, the corrector recommends words based on spelling resemblance, but the performance of these systems declines when context is ignored. Language models (LM) have been suggested to generate corrections that account for context.

Recent studies have further advanced the approaches to spelling error correction by integrating contextual information into more sophisticated language models and neural networks. These models leverage large datasets and deep learning techniques to achieve higher accuracy in detecting and correcting errors. For instance, transformer-based architectures, such as BERT, have shown promising results in handling both lexical and grammatical errors by using contextual embeddings to predict the most probable correction within a sentence [19]. Furthermore, the integration of attention mechanisms in neural networks has significantly improved error correction by enabling the model to focus on relevant parts of the input during both the detection and correction phases [20].

Other studies have used spelling checkers, which generally use general algorithms to detect errors involving transposition of two adjacent letters, missing or extra letters [14,15]. Especially in languages such as Japanese, non-native English users make unique spelling errors due to differences in the phonetic structure of the language [11]. For example, since Japanese does not have the sounds /th/ or /v/ in English, errors such as "thunderstorm" instead of "sanderstorm" occur as a result of incorrect perception of these sounds. Differences in phoneme sequence and morphological errors also lead to such errors. Additionally, phonological errors have been addressed in recent works, especially in languages with significant phonetic deviations, such as Japanese and Chinese, by incorporating language-specific phoneme models into spelling correction systems [21]. These models improve the accuracy of corrections by accounting for the phonetic discrepancies between the native and target languages. The literature on English grammar error correction has developed rapidly based on machine translation techniques. The transition from statistical methods to neural network-based approaches has provided significant progress in the field of error correction [16,17]. In particular, the methods used in the enrichment of data and

training processes increase the performance of these systems [18]. Moreover, recent advances in data augmentation techniques and adversarial training have further improved the robustness and generalization of these models in diverse spelling error correction tasks across multiple languages [22].

The rest of the paper is organized as follows: Section 2 presents the sub-steps required for dictionary creation. Section 3 details the proposed methodology, including the dataset, preprocessing, error detection, and spell corrector modules. Section 4 presents the experimental results and discussion. Section 5 represents the conclusion of the paper.

2. Dictionary Creation

In spell checkers, the dictionary is one of the cornerstones of the system [18]. The dictionary acts as a reference source to check the accuracy of written texts. In order to determine whether a word is correct, it is checked whether the word is in the dictionary [23]. Therefore, creating an accurate and comprehensive dictionary is critical to the success of the spell checker. In languages with complex grammatical features such as Turkish, such dictionaries need to be created especially carefully. In this study, a new spell checker specialized for Turkish is presented and a comprehensive dictionary is created for this checker. Unlike the studies in the literature, this dictionary was developed by taking into account the original language structure of Turkish, letter changes and word variants. Figure 1 shows sub-steps of the dictionary creation.

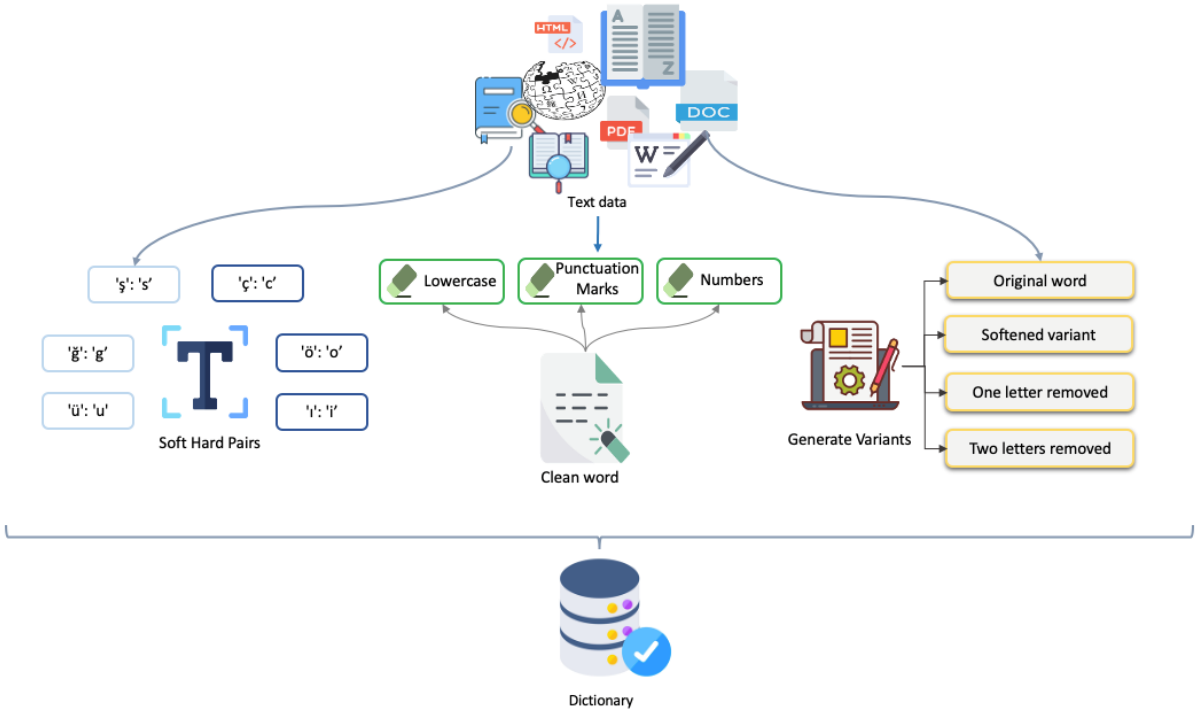


Figure 1. Steps of the Dictionary Creation

2.1. Text Data Processing

The first stage is the processing of text data received from various sources. This data can be in different formats (HTML, PDF, DOC, etc.). The system takes this data and makes it analyzable by subjecting it to cleaning and normalization stages. The cleaned and normalized data is the basis for creating word variants.

2.2. Cleaning Stage (Clean Word)

Various cleaning operations are performed on the text data:

Lowercase: All letters are converted to lowercase. This helps the spell checker to ignore errors caused by uppercase/lowercase differences.

Punctuation Marks: Punctuation marks in the text are removed. This allows the focus to be on the meaning of the words only.

Numbers: Numbers found in the text are removed. Since the spell checker usually only deals with words, the presence of numbers may be unnecessary. The "clean word" obtained after these cleaning operations is used to create variants.

2.3. Making Soft-Hard Pairs

In this step, soft and hard letter changes, which are common in languages such as Turkish, are taken into account. Some letters in Turkish (e.g. 'ğ', 'ç', 'ş', 'ı', 'ö', 'ü') can be written differently in certain situations (e.g. 'ğ' -> 'g', 'ç' -> 'c'). Such letter changes are important to make it easier for the spell checker to match an incorrect word with the correct word.

2.4. Generate Variants

During the variant generation process, different types of variations on incorrect words are generated. These variants are important for spelling checker algorithms to detect misspelled words and convert them into correct words. Table 1 shows how these variants are generated and examples for each type of error.

Table 1. Types of Spelling Mistakes and their Correct Forms

Type of mistake	Misspell sentence	Correct sentence
Soft Hard Pairs characters	yogurt	yoğurt
Missing characters	yğurt	yoğurt
Extra characters	yoğurrt	yoğurt
Wrong characters	yourt	yoğurt
Shuffled characters	yoğutr	yoğurt

Soft Hard Pairs characters: Soft (ğ, ş, ç) and hard (g, s, c) letter changes, which are frequently encountered in Turkish, are important for producing variants that will match the correct form of the word. For example, writing the word "yoğurt" as "yogurt" is an error that can occur due to soft-hard letter changes.

Missing characters: The situation where one or more letters are written missing in a word. In the example "yğurt", a letter is missing from the word "yoğurt".

Extra characters: Represents errors caused by adding one or more extra letters to the word. In the example "yoğurrt", an extra letter "r" has been added to the word "yoğurt".

Wrong characters: The situation where an incorrect character is used instead of the correct character in the word. In the example "yourt", "u" is used instead of the letter "ğ" in the word "yoğurt".

Shuffled characters: Shows errors caused by changing the places of the characters in the word. In the example of "yoğutr", the characters in the word "yoğurt" are swapped.

Our goal in creating these variants is to help the spelling checker algorithm determine which word is closest to the correct word when it encounters misspelled words, using metrics like edit distance. This makes the system more effective at correcting misspelled words.

At the end of this process, all variants created were stored in a dictionary, then this data was converted to a pandas DataFrame and made suitable for analysis and saving. The results were saved as a .txt file, making them available for analysis in areas such as text processing and error detection. These methods offer a flexible and useful approach for detecting and correcting spelling errors in the field of language processing and for analyzing various language variants.

3. Materials and Method

In this section, unlike other studies, the steps of a new spell checker model that creates variants of words in texts and analyzes these variants to correct spelling errors in Turkish data are explained. This model integrates advanced linguistic analysis techniques, making it a unique contribution to the field of natural language processing. This model presents a new spelling checker specifically designed for Turkish, taking into account the unique characteristics and complexities of the language. Unlike generic spell checkers, this tool is tailored to handle the agglutinative nature of Turkish, where suffixes can significantly alter the meaning and grammatical function of a word. This improved checker detects and corrects errors specific to the Turkish language structure more effectively. For instance, it considers vowel harmony and consonant mutation, which are critical aspects of Turkish phonology, enabling the detection of errors that might be overlooked by traditional methods. The datasets, processing techniques, and spelling check algorithms used in the study are explained in detail below. Additionally, this section will outline the theoretical framework underpinning the spell checker's design, providing insights into the linguistic principles that inform its functionality. The significance of this study lies in its potential applications, ranging from educational tools for language learners to enhancing the efficiency of text processing in various digital platforms. By addressing the intricacies of the Turkish language, this model not only contributes to the advancement of computational linguistics but also provides a robust resource for researchers and developers working in the field. The subsequent sections will delve into the methodological aspects of the model's development, including data collection strategies, preprocessing techniques, and the rationale behind the chosen algorithms. Figure 2 shows the block diagram of the Turkish spell checker model.

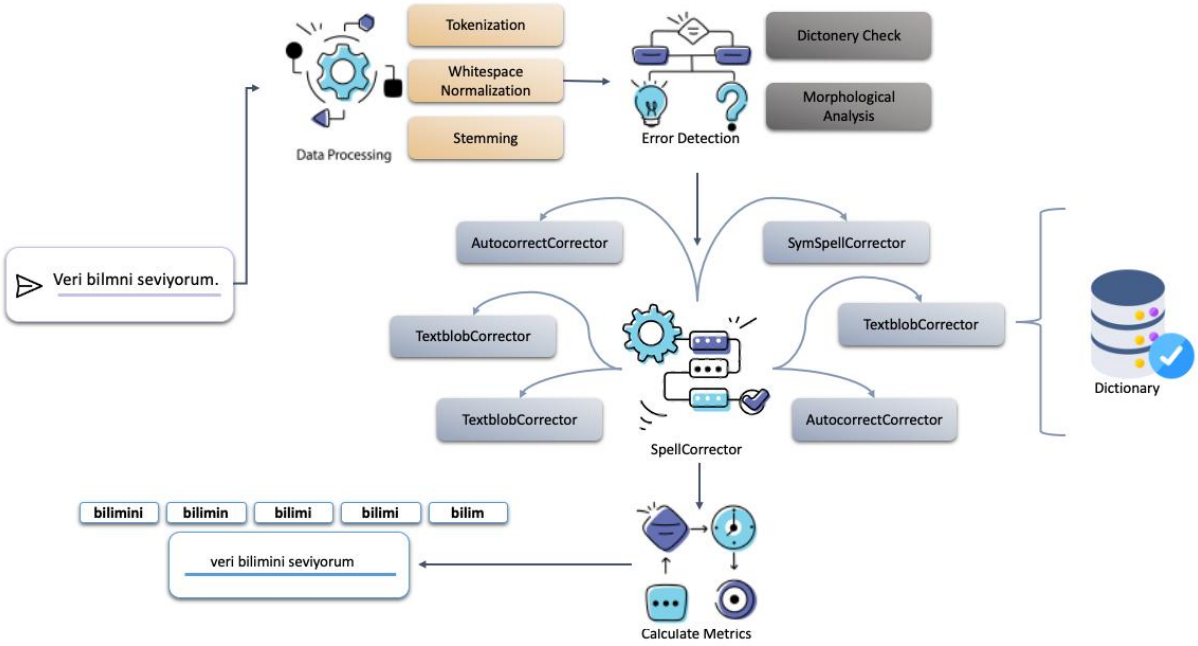


Figure 2. The Block Diagram of the Turkish Spell Checker Model

3.1. Data Collection

In this study, a new dataset named "TurkishLex" was developed, which contains diverse linguistic elements, including idioms, proverbs, regional dialects, and common expressions. TurkishLex is specifically designed to support tasks that require an understanding of both syntactic structure and semantic content in Turkish. Throughout the rest of this article, we will refer to this dataset as TurkishLex. The created dataset was collected from sites (openlibrary, archive, Gutenberg, books.google, manybooks) that include traditional printed electronic books. More than 4,800 books were used to create the dataset, resulting in a total of 3,913,362 new words. These words were derived from the content of the books in question, and each provides important examples for analyzing spelling errors. The derived words were enriched with variations to show how different spelling errors look and which correct word they are closest to. The created dataset has been uploaded to the Hugging Face platform and is available for researchers and developers. The following link can be used to access the dataset: https://huggingface.co/datasets/pnr-svc/spellchecker-dataset

3.2. Data Processing

This stage includes a series of preprocessing steps for processing text data:

Tokenization: First, the tokenization process is performed; in this step, the text is divided into smaller units such as words or sentences so that each word can be analyzed independently.

Whitespace Normalization: By applying the normalization process, unnecessary spaces in the text are standardized. This step helps to detect spelling errors more accurately.

Stemming: Words are reduced to their roots with the stemming process. This process is especially critical for the normalization of grammatical variations; for example, reducing the word "bilimini" to the root "bilim" makes it easier for the spelling checker to find the correct word. These preprocessing steps increase the efficiency of error detection and correction processes by providing a homogeneous structure to the data set.

3.3. Error Detection

In the error detection phase, potential spelling errors are determined on the text data. This process is carried out with two main methods: First, Dictionary Check is used to check whether each word in the text is in the dictionary. When the word is not in the dictionary, this indicates a potential error. Second, Morphological Analysis analyzes the structure and suffixes of the word. This step is especially important for detecting grammatical errors because some words can be used with incorrect suffixes even if they are spelled correctly. These methods come together to provide a high accuracy rate in detecting spelling errors.

3.4. Spell Corrector Modules

Various spelling correction algorithms are used to correct the detected errors. These algorithms offer the most appropriate correction suggestions using different methods. AutocorrectCorrector is a simple automatic correction algorithm for quickly correcting common errors. Symspell is an algorithm that produces high-performance results on large data sets and processes

a large vocabulary with low memory usage [24]. The TextblobCorrector module uses the TextBlob library to perform grammar and word prediction and works with a simple language model. Finally, the HunspellCorrector module uses the Hunspell library to perform grammatical and word-based corrections; this feature is especially important in languages with complex morphology such as Turkish. The combination of these modules ensures that spelling errors are corrected accurately and effectively [25]. Other spell corrector algorithms are:

Jamspell is an algorithm developed for spell checking, which provides more accurate corrections by taking context into account. It corrects misspellings by inferring meaning from surrounding words. This algorithm works at high speed and is capable of processing approximately 5,000 words per second. For improved accuracy, Catboost uses a gradient-assisted decision tree model to rank candidate corrections and split concatenated words [26].

Hunspell is a powerful spelling checker library, especially for open-source spellers and grammar checkers. It offers multilingual support and is especially suitable for complex agglutinative languages (Turkish, Hungarian, etc.). Flexible grammar rules can perform root detection and affix removal operations, thus providing effective spelling corrections with a wide dictionary and language support[25].

Symspell is a fast and lightweight spelling check algorithm. This algorithm, which was developed specifically for performance, offers faster correction suggestions while using less memory compared to other approaches. It produces very fast results by correcting misspelled words with minimum editing distance. It is preferred in large volume data sets due to its high accuracy rate and memory efficiency [26].

Autocorrector is a simple and effective automatic correction tool. This system, which is usually used to correct common spelling errors, detects and corrects spelling errors at the word level. It is easy to use and offers a fast correction for large volumes of text. It generally works with a very simple and rule-based approach [27].

Pyspellchecker is a simple spelling checker library written in Python. It detects missing or misspelled words by considering words individually and offers possible corrections. It does not take into account context, so it works only on a word basis, but it attracts attention with its high speed and ease of use. It can be extended based on language models [25].

TextBlob is a simple library for text processing and natural language processing (NLP) tasks for Python. It supports tasks such as sentiment analysis, language detection, language translation, sentence/paragraph parsing and spell checking. It uses basic rule-based methods when doing spelling check and automatic correction, but it can also be used in more complex NLP tasks[28].

4. Experimental Results and Discussion

In this section, we present and analyze the experimental results obtained from evaluating the proposed Turkish spell checker model. The performance of the model was assessed using several key metrics: recall, precision, identifying accuracy, the percentage of invalid words remaining after the checker, the percentage of correctly fixed misspellings, the percentage of misspellings not fixed but with the correct suggestion within the top 5 candidates, and the percentage of valid words incorrectly altered by the spell checker. These metrics provide a comprehensive understanding of the model's effectiveness in both identifying and correcting spelling errors, as well as its impact on valid words. The performance metrics used are as follows:

Accuracy: Identifying accuracy refers to the ratio of correct decisions made by the spell checker (the sum of true positives and true negatives) to the total number of decisions made (the sum of all true and false positives and negatives). It is shown in Equation 1. This metric provides a comprehensive overview of the spell checker's competency by determining how accurately it performs the task assigned. The closer this accuracy is to 100%, the better the overall performance of the spell checker.

$$Identified\ accuracy = \frac{(tp+tn)}{(tp+fp+fn+fp)} \quad (1)$$

Recall: It is defined (Equation 2) as the ratio of the number of invalid words correctly identified by the spell checker as misspelled (true positives) to the total number of invalid words in the text (the sum of true positives and false negatives). The ideal scenario for a spell checker is to recognize all invalid words as misspelled, thereby achieving as high a recall as possible, ideally approaching 100%. Recall provides an indication of the comprehensiveness of the spell checker's dictionary and its ability to detect incorrect words.

$$recall = \frac{tp}{(tp+fn)} \quad (2)$$

Precision: It is the ratio of the number of words that are correctly identified as misspelled by the spell checker (true positives) to the total number of words flagged as misspelled by the spell checker (the sum of true positives and false positives) (Equation 3). The optimal condition for a spell checker is to correctly identify all and only the invalid words as misspelled, thereby achieving as high a precision as possible, ideally approaching 100%. Precision reflects the accuracy of the spell checker in flagging words as misspelled.

$$precision = \frac{tp}{(tp+fp)} \quad (3)$$

Percentage of Correctly Fixed Misspellings: This metric refers to the percentage of misspelled words that are correctly corrected by the spell checker (Equation 4). For a spell checker, the ideal value is 100%. The higher this percentage, the better the performance of the spell checker.

$$fixedMisspellingsPerc = \frac{fixedMisspellings}{allMisspellsInText} \quad (4)$$

Non-fixed with correction in top-5: This metric calculates the percentage of uncorrected misspellings where the correct correction is among the top 5 suggestions provided by the spell checker (Equation 5). The ideal value for a spell checker is 100%, indicating that for every uncorrected misspelling, the correct suggestion is among the top 5 candidates. Higher percentages are preferable.

$$\text{notFixedButInTop5Perc} = \frac{\text{notFixedButCorrectionInTop5Candidates}}{\text{notFixedMisspells}} \quad (5)$$

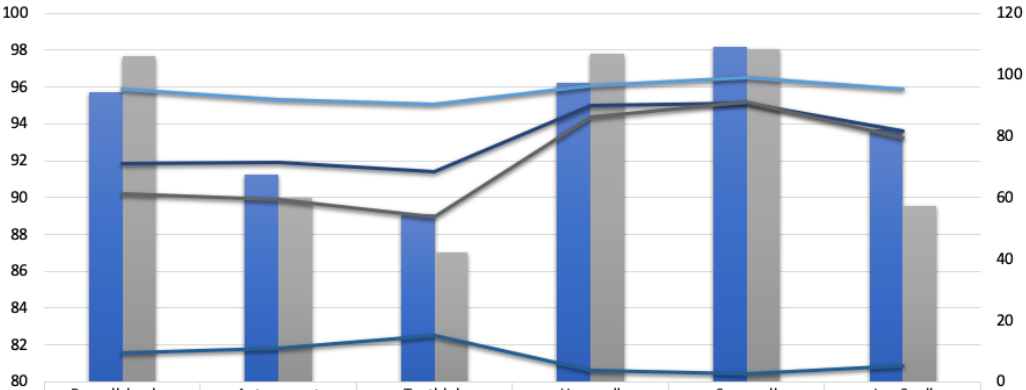
Percentage of Valid Words Disrupted: This metric measures the percentage of originally valid words that are incorrectly altered by the spell checker (Equation 6). The ideal value for a spell checker is 0%, indicating that no valid words have been incorrectly altered. Lower percentages are better.

$$\text{brokenValidsPerc} = \frac{\text{brokenValids}}{\text{allOriginallyValids}} \quad (6)$$

The spell checker models were tested against a newly created Turkish dataset containing a mixture of correctly and incorrectly spelled words. The dataset was carefully curated to reflect the challenges inherent in Turkish, such as its agglutinative nature and complex morphological structure. Each model's performance was evaluated based on its ability to correctly identify and fix spelling errors while minimizing the disruption to valid words. Table 2 shows results of Turkish spell checker modules for performance metrics

Table 2. The Performance Results of the Turkish Spell Checker Model for Modules

	Accuracy	**Recall**	**Precision**	**Fixed**	**Non-fixed with correction in top-5**	**Broken**
Pyspellchecker	95.72	97.71	95.24	70.93	61.2	9.33
Autocorrect	91.23	90.02	92.09	71.54	59.32	10.93
Textblob	89.12	87.04	90.32	68.44	53.81	1523
Hunspell	96.21	97.84	96.43	89.90	86.39	3.91
Symspell	98.21	98.03	99.03	90.85	91.05	2.74
JamSpell	93.72	89.57	95.22	81.63	79.47	5.36



	Pyspellchecker	Autocorrect	Textblob	Hunspell	Symspell	JamSpell
Accuracy	95,72	91,23	89,12	96,21	98,21	93,72
Recall	97,71	90,02	87,04	97,84	98,03	89,57
Precision	95,24	92,09	90,32	96,43	99,03	95,22
Fixed	70,93	71,54	68,44	89,9	90,85	81,63
Non-fixed with correction in top-5	61,2	59,32	53,81	86,39	91,05	79,47
Broken	9,33	10,93	15,23	3,91	2,74	5,36

Figure 3. Graphical Representation of Performance Results

The results, as summarized in Figure 3, indicate varying degrees of success across the different models. SymSpell emerged as the top-performing model, achieving an accuracy of 98.21%, a recall of 98.03%, and a precision of 99.03%. This model also excelled in fixing misspellings, with 90.85% of errors corrected, and maintaining a low percentage of broken valid words at just 2.74%. On the other hand, Pyspellchecker also performed well, with an accuracy of 95.72% and a recall of 97.71%. However, its precision was slightly lower at 95.24%, and it had a higher percentage of broken valid words (9.33%) compared to SymSpell. Hunspell, another widely used spell checker, demonstrated strong performance with a precision of 96.43% and an impressive recall of 97.84%. It also maintained a relatively low percentage of broken valid words (3.91%). However, its accuracy (96.21%) and the percentage of correctly fixed misspellings (89.9%) were slightly lower than those of SymSpell. Autocorrect and TextBlob, while still effective, showed lower performance overall, particularly in terms of recall and the percentage of broken valid words, with TextBlob having the highest percentage of broken valid words at 15.23%. JamSpell, while achieving a decent overall performance with an accuracy of 93.72% and a precision of 95.22%, struggled with recall (89.57%) and had a higher percentage of broken valid words (5.36%) compared to SymSpell and Hunspell.

The comparative analysis reveals that SymSpell outperforms the other models in most of the key metrics, particularly in precision and the ability to fix misspellings. Its low percentage of broken valid words further underscores its suitability for applications where maintaining the integrity of valid text is critical. Pyspellchecker and Hunspell also demonstrate strong performances, making them viable alternatives, especially in scenarios where slightly lower precision and accuracy can be tolerated. Autocorrect and TextBlob, while less effective in this context, may still be valuable in specific use cases where the simplicity of implementation and general-purpose spell checking are more important than absolute precision. JamSpell, although performing well in terms of precision, shows limitations in recall and the preservation of valid words, suggesting that it may be more suitable for environments where precision is prioritized over recall. The percentage of non-fixed misspellings that have the correct suggestion within the top 5 candidates is an important metric for understanding the models' ability to offer useful alternatives when the top suggestion is incorrect. SymSpell again leads in this area, with 91.05%, indicating its robustness in providing accurate suggestions even when the first choice is not correct. In addition to the overall improvements in spelling correction, the proposed model was evaluated for its impact on idioms and proverbs. The semantic analysis component was particularly effective in identifying and correcting common idiomatic expressions. For instance, idiomatic phrases that do not follow the standard syntactic rules, such as ‘atı alan Üsküdar’ı geçti’ (a well-known Turkish idiom), were correctly identified and preserved in their semantic integrity. This demonstrates the model’s capacity to perform corrections without losing meaning in idiomatic contexts. The semantic disambiguation, achieved through the use of contextual embeddings, allows the system to retain the intended meaning of idioms and proverbs.

Confusion matrices are essential for evaluating classification models' performance, as they provide detailed insights into how well a model can distinguish between different classes. In this study, we also present the confusion matrix for the proposed SymSpell algorithm applied to Turkish place names, as shown in Figure 4. The matrix reveals the distribution of predictions across various true labels, indicating both correct classifications and misclassification. From the matrix, it is clear that the SymSpell model demonstrates strong performance for certain classes, such as "Taşköprü" and "Yenipazar," with a high accuracy rate of 1.00 in these categories. However, there are instances where the model struggles, such as with "Toprak" and "Körfez," showing misclassifications across different categories. Notably, the model has difficulty distinguishing "Taşkent"

and "Boztepe," with significant confusion between these classes. Additionally, the off-diagonal values suggest that the algorithm tends to incorrectly predict certain classes, such as confusion between "Toprak" and "Köprüköy," highlighting areas where further refinement could improve performance. The proposed model's ability to handle these misclassifications is critical, especially in the context of Turkish spelling variations, where small errors can significantly affect the outcome.

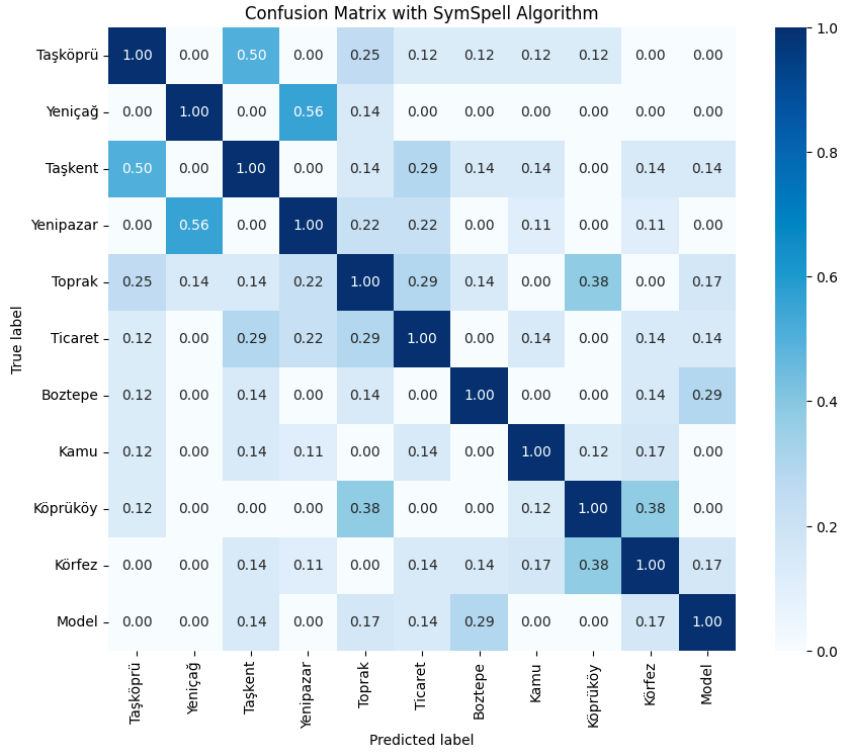


Figure 4. Confusion Matrix of the Proposed SymSpell Algorithm on Turkish labels

The comparative analysis presented in Table 3 highlights the significant advancements achieved by the proposed model (TurkishLex) over existing Turkish spell-checker studies. With an accuracy of 98.21%, the hybrid approach combining rule-based methods with contextual semantic analysis outperforms other models, particularly in handling idiomatic expressions and proverbs. Necva Bölücü's 2019 model, despite utilizing a large dataset and incorporating contextual analysis through a Noisy Channel and HMM approach, achieved only 57.50% accuracy. Aydoğan et al. (2020) and Osman Büyük et al. (2020) demonstrated improvements with their dictionary-based LSTM method (85.80%) and Seq2Seq model with context (92.30%), respectively, but neither fully addressed the complexities of Turkish semantics and idiomatic variations. The proposed model's ability to integrate semantic context into error correction, especially for idiomatic expressions, represents a key advancement in Turkish NLP, offering superior accuracy and applicability across diverse text types.

Table 3. Comparison of Turkish Spell Checker Models

Study	**Approach**	**Dataset size**	**Accuracy**	**Methodology**	**Key Improvement**
Bölücü et al. (2019) [29]	Noisy Channel + HMM	423M words (BOUN)	57.50	SMM with contextual analysis	Improved over Zemberek by 9%
Aydoğan et al. (2020) [30]	Dictionary Method + LSTM	10.5B words	85.80	Word2Vec + LSTM for word prediction	Accuracy increased by 8.68%
Büyük et al. (2020) [31]	Seq2Seq with Context	4M sentences	92.30	Seq2Seq with 3-character context	Context improved accuracy by 9.2%
Proposed Model (TurkishLex)	Hybrid (Rule-based + Semantic)	3.9M words	98.21	SymSpell + Contextual Embeddings	Handles idioms/proverbs effectively

The scalability of the proposed model was also analyzed by testing it on datasets of increasing size. As the text size increased, the response speed showed a linear complexity, maintaining consistent accuracy across the expanded datasets. Specifically, the time complexity was measured as O(n), where 'n' represents the number of words in the text. The accuracy remained within 1% of its peak performance for text sizes ranging from 10,000 to 100,000 words, demonstrating the robustness of the model. This consistency in accuracy rates suggests that the model is both scalable and efficient for large-scale text-processing tasks.

5. Conclusion

This study, the evaluation of spell checkers using the outlined metrics provides a detailed understanding of their effectiveness in identifying and correcting misspellings in Turkish texts. The study introduces both a new Turkish dataset and a spell corrector model that can be used in Turkish texts. The obtained performance metrics prove that the proposed Turkish spell checker model performs very well. Achieving optimal values for these metrics is essential for enhancing the performance of spell checkers and ensuring their effectiveness in diverse linguistic contexts. This paper will guide future improvements in spell-checking models and inform their deployment in real-world NLP tasks. In future work it may be focused on refining these metrics and exploring additional factors that influence spell-checking accuracy to further improve tool performance

Acknowledgment

This work is supported by the Republic of Turkey, Ministry of Science, Technology and Industry project named "AI Based Smart Digital Assistant Customer Dialog Bot project" and project code AR-22-087-0001. It is funded by R&D project within the scope of law 5746 by the Arçelik Digital Transformation, Big Data and Artificial Intelligence R&D Center.

References

[1] Y. Chaabi and F. Ataa Allah, “Amazigh spell checker using Damerau-Levenshtein algorithm and N-gram,” *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 8, Part B, pp. 6116–6124, Sep. 2022, doi: 10.1016/j.jksuci.2021.07.015.

[2] V. J. Hodge and J. Austin, “A comparison of a novel neural spell checker and standard spell checking algorithms,” *Pattern Recognition*, vol. 35, no. 11, pp. 2571–2580, Nov. 2002, doi: 10.1016/S0031-3203(01)00174-1.

[3] R. Garfinkel, E. Fernandez, and R. Gopal, “Design of an interactive spell checker: Optimizing the list of offered words,” *Decision Support Systems*, vol. 35, no. 3, pp. 385–397, Jun. 2003, doi: 10.1016/S0167-9236(02)00115-X.

[4] M. Nejja and A. Yousfi, “The Context in Automatic Spell Correction,” *Procedia Computer Science*, vol. 73, pp. 109–114, Jan. 2015, doi: 10.1016/j.procs.2015.12.055.

[5] K. Sarıtaş, C. A. Öz, and T. Güngör, “A comprehensive analysis of static word embeddings for Turkish,” *Expert Systems with Applications*, vol. 252, p. 124123, Oct. 2024, doi: 10.1016/j.eswa.2024.124123.

[6] S. Demir and B. Topcu, “Graph-based Turkish text normalization and its impact on noisy text processing,” *Engineering Science and Technology, an International Journal*, vol. 35, p. 101192, Nov. 2022, doi: 10.1016/j.jestch.2022.101192.

[7] Y. B. Kaya and A. C. Tantuğ, “Effect of tokenization granularity for Turkish large language models,” *Intelligent Systems with Applications*, vol. 21, p. 200335, Mar. 2024, doi: 10.1016/j.iswa.2024.200335.

[8] Kukich K. Techniques for automatically correcting words in text. ACM computing surveys (CSUR). 1992 Dec 1;24(4):377-439.

[9] P. T. Hacken and C. Tschichold, “Word Manager and CALL: Structured access to the lexicon as a tool for enriching learners’ vocabulary,” *ReCALL*, vol. 13, no. 1, pp. 121–131, May 2001, doi: 10.1017/S0958344001001112.

[10] W. Phatthiyaphaibun, K. Chaovavanich, C. Polpanumas, A. Suriyawongkul, L. Lowphansirikul, and P. Chormai, *PyThaiNLP: Thai Natural Language Processing in Python*. (Jun. 2024). Python. Accessed: Aug. 27, 2024. [Online]. Available: https://github.com/PyThaiNLP/pythainlp

[11] *hunspell/hunspell*. (Aug. 27, 2024). C++. hunspell. Accessed: Aug. 27, 2024. [Online]. Available: https://github.com/hunspell/hunspell

[12] A. Lertpiya, T. Chaiwachirasak, N. Maharattanamalai, T. Lapjaturapit, T. Chalothorn, N. Tirasaroj, et al., "A preliminary study on fundamental Thai NLP tasks for user-generated Web content", *Proc. Int. Joint Symp. Artif. Intell. Natural Lang. Process. (iSAI-NLP)*, pp. 1-8, Nov. 2018.

[13] S. Watcharabutsarakham, "Spell checker for Thai document", *Proc. IEEE Region Conf.*, pp. 1-4, Nov. 2005.

[14] M. Rodphon, K. Siriboon and B. Kruatrachue, "Thai OCR error correction using token passing algorithm", *Proc. IEEE Pacific Rim Conf. Commun. Comput. Signal Process.*, pp. 599-602, 2001.

[15] B. Kruatrachue, K. Somguntar and K. Siriboon, "Thai OCR error correction using genetic algorithm", *Proc. 1st Int. Symp. Cyber Worlds*, pp. 137-141, 2002.

[16] H. T. Ng, S. M. Wu, T. Briscoe, C. Hadiwinoto, R. H. Susanto and C. Bryant, "The CoNLL-2014 shared task on grammatical error correction", *Proc. 18th Conf. Comput. Natural Lang. Learn. Shared Task*, pp. 1-14, 2014.

[17] A. Rozovskaya and D. Roth, "Grammatical error correction: Machine translation and classifiers", *Proc. 54th Annu. Meeting Assoc. Comput. Linguistic*, pp. 2205-2215, Aug. 2016, [online] Available: https://www.aclweb.org/anthology/P16-1208.

[18] M. Junczys-Dowmunt and R. Grundkiewicz, "Phrase-based Machine Translation is State-of-the-Art for Automatic Grammatical Error Correction", *Proc. Conf. Empirical Methods Natural Lang. Process.*, pp. 1546-1556, Nov. 2016, [online] Available: https://www.aclweb.org/anthology/D16-1161.

[19] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

[20] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. In *Advances in Neural Information Processing Systems* (pp. 5998-6008).

[21] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019). RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*.

[22] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative adversarial nets. In *Advances in Neural Information Processing Systems* (pp. 2672-2680).

[23] D. N. Mati, M. Hamiti, B. Selimi and J. Ajdari, "Building Spell-Check Dictionary for Low-Resource Language by Comparing Word Usage," *2021 44th International Convention on Information, Communication and Electronic Technology (MIPRO)*, Opatija, Croatia, 2021, pp. 229-236, doi: 10.23919/MIPRO52101.2021.9597183.

[24] A. Kicsi, K. Szabó Ledenyi, and L. Vidács, "Radiologic text correction for better machine understanding," *Engineering Reports*, vol. n/a, no. n/a, p. e12891, doi: 10.1002/eng2.12891.

[25] D. Pogrebnoi, A. Funkner, and S. Kovalchuk, "RuMedSpellchecker: A new approach for advanced spelling error correction in Russian electronic health records," *Journal of Computational Science*, vol. 82, p. 102393, Oct. 2024, doi: 10.1016/j.jocs.2024.102393.

[26] E., O'Neill, R., Young, E., Thiaville, M., MacCarthy, J., Carson-Berndsen, & A. Ventresque, S-capade: Spelling correction aimed at particularly deviant errors. In *Statistical Language and Speech Processing: 8th International Conference, SLSP 2020, Cardiff, UK, October 14–16, 2020, Proceedings 8* (pp. 85-96). Springer International Publishing.

[27] U., Liyanapathirana, K., Gunasinghe, & G. Dias. Sinspell: A comprehensive spelling checker for sinhala. *arXiv preprint arXiv:2107.02983*, 2021.

[28] O. Abiola, A. Abayomi-Alli, O. A. Tale, S. Misra, and O. Abayomi-Alli, "Sentiment analysis of COVID-19 tweets from selected hashtags in Nigeria using VADER and Text Blob analyser," *Journal of Electrical Systems and Inf Technol*, vol. 10, no. 1, p. 5, Jan. 2023, doi: 10.1186/s43067-023-00070-9.

[29] N. Bölücü and B. Can, "Context Based Automatic Spelling Correction for Turkish," *2019 Scientific Meeting on Electrical-Electronics & Biomedical Engineering and Computer Science (EBBT)*, Istanbul, Turkey, 2019, pp. 1-4, doi: 10.1109/EBBT.2019.8742067.

[30] Aydoğan, M., & Karci, A. (2020). Spelling Correction with the Dictionary Method for the Turkish Language Using Word Embeddings. *Avrupa Bilim ve Teknoloji Dergisi*, 57–63. https://doi.org/10.31590/ejosat.araconf8

[31] O. Büyük, M. Erden and L. M. Arslan, "Context Influence on Sequence to Sequence Turkish Spelling Correction," *2019 27th Signal Processing and Communications Applications Conference (SIU)*, Sivas, Turkey, 2019, pp. 1-4, doi: 10.1109/SIU.2019.8806476.

Author(s) Contributions

Pinar Savci: Data generation, performing analysis, writing.

Bihter Das: Supervision, writing, review, and editing.

Conflict of Interest Notice

The authors declare that there is no conflict of interest regarding the publication of this paper.

Support/Supporting Organizations

This work is supported by the Republic of Turkey, Ministry of Science, Technology and Industry project named "AI Based Smart Digital Assistant Customer Dialog Bot project" and project code AR-22-087-0001. It is funded by R&D project within the scope of law 5746 by the Arçelik Digital Transformation, Big Data and Artificial Intelligence R&D Center.

Ethical Approval and Informed Consent

It is declared that during the preparation process of this study, scientific and ethical principles were followed.

Availability of data and material

The following link can be used to access the dataset: https://huggingface.co/datasets/pnr-svc/spellchecker-dataset

Plagiarism Statement

This article has been scanned by iThenticate ™.

SAKARYA UNIVERSITY JOURNAL OF COMPUTER AND INFORMATION SCIENCES
http://saucis.sakarya.edu.tr/
e-ISSN: 2636-8129
Publisher: Sakarya University
Vol. 7, No. 3, 2024, 416-426
DOI: 10.35377/saucis...1493368
SAKARYA UNIVERSITY
RESEARCH ARTICLE

Ensemble-Based Alzheimer's Disease Classification Using Features Extracted from Hog Descriptor and Pre-trained Models

Nedim Muzoğlu[1*], Enver Akbacak[2]

[1] University of Health Sciences, Department of Bioengineering, Istanbul, Türkiye
[2] Fenerbahçe University, Department of Computer Engineering, Istanbul, Türkiye

Corresponding author:
Nedim Muzoğlu, University of Health Sciences, Department of Bioengineering, Istanbul, Türkiye,
nedim.muzoglu@sbu.edu.tr

Article History:
Received: 31.05.2024
Accepted: 30.09.2024
Published Online: 23.12.2024

ABSTRACT
Alzheimer's Disease is the most common type of dementia and is a progressive, neurodegenerative disease. The disease worsens over time, and the patient becomes bedridden, unable to move or understand what is happening around him. The main concern of medicine is to slow down the progression of the disease for which no treatment has yet been developed. Artificial intelligence studies have achieved significant success in detecting many diseases. In this study, an artificial intelligence-based approach that uses MR images of the early stage of Alzheimer's Disease to detect the disease at an early stage is presented. Initially, a new dataset was created through the application of the fuzzy technique, thereby expanding the feature space. Then, an ensemble learning-based hybrid deep learning model was developed to reduce the misclassification rate for all classes. The features derived from the inception module, residual modules, and histogram of oriented gradients descriptor are subjected to classification through bagging and boosting algorithms. The proposed model has surpassed many state-of-the-art studies by achieving a high success rate of 99.60% in detecting Alzheimer's disease in its early stages.

Keywords: Alzheimer disease, Hog features, Imbalance dataset, Bagging and boosting, Fuzzy image enhancement

1. Introduction

Alzheimer's Disease (AD) is the most common form of dementia. According to 2020 data, more than 55 million AD patients worldwide. This number is estimated to reach 139 million in 2050 [1]. Moreover, it is also reported that most of the disease will occur in developing countries, where the elderly population is increasing, and this poses a worldwide public health problem. According to TUIK data, and cause-of-death statistics in our country, the rate of older people dying from AD was 3.2% in 2022 [2].

AD is a neurological disease that affects neurons in the brain [3]. In patients with AD, abnormal proteins called amyloid plaques and neurofibrillary tangles accumulate in the brain, interrupting the communication between neurons and causing neurons to lose their functions and die. As AD progresses, these plaques and tangles spread and seriously affect some areas of the brain. This spread causes shrinkage in the hippocampus, which stores memories from the patient's recent past and reduces the ability to remember. Another area of the brain affected is the parietal lobe [4]. The shrinkage of the parietal lobe negatively affects various abilities, such as maintaining body balance, perception ability, and calculation. Thus, the disease destroys brain functions. It prevents the patient from performing tasks such as swallowing and breathing due to the weakening of the respiratory muscles, thus causing complications in the organs and causing the death of the patient.

The clinical evaluation of the disease encompasses three main phases. Symptoms observed in the mild phase include memory and speech problems, difficulty in routine daily tasks, loss of balance, impairment in abstract thinking, challenges in planning and problem-solving, misplacement of items, decision-making difficulties, and manifestations of depression. In the intermediate phase, notable are the deficits in executive functions, particularly in personal care. In the advanced stage, there is an increasing dependence on the bed and family members, accompanied by an increased risk of infection and embolism [5]. Due to its gradual progression spanning numerous years, AD may necessitate up to two decades for early-stage symptoms to manifest. Early diagnosis of the disease is crucial for initiating pharmacological treatment methods. Although early treatment methods cannot stop the progression of the disease, they can slow it down.

Cite as: N. Muzoğlu, E. Akbacak, "Ensemble-Based Alzheimer's Disease Classification Using Features Extracted from Hog Descriptor and Pre-trained Models," *Sakarya University Journal of Computer and Information Sciences*, vol.7, no. 3, pp. 416-426, 2024. Doi: 10.35377/saucis…1493368

PET, CT/PET, MRI, and PET/MRI imaging technologies are medical imaging techniques used to detect disease. Among these health technologies, magnetic resonance (MR) imaging emerges as an economically viable modality, devoid of radiological effects, and widely accessible across numerous health institutions. This method requires the expertise of radiologists to discern the course of the disease through examination of anatomical changes identified in MRI images. However, examining high-dimensional data obtained from neuroimaging modalities requires experienced radiologists, and it is time-consuming. Therefore, in the last twenty years, significant successes have been achieved in disease detection through artificial intelligence methods, aiming to support this issue. Researchers have also undertaken artificial intelligence studies focusing on the cerebral alterations induced by this disease, leveraging the features extracted from MRI images.

Traditional feature extraction methods and machine learning algorithms are commonly employed in studies aimed at detecting AD are as follows. Many machines learning models have been developed to classify brain tissue types, including white matter, cerebrospinal fluid (CSF), and gray matter (GM), with the aim of detecting AD. Features extracted from MRI images have been commonly employed in these studies. A local feature descriptor was obtained by [6] combining the fast Hessian detector to detect important voxels from different sections of MRI slices and the local binary pattern texture operator to determine variations in different sections, resulting in an 88.99% success rate achieved through training with a CNN network. [7] utilized the bag-of-feature method in conjunction with support vector machines (SVM) to classify MRI images within the AD Neuroimaging Initiative (ADNI) dataset. This approach yielded an accuracy rate of 93%. Another deep learning team is utilized ADNI dataset to detect the stages of AD using machine learning methods. They applied k-nearest neighbours, decision tree, Naive Bayes, and generalized linear model GLM with 88.24% accuracy [8]. In a different study [9], initially input the features obtained from the ADNI dataset into the kernel principal component analysis module and subsequently projected the Principal Component Analysis (PCA) coefficients onto the linear discriminant space. Then, they employed SVM for classification and achieved a disease detection accuracy of 93%. In the study [10] utilized the semi-supervised machine learning method using the ADNI dataset to detect the transition stage from mild cognitive impairment to AD. MRI images underwent a 10-fold cross-validation after the feature selection process, resulting in a 90% AUC score. [11] attained a success rate of 98.17% in the early diagnosis of Alzheimer's disease (AD) by employing a combined feature method. This method integrates the extraction of Histogram of Oriented Gradients (HOG) features and PCA feature selection, followed by classification using decision trees. Unlike many other studies, Liu at all. utilized spectrogram features extracted from speech data instead of image data to detect AD. In the study comparing speech data from AD patients with the Dem@Care dataset, they achieved an accuracy of 83% using the logistic regression method [12].

Although machine learning studies have been carried out to analyze high-dimensional data, they could be time-consuming since feature extraction, feature selection, dimensionality reduction, and classification steps could be computationally intensive. Given that these studies demand researchers to possess subject-specific skills, deep learning methods, which offer numerous advantages surpassing those of machine learning studies, have become increasingly popular in recent years. The LeNet deep learning architecture constitutes a pivotal milestone in the evolution of deep learning models [13]. It is based on providing the features extracted from convolutional networks as input to increase artificial neural networks' training and classification success. Subsequently, numerous deep learning models have emerged, showcasing their efficacy through notable achievements in ImageNet Large Scale Visual Recognition Challenge (ILSVRC) competitions. AlexNet and VGG16, with their sequential layer structure, were among the most prominent models in the following years [13]. However, these models could not overcome some problems, such as vanishing gradient problems. The solution to this problem has been ResNet models [14] developed with the help of residual blocks. While these models have addressed the issue of updating weights in deep layers, the increase in dimensionality with increasing depth remains another problem that needs to be resolved. GoogLeNet [15] , developed by Google Researcher, solved this problem with its Inception modules. Deep learning models continue to develop by providing solutions to issues encountered.

On the other hand, various studies have shown that machine learning applications have been successfully used to classify the features extracted by deep learning methods. Machine learning methods indeed have the potential to be highly successful in classifying datasets with unbalanced data distributions between classes. [16] introduced a deep learning-based pipeline model for AD diagnosis. This model implements an average strategy, concatenating bagging and majority voting methods, and combining features extracted from fine-tuned models, namely AlexNet, ResNet101, and InceptionResNetV2. It achieved a success rate of 98.51%. In the capsule CNN-based model proposed by [17] for the early diagnosis of AD, the preprocessing steps include histogram equalization and Gaussian filtering. Subsequently, segmentation is performed using UNET, while feature extraction is based on the Improved Gray-Level Run Length Matrix and the Gray-Level Size Zone Matrix. In addition, the model integrates Equestrian Ecosystem Optimization for feature selection. This ensemble approach yields to an overall success rate of 94.52%. [18] employed an adaptive synthetic oversampling technique on the highly imbalanced ADNI dataset to efficiently discern AD. Subsequently, the combined prediction values of the EfficientNet-B2 and VGG-16 models were obtained. This approach resulted in a prediction success rate of 94%.

In the literature review, the efficacy of methodologies leveraging Histogram of Oriented Gradients (HOG) features, deep learning models, and ensemble machine learning methods stands out. Based on this observation, the current study proposes the classification of features derived from both HOG and deep learning models employing ensemble machine learning methods. Since the AD data set, we used to be imbalanced between classes, the proposed approach as a solution is as follows. The distinctive features extracted from both inception and residual deep learning models, which capture detailed and coarse

features from images at various scales and exhibit superior performance in handling gradient problems, were merged. These were later combined with HOG features. Subsequently, these combined features were classified using ensemble machine learning methods, among the leading approaches for handling imbalanced datasets between classes. The contributions of this study are:

- The reconstruction of the AD data set utilizing the Fuzzy logic image enhancement method to expand the feature space.
- The problem of features that cannot be learned sufficiently has been tried to be overcome with the help of multi-scale deep learning models and residual structures.
- The Discriminant HOG features extracted from the the dataset were also utilized to capitalize on their potential to enhance classification accuracy.
- Ensemble learning-based bagging and boosting methods achieved high classification success as a solution to the misclassification problem in minority classes in imbalanced datasets.

The remaining sections of this research are organized as follows: Next section will provide an exposition on the dataset, models, and the proposed method, while the subsequent section will detail the experimental results and a comparison of the success achieved in similar studies pertaining to AD classification will be conducted. Lastly, the concluding section will include the discussion and offer conclusive remarks.

2. Materials and Methods

2.1. Dataset

The AD dataset utilized in this study was obtained from Kaggle and is openly accessible [19]. The dataset has 6400 MRI images and consists of four classes of early stages of the disease: non-dementia, very mild dementia, mild dementia, and moderate dementia. The ratio of training to testing was set at 80:20, a common choice observed in numerous successful studies of a similar nature. Brain MRI images are sized at 176x208 pixels and are in .jpg format. Sample images from each class of both the Alzheimer Dataset and the AlzheimerF Dataset, which underwent enhancement using the fuzzy method, are depicted as seen in Figure 1. Additionally, details regarding the number of images in the datasets are as seen in Table 1.

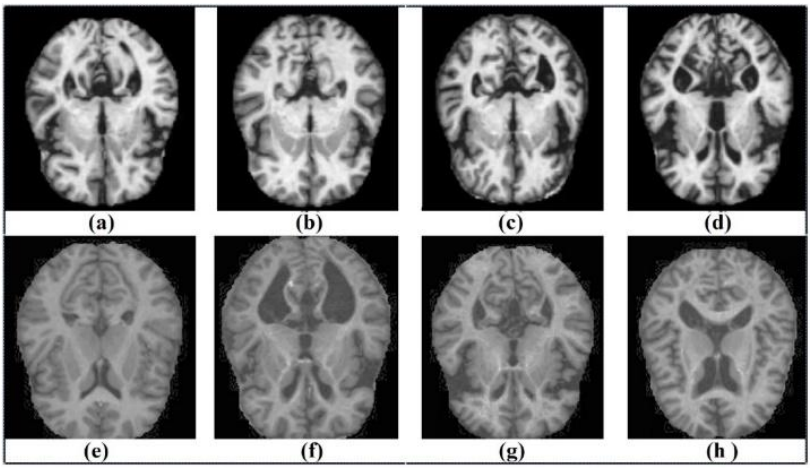


Figure 1. (a) Alzheimer Non-dementia, (b) Alzheimer Very Mild Dementia, (c) Alzheimer Mild Dementia, (d) Alzheimer Moderate Dementia, (e) AlzheimerF Non-dementia, (f) AlzheimerF Very Mild Dementia, (g) AlzheimerF Mild Dementia, (h) AlzheimerF Moderate Dementia

Table 1. AD Early-Stage Dataset Details

Classes	Train	Test	Total
NonDemented	2560	640	3200
VeryMildDemented	1792	448	2240
Mild dementia	717	179	896
ModerateDemented	51	13	64
Total	5120	1280	6400

2.2. Deep Learning Models

GoogLeNet [20] and ResNet18 [21] were employed in the deep architecture of this study. GoogLeNet was initially introduced in 2014 [22] at the ILSRVC, where it competed on behalf of Google Research's. Before GoogLeNet, researchers assumed that the deeper a network is, the more efficient it is. However, as the layers deepen, it causes the gradients to approach zero, and the learning process becomes saturated, known as the vanishing gradient problem [23]. In addition, the computational costs increase. With GoogLeNet, it is ensured that although a CNN network is more complex and deeper, its computational cost is not excessively high. It is based on using parallel paths, i.e., initial modules, each with different filter sizes, instead of the gradual growth of the network depth. In an inception module [24], the outputs of CNN paths with filters of different sizes are concatenated. Thus, features at different scales are captured at the same time. In the inception module in a layer, each of the various size filters, such as 1x1, 3x3, and 5x5, is located on a different parallel path. Paths with 3x3 and 5x5 filters allow both small and global patterns to be captured. 1x1 convolutions performed before those performed by 3x3 and 5x5 filters reduce the number of input channels and provide dimension reduction. In this way, the computational cost is also reduced. As a result, although the network is deeper, both efficiency increases and computational cost is reduced. GoogLeNet consists of multiple stacked inception layers, each with an average pooling or fully connected layer at the end. Additionally, auxiliary classifiers were used in intermediate layers. These classifiers reduce the vanishing gradient problem by providing additional supervision information for backpropagation. Overall, the GoogLeNet architecture achieves multi-scale feature extraction, decreased effectiveness of the vanishing gradient, and computational efficiency. GoogLeNet has 22 layers, and inception modules are stacked. Each inception module contains Convolution, pooling, and concatenation, normalization operations, respectively. There are different versions of inception deep learning. GoogLeNet is also known as InceptionV1.

In deep networks, the gradients calculated for previous layers tend to shrink exponentially as the network gets deeper, and updating the weights becomes difficult, thus losing the network's ability to learn complex relationships in the data. As a solution to this challenging problem, Residual Network was introduced in the 2015 paper "Deep Residual Learning for Image Recognition" by [25]. This groundbreaking architecture, which enables easier flow from previous to subsequent layers by directly skipping several layers, has now solved the gradient problem. In the ResNet structure, the block that passes the input value through layers and connects the input back to the output is called the residual block, and it is shown as seen in Figure 2.

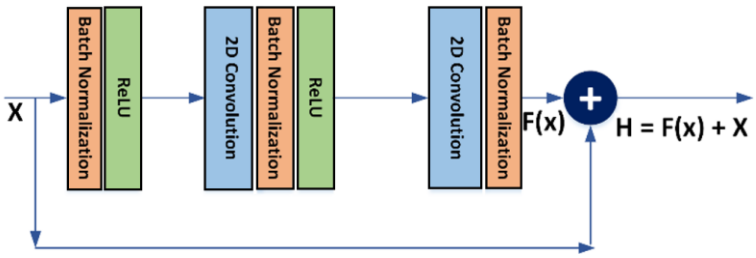


Figure 2. The Residual Blocks Architectures

ResNet18 and GoogLeNet were used in this study to benefit from these superior features. Moreover, given the imbalance within the dataset and the limited number of images in the minority classes, the first 10 layers of the models were frozen, and weight transfer based on transfer learning was implemented. Gradient Descent [26] was preferred for training the models and optimizing the error function.

2.3. Histogram of Oriented Gradients

The Histogram of Oriented Gradients (HOG) method continues to be recognized as a robust feature descriptor, maintaining relevance due to its resilience against illumination variations and transformations induced by medical devices in tissue imaging [27]. HOG can basically be defined as a feature descriptor that discards unnecessary elements while selectively extracting relevant information. The utilization of HOG descriptors on images is linked to the concept of gradients and directions. Gradients represent sudden transitions between pixels within an image, where the shift from darker to lighter shades is denoted as a positive gradient. To accomplish this task, the gradient magnitude and gradient angle of each pixel must first be computed. Subsequently, the histogram of gradients for each cell should be derived within the image partitioned into cells. Thus, the descriptors of the image are obtained by aggregating the HOG features from each cell [28]. In this study, the cell size for the Histogram of Oriented Gradients (HOG) feature extraction algorithm was set to 32 x 32 pixels, while all other parameters were maintained at their default values as provided by MATLAB 2022b. In this study, a total of 780 features were extracted per image from both the Alzheimer and AlzheimerF datasets, employing the HOG descriptor according to the parameters.

2.4. Fuzzy Image Enhancement Method

In medical imaging, image enhancement methods increase hidden image details or contrast, allowing medical professionals to identify abnormalities, tumors, and other critical information more effectively. Fuzzy image enhancement is achieved by mapping the gray levels of the image onto a fuzzy plane [29]. This technique creates an image with higher contrast than the

original image by giving more weight to the image's average gray levels than those further away from the average value. Fuzzy image processing consists of fuzzification, where the image is encoded, fuzzy layer operations, and defuzzification, where the results are decoded [30]. The fuzzy image enhancement was performed by utilizing the Python Open CV library. The epsilon, gamma and variance values used in the fuzzy image generation are 0.00001, 1, and 0.35, respectively.

We aim to obtain rich features by using the Alzheimer Dataset, and AlzheimerF Dataset which is improved with the fuzzy logic image enhancement method. This approach based on the deep integration of GoogLeNet, which can extract multi-scale features, and the residual neural network, which successfully vanishes gradient problems, and also HOG technique.

2.5. Ensemble Methods

The AD data sets were classified with Bagging and Boosting structures, which are ensemble algorithms [31]. In the bagging method, random samples are selected from the original dataset for each of the different classifiers. However, the total number of samples in the subsample groups must be equal to the total number of samples in the original data set. But the selected samples continue to remain in the original dataset. This means that the same samples can be present in different classifiers [32]. The average of the prediction of each trained classifier is taken, so the goal is to reduce the variance with the bagging method, and thus overfitting. In this study, decision trees were used as learners in the bagging method, and the number of learners and the maximum split were set to 20 and 5000, respectively. The Boosting method is based on a series of models that works sequentially and focus on examples that are difficult to classify [31]. Subsequently, the training process of the method proceeds by assigning greater weight to predictions that were misclassified by the preceding classifier.

2.6. Artificial Intelligence Approaches to Tackle the Imbalance Data Distribution

The ratio of samples in the four classes of the data set is 1:13:35:50. Therefore, the dataset is imbalanced between classes. In imbalanced data sets, the majority classes dominate in determining the weights during training. This situation results in misclassification during the prediction process. There are three different machine learning approaches to dealing with the classification of imbalanced data sets: data level, cost-sensitive, and ensemble methods [33]. In data-level methods, imbalance problems are dealt with by modifying the data distribution. Under-sampling, a method of removing data from the majority classes, and oversampling, a technique of adding synthetic data to minor classes, are data-level imbalance methods. Since new samples are created by copying existing data in the oversampling method, these samples cause memorization of specific patterns and noise in the training data, which results in overfitting rather than accurate prediction of the test data. On the other hand, since under sampling methods discard potentially valuable data from the majority class, the relevant class cannot be represented correctly, causing errors in the analysis [34]. Another approach, cost-sensitive learning aims to minimize misclassification of the minority classes by assigning cost values to different classes, usually appropriate to the class's number of samples [35]. On the other hand, ensemble machine learning approaches, which combine multiple classifiers to reduce errors and increase their accuracy, achieve better results in classifying imbalance datasets than the other two methods. However, these approaches may not guarantee high classification scores unless the features are considerably representative. Obtaining the features that best represent the dataset is an essential issue in classification problems, and convolutional deep learning models are one of the most effective methods for extracting representative features. However, feature extraction techniques from imbalanced datasets may result in biased features favoring the majority class and poor classification performance for the minority class [36]. From this point of view, it is inevitable to use the rich feature extraction power of deep learning models and the powerful structure of ensemble classifiers in the classification of imbalanced data sets.

Various valuable studies have been conducted on hybrid structures in which deep learning models are used together with ensemble machine-learning methods. Yuan et al. reported that they achieved a 24.7% increase in maximum accuracy by applying regularization methods to the unbalanced endoscope video dataset in the detection of bowel cancer [37]. In the study, the dataset they employed is balanced with synthetic data. [34] integrated the synthetic minority oversampling technique (SMOTE) method with the cross-validated committee filter (CVCF) technique to overcome the problems that cause fatal errors in imbalanced datasets [34]. In their work, they used the ensemble support vector machine introduced as a classifier, together with the majority voting strategy. [38] used pre-trained convolutional neural networks to detect skin cancer. They proposed an ensemble model to solve the imbalance class problem due to the scarcity of representative images from malignant tumors. Considering the proposed technique's area under the ROC curve (AUC-ROC) performance, the ensemble SVM structure has been reported to provide a 16% increase over the pure CNN structure. So, pure CNN feature extraction techniques used with imbalanced datasets may result in biased features favoring the majority class and poor classification performance for the minority class. To address this problem, this study proposes an approach based on the deep integration of GoogLeNet, which can extract multi-scale features for handling the imbalanced AD dataset and the residual neural network, which is successful in mitigating vanishing gradient problems. Moreover, ensemble models were employed to mitigate the impact of biased features favoring the majority class and attain high classification accuracy.

2.7. Proposed Approach

The proposed method entails utilizing 6400 images from an open-access database containing four classes for the early-stage diagnosis of AD. To enrich the feature space, the AlzheimerF dataset was generated by applying the fuzzy enhancement technique to this dataset. Features from both datasets were extracted from the final layers of the GoogLeNet and ResNet18

models, resulting in 1000 features per image in each dataset. Additionally, features from both datasets were extracted using the Histogram of Oriented Gradients (HOG) technique, renowned for its efficacy in feature extraction studies, thereby yielding 5560 features obtained through concatenation of the features extracted from the pre-trained models. To mitigate the misclassification error of minority classes in imbalanced datasets, Bagging and RUS-Boosting tree algorithms were employed, owing to their specific operational algorithms. During the training of these algorithms, the 5-fold cross-validation method, known for yielding optimal results, was favored to alleviate the potential high error stemming from the train and test distribution of the unbalanced dataset. They were then classified to predict the early stages of AD with the highest accuracy. The block diagram of the proposed model is given as seen in Figure 3.

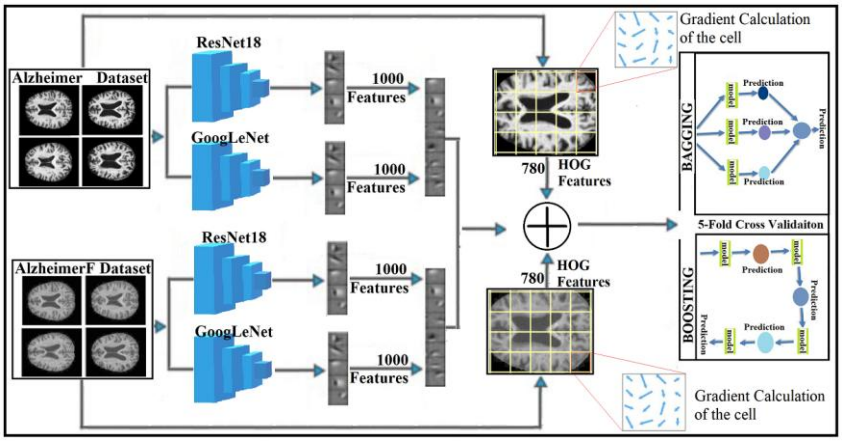


Figure 3. The Functional Block Diagram of The Proposed Method

3. Experimental Results

3.1. Experimental Setup

The confusion matrix values were utilized to evaluate the model. The accuracy metric (Acc) is obtained by dividing the number of correctly predicted samples by the total number of predicted samples. To obtain recall and F-Score (F-Scr), which are other essential evaluation metrics, true positive (TP), false positive (FP), true negative (TN), and false negative (FN) values must be obtained from the confusion matrix [39]. Evaluation metrics are obtained as shown in Eq. 1-3.

$$\text{Recall} = \text{TP}/(\text{TP} + \text{FN}) \tag{1}$$

$$\text{F} - \text{Scr} = \frac{2xTP}{2xTP + FP + FN} \tag{2}$$

$$\text{Acc} = \frac{TP + TN}{TP + TN + FP + FN} \tag{3}$$

This study performs experimental results in Keras API, Python, and MATLAB R2022b. The computer used in the study has a Windows 10 (64-bit) operating system and an NVIDIA GeForce RTX3070 graphics card. The mini-batch size, which is related to the suitability of the equipment, was preferred as 64. The learning rate and the number of epochs were set to 0.0001 and 100, respectively.

3.2. Experimental Results

In the first step of this study, deep learning models were trained with the AD Dataset, and AlzheimerF datasets obtained by the fuzzy logic image enhancement method. The training progress is presented as seen in Figure 4. This figure shows the overfitting problem commonly encountered during the training process of the imbalanced datasets. In this step, 85.26% and 86.72% accuracy are achieved in the GoogLeNet and ResNet18 models using the AD Dataset, respectively. For the AlzheimerF dataset, these accuracies are 86.02% and 87.26%, respectively.

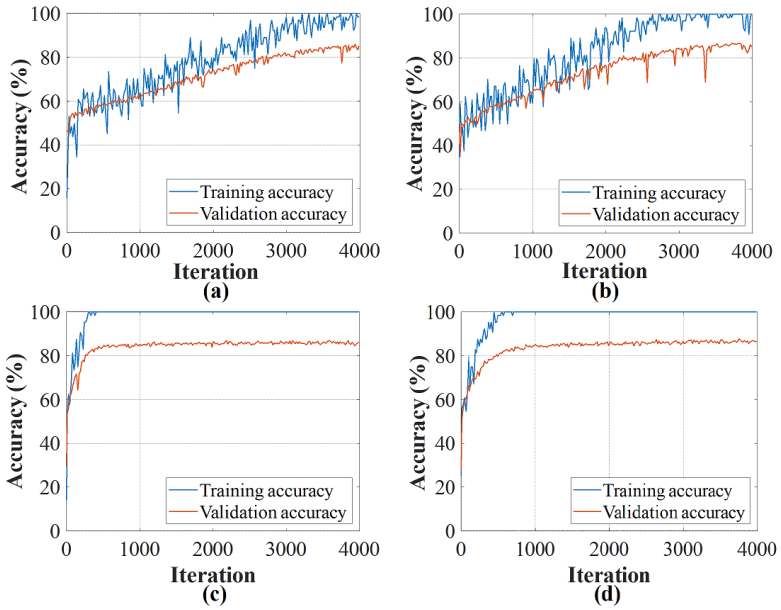


Figure 4. (a) GoogLeNet Alzheimer Dataset, (b) GoogLeNet AlzheimerF Dataset, (c) ResNet18 Alzheimer Dataset, (d) ResNet18 AlzheimerF Dataset

Figure 5 shows the confusion matrix results of these classification processes. It is seen that the classification success of the Softmax classifier with 1000 features in each model is not sufficiently high, particularly when evaluating human health.

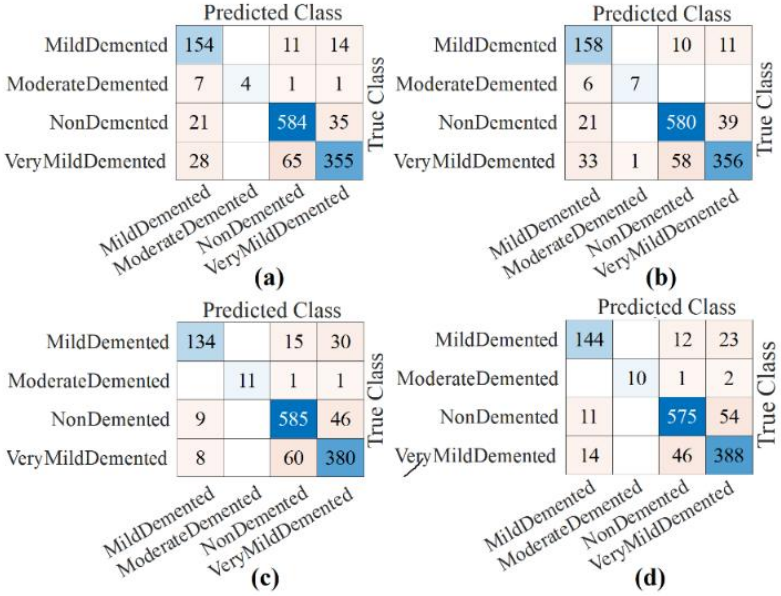


Figure 5. (a) GoogLeNet Alzheimer Dataset, (b) GoogLeNet AlzheimerF Dataset, (c) ResNet18 Alzheimer Dataset, (d) ResNet18 AlzheimerF Dataset

Given that the method's primary aim is to minimize misclassification in minority classes by reducing overfitting, the features that best represent the datasets were initially extracted from the thoroughly trained GoogLeNet, ResNet18 models. These features were then fed into the Bagging and RUS-Boosting Tree modules. Bagged Trees and Boosted Trees classifiers, which are ensemble learning algorithms, are preferred in classifying the imbalance dataset to classify minority classes with low misclassification errors. In the first step of the proposed approach achieved classification success rates of 97.34% and 97.57% for the Bagging and RUS-Boosting Tree models, respectively, in the detection of AD solely through the utilization of features acquired from deep learning modules. Figure 6 illustrates the confusion matrices acquired subsequent to the classification process of Bagging and RUS-Boosting Tree models utilizing 4000 features extracted from Alzheimer and AlzheimerF datasets through deep learning models.

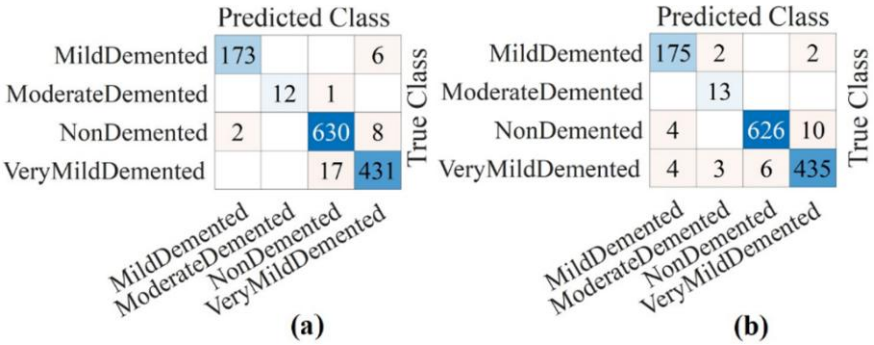


Figure 6. (a) 4000 Feature with Bagged Trees, (b) 4000 Feature with Rus-boosted Trees

In the Figure 7, the outcomes of the proposed approach are showcased, achieved through ensemble classification by enriching the features extracted from deep learning models with an additional set of 1440 HOG features.

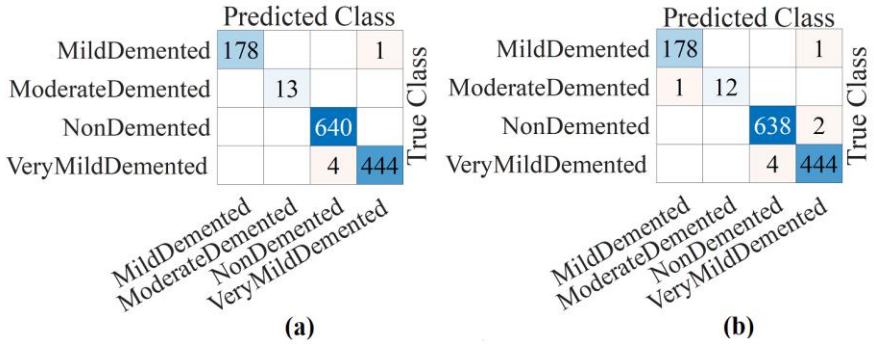


Figure 7. (a) 5440 Feature with Bagged Trees, (b) 5440 Feature with Rus-boosted Trees

The confusion matrices in Figure 7 clearly show that enriching the feature space through the inclusion of HOG features has facilitated ensemble classifiers in attaining notable success rates in reducing misclassifications. When the results scrutinized concerning misclassification errors within minority classes stemming from the imbalanced dataset, one of the primary objectives of this study, the Bagging method rendered predictions with only a single error in total for the mild and moderately demented classes. Moreover, in the very mild dementia class, closely associated with early-stage disease detection, both the Bagging and Boosting methods achieved remarkably high classification success, with only four misclassifications predicted in total. While the utilization of ResNet-18, a prominent deep learning model, attained an 87% success rate on the AlzheimerF dataset, the proposed model achieved a substantial 14.14% increase in accuracy, showcasing its efficacy and superiority in performance.

In conclusion, the application of the proposed approach resulted in notable accuracies for both bagging and boosting methods. Specifically, bagging methods yielded an accuracy of 99.60%, while boosting methods achieved an accuracy of 99.40%. The classification results are evaluated as seen in the Table 2.

Table 2. Performance Metrics Obtained with Ensemble Learning Classifier for the AD

Models	Classes	Recall (%)	Pre (%)	F-Scr (%)	Acc (%)
Bagged Trees	Mild Demented	100	99.44	99.72	99.60
	Moderate Demented	100	100	100	
	Non Demented	99.38	100	99.69	
	VeryMild Demented	99.76	99.11	99.44	
RUS-Boosted Trees	Mild Demented	99.44	99.44	99.44	99.37
	Moderate Demented	100	92.31	96	
	Non Demented	99.38	99.69	99.53	
	VeryMild Demented	99.33	99.11	99.22	

Numerous studies have been undertaken concerning the early-stage detection of AD employing a variety of methodologies, including traditional techniques, deep learning architectures, and hybrid models. Table 3 compares similar approaches using the AD dataset.

Table 3. The Comparison of the Proposed Approach with Similar Studies

Authors	Methods	Recall (%)	F-Scr (%)	Acc (%)
[19], 2018	Ensemble Learning, Convolutional Neural Networks.	93	92	93.18
[40], 2022	Transfer Learning, Alex Net.	94.21	94.12	94.53
[41], 2021	Oversampling, Occlusion Sensitivity Maps.	95.20	95.30	95.20
[42], 2022	Transfer Learning, Oversampling.	97	97	97.05
[43], 2022	Convolutional Neural Networks, Slice Based methods.	98	97	98.37
[44], 2022	Transfer Learning, Gradient Activation Maps	98.14	98.14	98.21
[45], 2023	Transfer Learning, PCA	99.87	99.89	99.88
Proposed approach	Ensemble Learning, Transfer Learning, Hog descriptor	99.79	99.64	99.60

4. Conclusion

This study proposes the detection of AD at an early stage through the utilization of an ensemble learning approach. The primary objective of this study is to mitigate the challenges posed by imbalanced data distributions through the expansion of the feature space. In this methodology, features extracted from the Alzheimer, and AlzheimerF datasets via traditional and transfer learning-based techniques are subsequently subjected to classification. The transfer learning models employed were chosen from two fundamentally different structures, the inception block, and the residual structure, to extract various features from the dataset. Moreover, the traditional hog feature extractor, which is still up to date, was preferred due to its outstanding success. Classifiers such as bagging and boosting, which obtain results with the predictive power of many classifiers, were used to minimize the misclassification error encountered in the imbalanced datasets. A closer look at the dataset reveals that the classes 'mild dementia' and 'moderate dementia' are significantly underrepresented and have only a small number of samples compared to the other classes. Especially the high imbalance moderately demented and mild demented classes were successful with the bagging method, except for one error.

Early diagnosis of Alzheimer's disease is crucial because early intervention can slow down the progression of the disease. The disease's progression can be assessed from MRI images based on radiologists' reports. However, differences in decision-making among radiologists may occur due to variations in cognitive load. Hence, this study and similar approaches are recommended as computer-supported artificial intelligence assistants. In future studies with this dataset, the aim is to detect the disease in minority classes with the lowest misclassification using a cost-sensitive learning-based deep learning approach, one of the ensemble methods.

References

[1] L. C. Arevalo-Flechas, “Latino Alzheimer ’ s Caregiving Neither a Burden nor a Carga,” *Univ. ealth Sci. J. Nurs.*, vol. 1, no. 2, pp. 92–103, 2019.

[2] M. ŞENER and H. H. TEKİN, “Sosyal Belediyecilik Bağlamında Yaşlı Bakım ve Alzheimer Gündüz Yaşam Merkezleri,” *Geriatr. Bilim. Derg.*, vol. 3, no. 3, pp. 138–146, 2020, doi: 10.47141/geriatrik.737313.

[3] D. S. Knopman *et al.*, “Alzheimer disease,” *Nat. Rev. Dis. Prim.*, vol. 7, no. 1, pp. 1–21, 2021, doi: 10.1038/s41572-021-00269-y.

[4] E. Keleş, U. Fzt, S. Özalevli, İ. Kâtip Çelebi Üniversitesi Sağlık Bilimleri Fakültesi, and F. ve Rehabilitasyon Bölümü, “Alzheimer Hastalığı ve Tedavi Yaklaşımları Alzheimer’s Disease and Treatment Approaches,” *İzmir*

Katip Çelebi Üniversitesi Sağlık Bilim. Fakültesi Derg., vol. 3, no. 2, pp. 39–42, 2018, [Online]. Available: https://www.alz.org/alzheimers_disease_stages_of_alzheimers.asp.

[5] P. S. Aisen, G. A. Jimenez-Maggiora, M. S. Rafii, S. Walter, and R. Raman, "Early-stage Alzheimer disease: getting trial-ready," *Nat. Rev. Neurol.*, vol. 18, no. 7, pp. 389–399, 2022, doi: 10.1038/s41582-022-00645-6.

[6] A. Francis and I. A. Pandian, "Early detection of Alzheimer's disease using local binary pattern and convolutional neural network," *Multimed. Tools Appl.*, vol. 80, no. 19, pp. 29585–29600, 2021, doi: 10.1007/s11042-021-11161-y.

[7] D. Bansal, K. Khanna, R. Chhikara, R. K. Dua, and R. Malhotra, "Classification of Magnetic Resonance Images using Bag of Features for Detecting Dementia," *Procedia Comput. Sci.*, vol. 167, no. 2019, pp. 131–137, 2020, doi: 10.1016/j.procs.2020.03.190.

[8] M. Shahbaz, S. Ali, A. Guergachi, A. Niazi, and A. Umer, "Classification of Alzheimer's disease using machine learning techniques," *DATA 2019 - Proc. 8th Int. Conf. Data Sci. Technol. Appl.*, no. Data, pp. 296–303, 2019, doi: 10.*5220*/0007949902960303.

[9] S. Alam and G. R. Kwon, "Alzheimer disease classification using KPCA, LDA, and multi-kernel learning SVM," *Int. J. Imaging Syst. Technol.*, vol. 27, no. 2, pp. 133–143, 2017, doi: 10.1002/ima.22217.

[10] E. Moradi, A. Pepe, C. Gaser, H. Huttunen, and J. Tohka, "Machine learning framework for early MRI-based Alzheimer's conversion prediction in MCI subjects," *Neuroimage*, vol. 104, pp. 398–412, 2015, doi: 10.1016/j.neuroimage.2014.10.002.

[11] J. A. Dinu and R. Manju, "A novel modelling technique for early recognition and classification of Alzheimer's disease," *2021 3rd Int. Conf. Signal Process. Commun. ICPSC 2021*, no. May, pp. 21–25, 2021, doi: 10.1109/ICSPC51351.2021.9451803.

[12] L. Liu, S. Zhao, H. Chen, and A. Wang, "A new machine learning method for identifying Alzheimer's disease," *Simul. Model. Pract. Theory*, vol. 99, no. November 2019, p. 102023, 2020, doi: 10.1016/j.simpat.2019.102023.

[13] X. Zhang, "The AlexNet, LeNet-5 and VGG NET applied to CIFAR-10," *Proc. - 2021 2nd Int. Conf. Big Data Artif. Intell. Softw. Eng. ICBASE 2021*, pp. 414–419, 2021, doi: 10.1109/ICBASE53849.2021.00083.

[14] C. Szegedy *et al.*, "Going deeper with convolutions," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 07-12-June, pp. 1–9, 2015, doi: 10.1109/CVPR.2015.7298594.

[15] C. Szegedy *et al.*, "Going Deeper with Convolutions," Sep. 2014, [Online]. Available: http://arxiv.org/abs/1409.4842.

[16] A. Loddo, S. Buttau, and C. Di Ruberto, "Deep learning based pipelines for Alzheimer's disease diagnosis: A comparative study and a novel deep-ensemble method," *Comput. Biol. Med.*, vol. 141, no. August 2021, p. 105032, 2022, doi: 10.1016/j.compbiomed.2021.105032.

[17] R. Butta, M. S. Shaik, and G. L. N. Murthy, "Ensemble deep learning approach for early diagnosis of Alzheimer's disease," *Multimed. Tools Appl.*, no. 0123456789, 2024, doi: 10.1007/s11042-023-18084-w.

[18] M. Mujahid, A. Rehman, T. Alam, F. S. Alamri, S. M. Fati, and T. Saba, "An Efficient Ensemble Approach for Alzheimer's Disease Detection Using an Adaptive Synthetic Technique and Deep Learning," *Diagnostics*, vol. 13, no. 15, pp. 1–20, 2023, doi: 10.3390/diagnostics13152489.

[19] J. Islam and Y. Zhang, "Brain MRI analysis for Alzheimer's disease diagnosis using an ensemble system of deep convolutional neural networks," *Brain Informatics*, vol. 5, no. 2, 2018, doi: 10.1186/s40708-018-0080-3.

[20] L. Balagourouchetty, J. K. Pragatheeswaran, B. Pottakkat, and G. Ramkumar, "GoogLeNet-Based Ensemble FCNet Classifier for Focal Liver Lesion Diagnosis," *IEEE J. Biomed. Heal. Informatics*, vol. 24, no. 6, pp. 1686–1694, 2020, doi: 10.1109/JBHI.2019.2942774.

[21] M. Guo and Y. Du, "Classification of Thyroid Ultrasound Standard Plane Images using ResNet-18 Networks," *Proc. Int. Conf. Anti-Counterfeiting, Secur. Identification, ASID*, vol. 2019-Octob, no. i, pp. 324–328, 2019, doi: 10.1109/ICASID.2019.8925267.

[22] O. Russakovsky *et al.*, "ImageNet Large Scale Visual Recognition Challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015, doi: 10.1007/S11263-015-0816-Y.

[23] S. H. Noh, "Analysis of gradient vanishing of RNNs and performance comparison," *Inf.*, vol. 12, no. 11, 2021, doi: 10.3390/info12110442.

[24] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-ResNet and the impact of residual connections on learning," *31st AAAI Conf. Artif. Intell. AAAI 2017*, pp. 4278–4284, 2017, doi: 10.1609/aaai.v31i1.11231.

[25] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2016-Decem, pp. 770–778, 2016, doi: 10.1109/CVPR.2016.90.

[26] S. Ruder, "An overview of gradient descent optimization algorithms," pp. 1–14, 2016, [Online]. Available: http://arxiv.org/abs/1609.04747.

[27] C. I. Patel, D. Labana, S. Pandya, K. Modi, H. Ghayvat, and M. Awais, "Histogram of oriented gradient-based fusion of features for human action recognition in action video sequences," *Sensors (Switzerland)*, vol. 20, no. 24, pp. 1–33, 2020, doi: 10.3390/s20247299.

[28] T. Q. Bao, N. T. T. Kiet, T. Q. Dinh, and H. X. Hiep, "Plant species identification from leaf patterns using histogram of oriented gradients feature space and convolution neural networks," *J. Inf. Telecommun.*, vol. 4, no. 2, pp. 140–

150, 2020, doi: 10.1080/24751839.2019.1666625.
[29] M. Hanmandlu and D. Jha, "An optimal fuzzy system for color image enhancement," *IEEE Trans. Image Process.*, vol. 15, no. 10, pp. 2956–2966, 2006, doi: 10.1109/TIP.2006.877499.
[30] S. Banerjee, S. K. Singh, A. Chakraborty, A. Das, and R. Bag, "Melanoma diagnosis using deep learning and fuzzy logic," *Diagnostics*, vol. 10, no. 8, 2020, doi: 10.3390/diagnostics10080577.
[31] M. Singh, S. Bansal, S. Ahuja, R. K. Dubey, B. K. Panigrahi, and N. Dey, "Transfer learning–based ensemble support vector machine model for automated COVID-19 detection using lung computerized tomography scan data," *Med. Biol. Eng. Comput.*, vol. 59, no. 4, pp. 825–839, Apr. 2021, doi: 10.1007/s11517-020-02299-2.
[32] L. Gang, Z. Haixuan, E. Linning, Z. Ling, L. Yu, and Z. Juming, "Recognition of honeycomb lung in CT images based on improved MobileNet model," *Med. Phys.*, vol. 48, no. 8, pp. 4304–4315, 2021, doi: 10.1002/mp.14873.
[33] Z. Chen, J. Duan, L. Kang, and G. Qiu, "Class-Imbalanced Deep Learning via a Class-Balanced Ensemble," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 33, no. 10, pp. 5626–5640, 2022, doi: 10.1109/TNNLS.2021.3071122.
[34] N. Liu, X. Li, E. Qi, M. Xu, L. Li, and B. Gao, "A novel ensemble learning paradigm for medical diagnosis with imbalanced data," *IEEE Access*, vol. 8, pp. 171263–171280, 2020, doi: 10.1109/ACCESS.2020.3014362.
[35] V. López, A. Fernández, J. G. Moreno-Torres, and F. Herrera, "Analysis of preprocessing vs. cost-sensitive learning for imbalanced classification. Open problems on intrinsic data characteristics," *Expert Syst. Appl.*, vol. 39, no. 7, pp. 6585–6608, Jun. 2012, doi: 10.1016/j.eswa.2011.12.043.
[36] F. Jia, S. Li, H. Zuo, and J. Shen, "Deep Neural Network Ensemble for the Intelligent Fault Diagnosis of Machines under Imbalanced Data," *IEEE Access*, vol. 8, pp. 120974–120982, 2020, doi: 10.1109/ACCESS.2020.3006895.
[37] X. Yuan, L. Xie, and M. Abouelenien, "A regularized ensemble framework of deep learning for cancer detection from multi-class, imbalanced training data," *Pattern Recognit.*, vol. 77, pp. 160–172, 2018, doi: 10.1016/j.patcog.2017.12.017.
[38] A. S. Qureshi and T. Roos, "Transfer Learning with Ensembles of Deep Neural Networks for Skin Cancer Detection in Imbalanced Data Sets," *Neural Process. Lett.*, vol. 55, no. 4, pp. 4461–4479, 2023, doi: 10.1007/s11063-022-11049-4.
[39] M. N. Hossin, M., Sulaiman, "A Review of Evaluation Metrics in Machine Learning Algorithms," *Med. Image Anal.*, vol. 80, no. 2, p. 102478, 2022, [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S1361841522001256.
[40] M. H. Al-Adhaileh, "Diagnosis and classification of Alzheimer's disease by using a convolution neural network algorithm," *Soft Comput.*, vol. 26, no. 16, pp. 7751–7762, 2022, doi: 10.1007/s00500-022-06762-0.
[41] S. Murugan *et al.*, "DEMNET: A Deep Learning Model for Early Diagnosis of Alzheimer Diseases and Dementia from MR Images," *IEEE Access*, vol. 9, pp. 90319–90329, 2021, doi: 10.1109/ACCESS.2021.3090474.
[42] M. M. S. Fareed *et al.*, "ADD-Net: An Effective Deep Learning Model for Early Detection of Alzheimer Disease in MRI Scans," *IEEE Access*, vol. 10, no. September, pp. 96930–96951, 2022, doi: 10.1109/ACCESS.2022.3204395.
[43] N. Goenka and S. Tiwari, "AlzVNet: A volumetric convolutional neural network for multiclass classification of Alzheimer's disease through multiple neuroimaging computational approaches," *Biomed. Signal Process. Control*, vol. 74, no. January, p. 103500, 2022, doi: 10.1016/j.bspc.2022.103500.
[44] M. Odusami, R. Maskeliūnas, and R. Damaševičius, "An Intelligent System for Early Recognition of Alzheimer's Disease Using Neuroimaging," *Sensors*, vol. 22, no. 3, 2022, doi: 10.3390/s22030740.
[45] S. A. Kumar and S. Sasikala, "Enhanced Alzheimer's Disease Classification Using Multilayer Deep Convolutional Neural Network-Based Experimentations," *Iran. J. Sci. Technol. - Trans. Electr. Eng.*, vol. 47, no. 4, pp. 1595–1621, 2023, doi: 10.1007/s40998-023-00622-9.

Conflict of Interest
The authors have no conflict of interest to declare.

Author(s) Contributions
Nedim Muzoğlu: Conceptualization, Methodology, Writing – Original Draft, Software
Enver Akbacak: Methodology, review & editing

Acknowledgments
Not Applicable

Ethical Approval and Informed Consent
This study adheres to scientific and ethical principles, with all referenced works appropriately cited in the bibliography.

Availability of Data and Materials
This dataset is openly accessible via the provided link: https://www.kaggle.com/code/aashidutt3/alzheimer-classification-with-mri-images/input?select=OriginalDataset.

Plagiarism Statement
This article has been scanned by iThenticate ™.

SAKARYA UNIVERSITY JOURNAL OF COMPUTER AND INFORMATION SCIENCES
http://saucis.sakarya.edu.tr/
e-ISSN: 2636-8129
Publisher: Sakarya University
Vol. 7, No. 3, 2024, 427-438
DOI: 10.35377/saucis...1503989
RESEARCH ARTICLE

A Comparison of Transfer Learning Models for Face Recognition

Dalhm Al-Shammari[1*], Devrim Akgun[2]

[1] Computer Engineering Department, Institute of Natural Sciences, Sakarya University, Sakarya, Türkiye
[2] Software Engineering Department, Computer and Information Sciences Faculty, Sakarya University, Sakarya, Türkiye

Corresponding author:
Dalhm Al-Shammari,
Computer Engineering Department,
Institute of Natural Sciences,
Sakarya University, Sakarya, Türkiye
dalhm.ashammari@ogr.sakarya.edu.tr

Article History:
Received: 24.06.2024
Accepted: 04.09.2024
Published Online: 23.12.2024

ABSTRACT
Face recognition (FR) is a method that uses face feature analysis and comparison to identify or verify individuals. Siamese neural networks (SNNs) are an effective method for FR, providing high accuracy and versatility, especially in situations where data is restricted. Unlike standard neural networks, SNNs learn to distinguish between pairs of inputs rather than individual inputs. However, detecting and recognizing faces in unconstrained environments poses a significant challenge due to various factors such as head pose, illumination, and facial expression variations. The aim of this paper is to design and develop an efficient approach based on SNNs and Transfer Learning methods. For this purpose LFW dataset and transfer learning architectures like VGG-16, EfficientNet, RestNet50 and ConvNext have been utilised. Performance of the architectures were measured using 5-Fold cross validation. According to results, EfficientNet, RestNet50 and ConvNext produced 78% accuracy, 95% and 93 % accuracy respectively. SNN with VGG-16 exhibited a low loss and produced the best accuracy in face recognition with 96%.

Keywords: Face recognition, Siamese neural network, VGG-16, ConvNext, EfficientNet, RestNet50

1. Introduction

FRsystems can be tackled either as an identification or verification problem. Face identification, also known as the 1: n matching problem, refers to the process of matching a given face with multiple faces in a database. The unidentified face is compared to all the faces in the known identities database, and a decision is reached based on the outcome of these comparisons. If the person is known to be in the database, the task is referred to as closed-set. Otherwise, it is referred to as open-set. Face verification is commonly referred to as the 1:1 matching problem. The query face's identity is determined by comparing it to the face data of the identity that is claimed in the database. it is either confirmed or rejected based on this comparison [1].

Various FRtechniques have achieved significant success in controlled environments [2], [3]. Due to the nonstable nature of facial images and the fact that it influences how the same person appears in multiple face images in real-life situations, these techniques commonly fail. Therefore, it is necessary to modify these techniques or design new suitable techniques. Due to illumination, pose variation, plastic surgery, expression, ageing, low resolution difficulties, as well as the fact that face acquisition processes may go through a broad range of modifications, the majority of FRalgorithms have not yet achieved the ideal level of recognition accuracy [4]. Therefore, it is important for any proposed solution to focus on enhancing the efficacy of the detection and recognition models while reducing the complexity of resource utilization during the recognition process. Illumination refers to the variations in lighting that can cause a face image to appear differently. The extraction of illumination invariant features continues to be a challenge for robust FR systems. the use of illumination has significant effects on an image, resulting in changes to its location, shadow shape, and contrast gradients [5]. Humans have the ability to recognize faces even when there are changes in lighting. However, FR systems struggle with this task and testing or training these systems is also challenging when lighting conditions vary. Therefore, it is necessary to use image processing techniques to normalize face images that have different lighting conditions.

The general face structure of people may vary significantly as they age, and their distinctive facial features may also alter. As a result, FRtechniques commonly fail in these types of situations [6]. In literature, it has been observed that there is a significant decrease in performance in face recognition-based ageing when there is a large age difference between the target image and the query image [7]. Therefore, the query imagine must be represented using sparse representation classification in order to acquire the residuals for each class, where each pixel is estimated as either occluded or not in the residual for each class.

Cite as: D. Al-Shammari, D. Akgun, "A Comparison of Transfer Learning Models for Face Recognition," *Sakarya University Journal of Computer and Information Sciences*, vol.7, no. 3, pp. 427-438, 2024. Doi: 10.35377/saucis…1503989

Humans are capable of displaying a variety of expressions on their faces at all times, unless they are frozen in a static position. These expressions are utilized for communicating diverse feelings and mental states to others. Facial emotion alters the geometry and appearance of the face, which decreases the accuracy of facial recognition [8]. Therefore, it is necessary to apply customized pre-processing techniques that can only extract features specific for certain expressions-from the face image.

It has been observed that FRtechniques are unable to identify individuals' faces accurately after they undergo plastic surgery [2]. In this scenario, the facial image of a person is completely altered, resulting in a completely different individual. the process of cosmetic surgery on the human face results in a change in skin texture between photographs of the same person, making the job of face identification challenging [9]. Therefore, it is necessary to apply the local binary pattern to important regions of the face image, rather than the entire face image. The concept is grounded in the notion that only that local binary patterns (lbp) containing essential information, such as corners and edges, will be valuable for facial recognition following plastic surgery.

The goal of this research is to develop and implement an efficient methodology based on SNN and transfer learning techniques. This is done using the LFW dataset with transfer learning architectures such as VGG-16, EfficientNet, RestNet50, and ConvNext. The performance of the architectures was evaluated using 5-fold cross validation on LFW dataset. The rest of this paper is organized as follows. Section 2 presents the proposed methodology that utilized to achieve the goals as well this section illustrates the used datasets. The results discussion and analysis of the proposed system is explained is section 3. Section 4 concludes this study.

2. Materials and Methods

This section consists of three actions: designing the SNN with pre-trained networks VGG-16, EfficientNetB0, ConvNeXtBase and ResNet50 as a backbone based FRmodel and providing an explanation of the LFW dataset. Finally, the SNN with VGG-16, EfficientNetB0, ResNet50 and ConvNextTiny model was trained using the same dataset.

2.1. Transfer Learning

A Transfer Learning [10] a particular techniques used to transfer previously learned knowledge gained from a related domain to a target field. Since our data set is compact with around 13,000 samples, to avoid training issues, we perform data transfer. There are specific steps to transfer the particular knowledge. First, we load the model we want to transfer in this study use VGG-16, EfficientNetB0, ConvNeXtBase and ResNet50, and then determine how many layers the previous training requires to keep. Next, we use the new model to be the backbone of the SNN model. The training set used in this second stage is the LFW in the next stage, the pre-trained network model. It is used for LFW testing. Finally, accuracy is calculated based on the task Transfer learning. Architectural flow plan for transfer of learning.

While selected to train the final two layers of the model without freezing it. This approach was selected to optimize the model's adaptability and responsiveness to the dataset, leading to consistent performance and learning In this research, the pre-trained models of VGG16, EfficientNetB0, ConvNeXtBase and Wide ResNet-50 were used. The VGG16, EfficientNetB0, ConvNeXtBase and Wide ResNet-50 pre-trained models expect input images normalized in mini-batches of 3-channel RGB images of shape (3 × H × W), where H and W are expected to be 62 x 47. The final classification layers were replaced with fully connected layers. ReLU activation functions and with a binary cross-entropy loss were also used. The initial layers from training were frozen, and the modified layer was fine-tuned with the LFW dataset. The model was trained for 120 epochs with a batch size of 32. Adam [11]. was used as the model optimizer with training rate (1e-4). To enhance the model’s efficiency in terms of computation time and performance, model enhancement techniques were used.

2.1. Siamese Neural Network Algorithm

A Siamese neural network [12] refers to a type of neural network architecture that consists of two or more subnetworks that are identical to each other. In this context, ‘identical’ refers to having the same configuration, parameters, and weights. The parameter updating is mirrored across both sub-networks. Artificial neural networks are utilized to determine the similarity between inputs by comparing their feature vectors. As a result, these networks find application in numerous domains [13].

Neural networks are nearly perfect at every task in the deep learning era of today, but they need more data to do so. However, in certain cases such as FRobtaining more data may not always be feasible. To address these challenges, a novel neural network architecture known as Siamese neural networks has been developed [14]. It utilizes a limited number of images to improve its predictions. Moreover, SNN have gained popularity in recent years due to their remarkable ability to learn from minimal amounts of data.

In this study, the Siamese network consists of 2 identical neural networks, each with VGG-16 as the backbone, that receive the input images. The last layers of these networks are subsequently connected to a binary cross-entropy loss function [15], responsible for measuring the similarity between the two inputs. To ensure simultaneous learning of identical features for both input images, the twin networks share the same configuration and parameters [16].Two scenarios are presented in this study; first scenario is having 2 images of the same person “positive pair”. Second scenario is feeding 2 images of different individuals “negative pair”. Through training, the network becomes capable of differentiating between positive and negative

pairs, which is also used to compare a new, unseen image with a stored image for authentication purposes. As an output, the network gives out a metric that quantifies how much the two input images are different. This metric is then compared with a threshold to decide whether the two images are of the same person or not. The threshold is determined based on the distribution of distances observed during the training phase. To summarize, the SNN with VGG-16 as the backbone presents a reliable and effective approach for FR. Its ability to handle diverse factors like pose, lighting, and expression renders it well-suited for practical applications, including the specific scenario described in this study. As seen in Figure 1 explain the architecture of SNN with VGG-16, EfficientNetB0, ConvNeXtBase and ResNet50 model as a backbone.

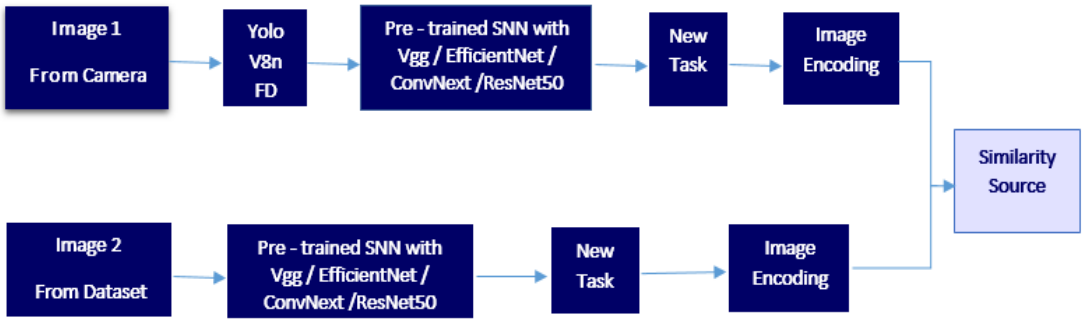


Figure 1. The Architecture of SNN with VGG-16, EfficientNetB0, ConvNeXtBase and ResNet50 Model as a Backbone

2.2. VGG-16 Network

VGG-16 [17], named after the visual geometry group (VGG) at oxford, is a convolutional neural network (CNN) architecture that has proven to be one of the most effective vision model architectures to date. It consists of 16 layers with weights, making it a relatively extensive network with a total of 138 million parameters. The architecture of VGG16 is simple and incorporates the most important features of convolutional neural networks. It uses small 3×3 convolution filters and a stride of 1, which are in the same padding, and a max pool layer of 2×2 filters for stride 2. This arrangement of convolution and max pool layers is consistent throughout the whole architecture. The network also includes three fully connected layers. the simplicity of the VGG-16 architecture is its main attraction, and its ability to handle diverse factors like pose, lighting, and expression makes it well-suited for practical applications [18]. As seen in Figure 2 illustrates the general architecture of the VGG-16 network before transfer learning.

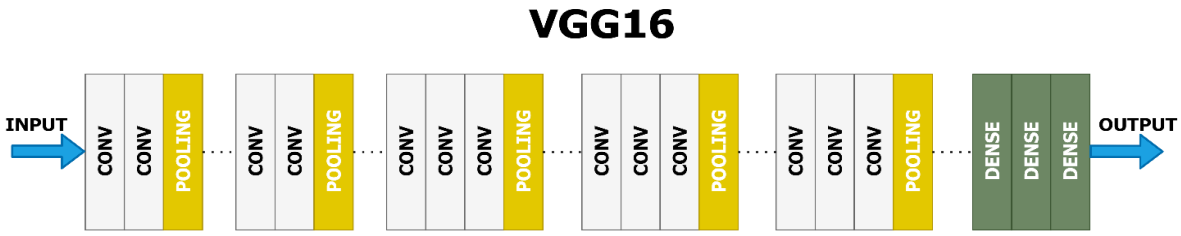


Figure 2. The General Architecture of the VGG-16 Network

To perform the transfer learning process, we freeze the layers to preserve the weights that were previously trained on ImageNet dataset, then separate the Vgg16 head, which consists of three fully connected layers and the final CNN, and replace them with SNN layers to perform the recognition process by adding relu as activation function As seen in Figure 3.

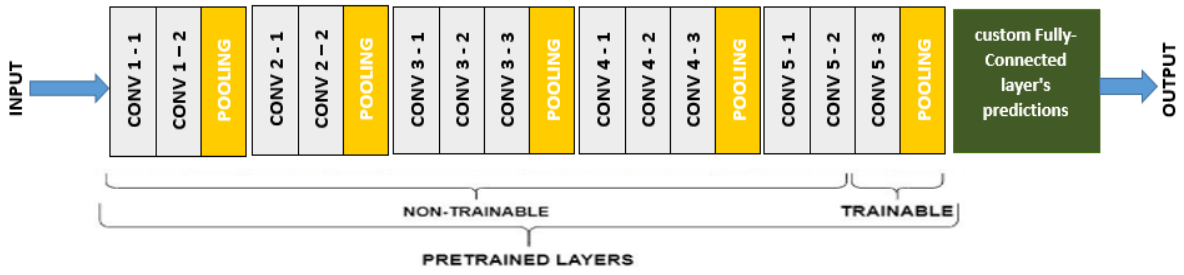


Figure 3. Adding New Layers on the Top of the VGG-16 Model [19]

2.3. EfficientNetB0

EfficientNetB0 [20], is a convolutional neural network architecture, It solves the challenging task of reconciling accurate modeling with speed of computation, with a target of reaching state-of-the-art performance with significantly fewer the parameters and utilized resources than previous versions. The design makes certain that all network parameters modify in a consistent manner, improving performance while decreasing computation weight. The primary feature of EfficientNet is its composite scaling mechanism, which constantly changes the network's depth, width, and accuracy according to a set of predefined scaling factors. EfficientNet includes a hierarchy of layers formed comprised of repeating blocks that gradually improve the network's depth, width, and resolution. EfficientNet-B0 that used in this study has 7 layers in each block. EfficientNet-B0 has fewer layers compared to EfficientNet-B7 has 19 layers in each block. Once the architecture lengths, a total number of layers increases significantly. After construct the SNN, EfficientNetB0's layers should be frozen except for the last layers. Train the SNNs using the LFW dataset while updating the unfrozen layers' weights. Figure.4 shows the general architecture of the EfficientNetB0. As well Figure 5 shows the transfer learning.

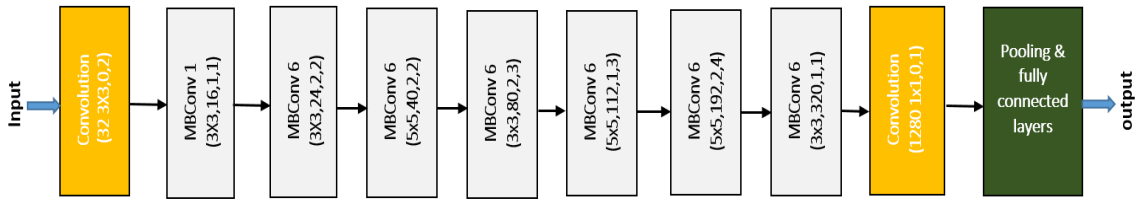


Figure 4 The general architecture of the EfficientNetB0 model [20].

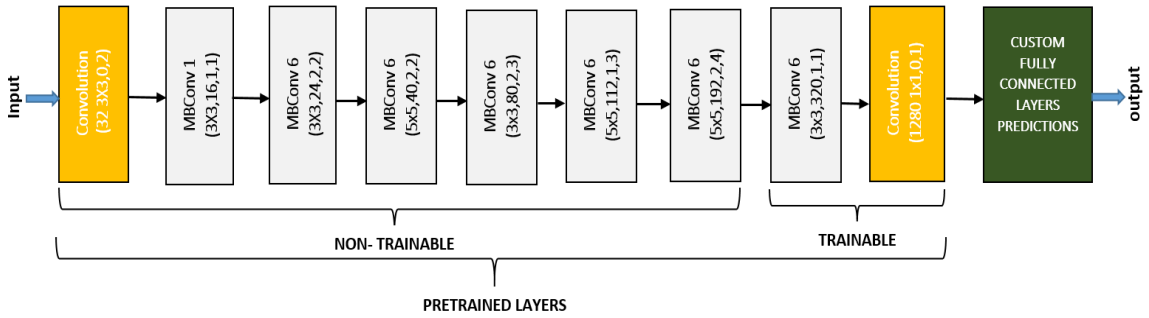


Figure 5. Adding New Layers on the top of the EfficientNetB0 Model [20]

2.4. ResNet50

ResNet50 is an architecture for convolutional neural networks that was first presented by [21]. It belonged to the ResNet (Residual Network) family, which has been developed to use residual learning to solve the degradation issue that extremely deep networks face. Furthermore, ResNet50 was developed using a fundamental component architecture and has 50 layers As seen in Figure 6 and Figure 7 show the trainable model.

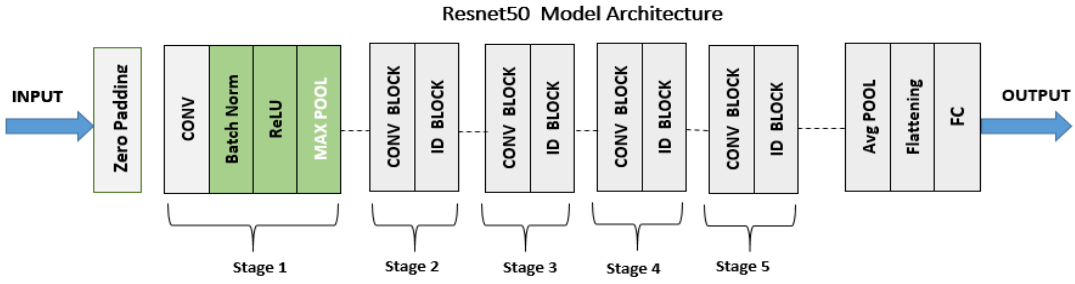


Figure 6. The General Architecture of Baseline ResNet50 Model

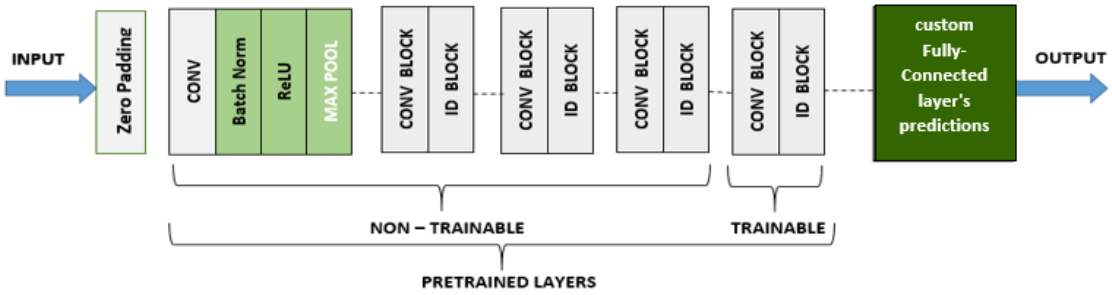


Figure 7. Adding New Layers on the Top of the. Baseline ResNet50 Model

2.5. ConvNeXtBase

ConvNets [22] another developed categorized image method. The ConvNeXtBase architecture is a convolutional neural network (CNN) design that improves traditional CNNs by using structured set convolutions. It is proposed as part of the "Aggregated Residual Transformations for Deep Neural Networks" (ResNeXt) [21], framework, which seeks to increase model performance and efficiency. The number of layers in a ConvNeXt network varies based on the version and architecture configuration, with a preference for residual block composition and pooling convolutions over a fixed overall number of layers. Consider using a moderately deep architecture, such as ResNeXt-50 or ResNeXt-101 [21], for FRtasks that use faces tagged in the Wild (LFW) dataset, which comprises a very limited number of face photographs per subject (typically approximately 10). These architectures find a mix between model complexity and computational efficiency, making them ideal for workloads that require medium-sized datasets, such as LFW. After loading the pre-trained ConvNeXt model, drop its last fully connected layer. This layer is typically used for classification tasks and is not required for feature extraction in the SNN weights As seen in Figure 8 [22].

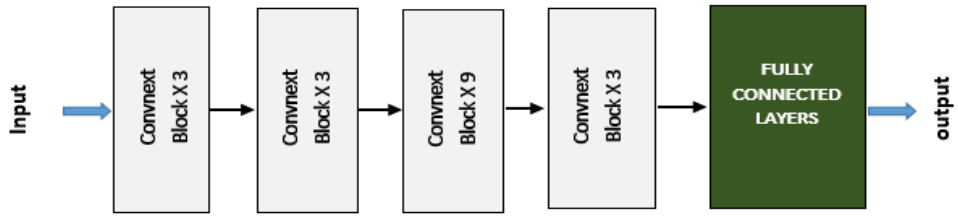


Figure 8. The General Architecture of ConvNeXtBase Model

2.6. Labelled Faces in the Wild (LFW) Dataset

The labelled faces in the wild (LFW) dataset [23] is a collection of face photographs specifically gathered for the purpose of studying unconstrained FR. The dataset comprises more than 13,000 facial images that have been gathered from various sources on the internet. The name of each person in the image was labelled on their respective face in this data set. The LFW dataset consists of four distinct sets of images. These sets include the original images as well as three types of aligned images. These aligned images are specifically designed to evaluate algorithms in various conditions. the dataset utilizes funnelled images [24], LFW-a, and deep funnelled images [25] for alignment purposes. Most face verification algorithms tend to yield better results when using deep funnelled and LFW-a images compared to funnelled images and original images. As seen in Figure 9 illustrates the LFW image dataset.

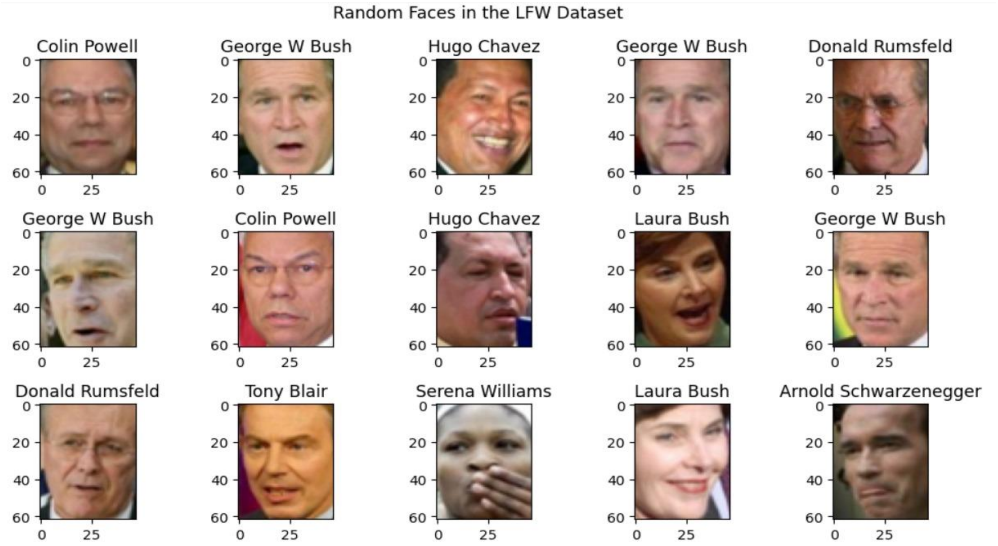


Figure 9. An Example of the LFW Image Dataset

3. Training SNN wit Transfer Learning Models

A commonly used neural network for determining the similarity between two comparable inputs is called the SNN. This study explores the possibility of two individuals having similar facial features. To ensure the highest accuracy in facial recognition and identification, it is essential to incorporate the SNN with VGG-16 as the backbone. The SNN with VGG-16 as the backbone have been trained using the LFW dataset, which contains images of 1680 individuals with two or more distinct image available. The purpose is to identify the individuals. The LFW dataset is specifically created for studying the challenge of FR in real-life scenarios, where conditions are unpredictable and include various factors such as pose, lighting, and expression variations. These conditions are commonly encountered in everyday situations.

In order to use the LFW dataset as input for the SNN, it is necessary to preprocess it. The pre-processing steps include resizing the images. The images have been resized to dimensions of 62 pixels by 47 pixels (see equation 3.4). The images are resized to decrease the computational complexity and to ensure that the network focuses on the essential features of the faces. The specific size was chosen because the dimensions of the head are always wider than its width. Since this study using VGG-16 which is a color-based model, the step to convert images to grayscale has been omitted.

$$resizedimage = resize\big(image, (62{,}47)\big) (2.4)$$

Since the LFW dataset contains limited number of samples (13,000), the proposed study utilised the Stratified K-Fold cross-validator [26] through splitting the LFW dataset into 5 sets to increase the number of dataset images and to a void the overfitting, allowing us each fold has a chance to appear in the training set (k-1) times, making sure each observation in the dataset appears in the dataset and enabling the model to learn the fundamental information distribution better. The number of observations (n= 5). Figure 10 depicts the process of splitting the utilised dataset into 5 loops.

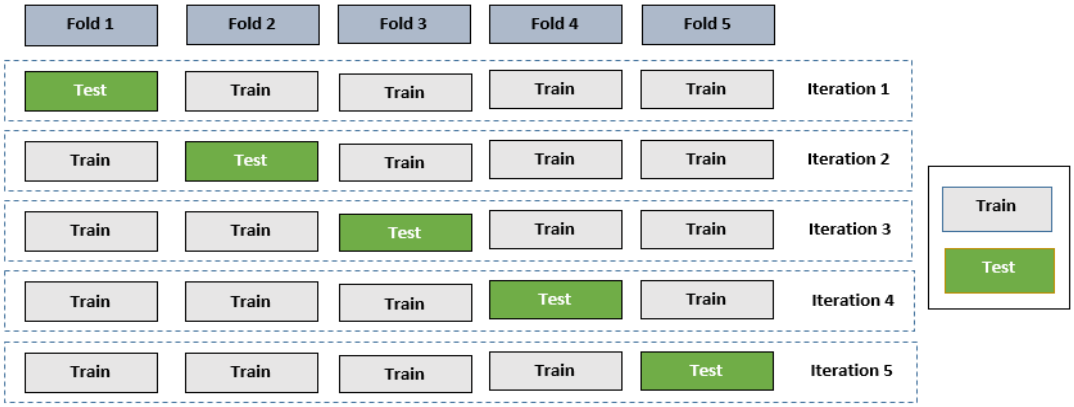


Figure 10. The Process of LFW Dataset Splitting

To evaluate the accuracy of the proposed FR approach, this study has utilised several metrics like accuracy, precision, f-measure, and recall as they are common metrics widely used [27]. False positive and false negative rates are another way to gauge the approximation error of the SNN model. These measures are well-known and commonly used in the existing

research [28], [29]. moreover, computational complexity, and memory utilization were used to evaluate the efficiency of the proposed scheme based on computation theory as it is commonly practised by the related works [30].As seen in Equations 5 , 6 , 7 , 8 and 9 were used to calculate the detection accuracy, detection rate, precision, false positive rate, and the f measure, respectively.

$$ACC = \frac{TP+TN}{TP+TN+FP+FN} (2.5)$$

$$DR = \frac{TP}{TP+FN} (2.6)$$

$$Precision = \frac{TP}{TP+FP} (2.7)$$

$$FPR = \frac{FP}{FP+TN} (2.8)$$

$$F1 = \frac{2\times Precision\times Recall}{Precision+Recall} (2.9)$$

where TP, TN, FP and FN denote the true positive, true negative, false positive and false negative respectively.

4. Experimental Results

The evaluation of the VGG16-based SNNs also yielded promising results, demonstrating its suitability for FR in the proposed model. As shown in the figure below, as the number of training epochs increased, the binary cross-entropy loss decreased, reaching lower values. This decrease in loss indicates that the network effectively learned the distinguishing features of each face, resulting in a minimal difference between the predicted and actual outputs. This effective learning makes the network as a robust tool for FR systems. Figure 11 depicts the Binary Cross Entropy Loss for the SNNs.

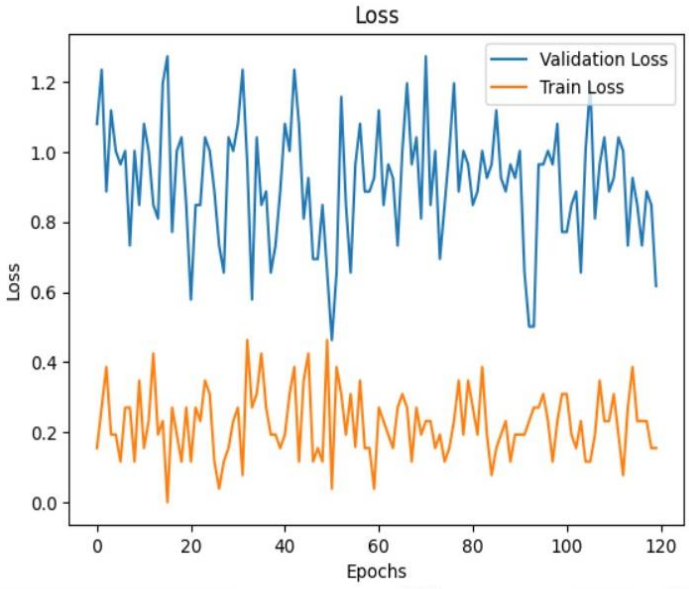


Figure 11. The Binary Cross Entropy Loss for Siamese Network

In addition, an accuracy curve was visualized showing a high accuracy score of 95% approximately on the validation set. This value signifies the ability of the model to correctly classify most of the input images. The high accuracy result renders the network suitable for the task of logging out the employees in the company. As seen in Figure 12 illustrates the SNN Accuracy with VGG16 as the backbone.

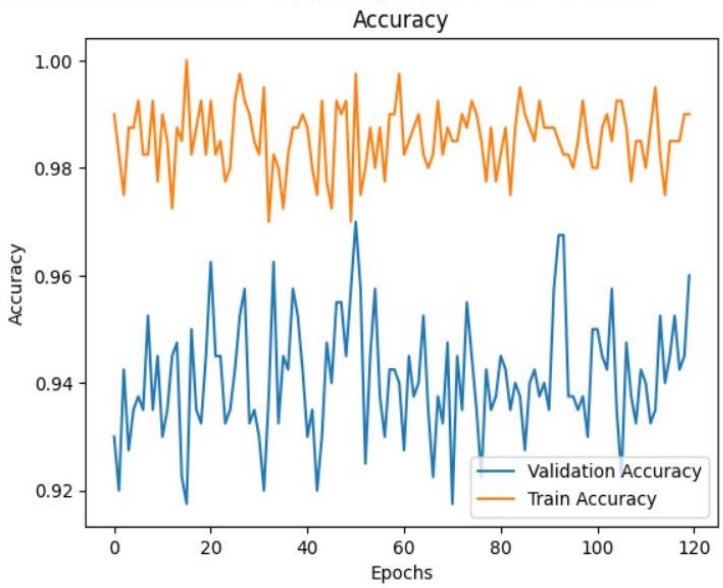


Figure 12. The Siamese Network Accuracy with VGG16 as the Backbone

Once the training of the SNN with VGG 16 was finished, it was tested on new test images to see how well the network performed. When presented with 3 images of the same individual but with different facial expressions, the network was still able to correctly identify the three images as the same person. This demonstrates the network's ability to accurately detect and group similar faces within the dataset. As seen in Figure 13 illustrates the test model on a Random Sample.

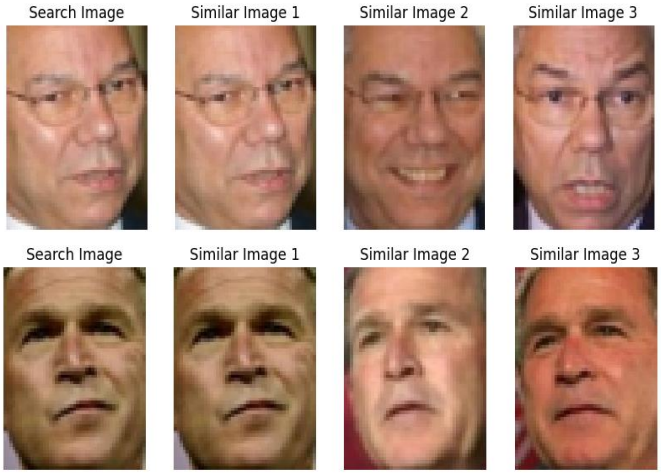


Figure 13. The test Model on Random Sample

As seen in Figure 14 below illustrates the results which had been achieved during training the proposed system using the suggested SNN with Efficient New algorithm. The achieved results explain that the proposed system has gain 78% as a validation accuracy. While as seen in Figure 15 depicts the results gained from training the proposed SNN with ConvNext, it had ganied 93% asa validation accuracy and As seen in Figure 16 show SNN with ResNet-50 has 95 %.

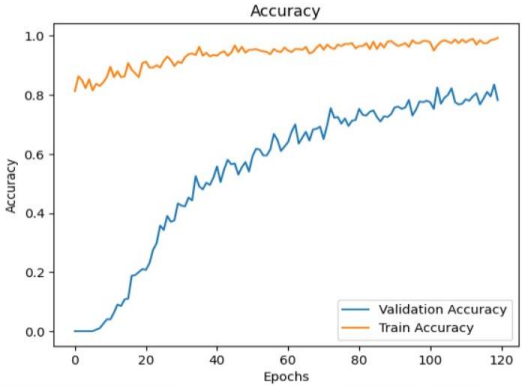


Figure 14. Siamese Network with Efficient

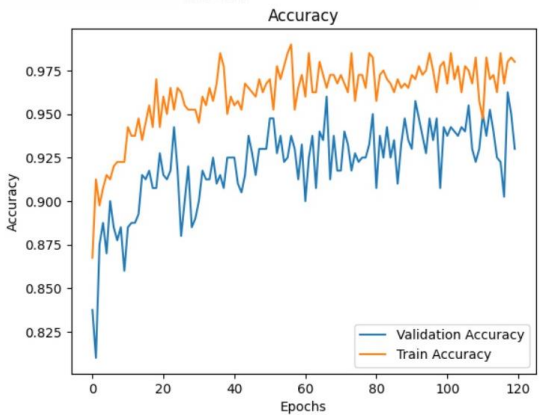


Figure 15. SNN with ConvNextTiny Algorithm

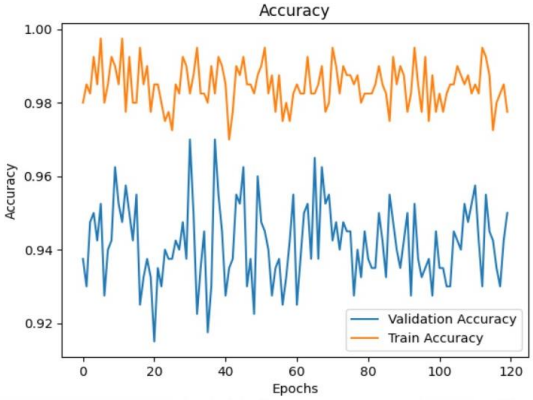


Figure 16. SNN with ResNet50 Algorithm

In this research, a SNN with VGG 16 as a backbone has been used for the FRmodel. These hybrid techniques have achieved an accuracy of 96% on the test set, which in comparison to other studies is a better level of accuracy, or at least equally as good to them. For instance, As seen in Table 1, the accuracy score of the proposed SNN with VGG 16 as a backbone is (96.00) which is much higher accuracy than some methods such as DLB (88.50), CFN+APEM (87.50), L-CSSE-KSRC (92.02), and SiameseFace1 (94.80). Other methods such as weighted PCA-EFMNet (95.00) and Siamese-VGG (95.62) show good results, that are still lower in accuracy compared to the proposed SNN with VGG 16 as a backbone. The only method that surpasses the proposed hybrid techniques in accuracy is CosFace, achieving 99.73 accuracy. However, despite achieving a lower accuracy compared to CosFace, the proposed SNN is faster, more lightweight, and suitable for running on simple hardware.

Table 4.1. SNN with VGG 16 as a Backbone Performance in Comparison to Other Methods

Reference	Method	Face Recognition ACC (%)
Chong et al [31]	DLB	88.50
Xiong et al [32]	CFN+APEM	87.50
A. Majumdar, R. Singh [33]	L-CSSE+KSRC	92.02
J. Zhang, X. Jin, [34]	SiameseFace1	94.80
B. Ameur, M. Belahcene [35]	Weighted PCA-EFMNet	95.00
M. Heidari and K.Fouladi-Ghaleh [16]	Siamese-VGG	95.62
H. Wang *et al.* [36]	CosFace	99.73
SNN with VGG 16 as a backbone		**96.00**
SNN with EfficientNetB0 as a backbone		**78.00**
SNN with ConvNextTiny as a backbone		**93.00**
SNN with ResNet50 as a backbone		**95.00**

To keep this study more robust and Efficient, two more comparative study have been utilized with VGG 16, these are Efficient loss and ConvNext networks. When evaluating the SNN's performance, it's essential to consider both accuracy and loss simultaneously. In this research, VGG16 serves as the backbone architecture. To keep the proposed method more robust, VGG16 have been benchmarked with ConvNext and EfficientNet. The performance of these networks are evaluated based on metrics such as accuracy and loss, providing insights into their effectiveness for FR tasks. By comparing the performance of different backbone architectures, this research aims to identify the most suitable model for enhancing the efficiency of FR approach. As seen in Figure 17 and 18 illustrate comparative studies between VGG16 , ResNet50 and Efficient loss and ConvNext networks. Based on the graph in Figure 17, it is not possible to definitively say which model is the best. However, the train VGG16 model appears to have the lowest loss. Based on the accuracy graph in the image, the Validation VGG16 Accuracy appears to be the highest throughout the epochs, which suggests it might be the best model for this task.

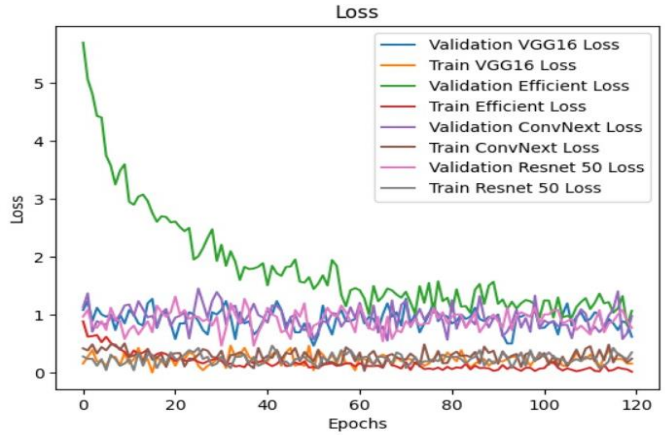


Figure 17. Comparative Study of validation Process Between VGG16 Lass, Efficient Loss and ConvNext loss and ResNet50

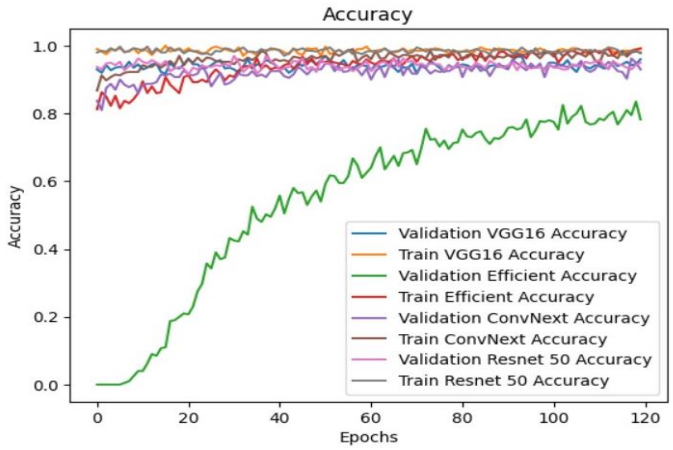


Figure 18. Comparative Study of Validation Process Between VGG16 Accuracy, Efficient Accuracy and ConvNext Accuracy

5. Conclusion

Many studies have been reported on FRin complex scenarios remains unresolved. A face detection and recognition-based system was implemented for recognition SNN with transfer learning models as the backbone. Various transfer learning architectures such as VGG-16, EfficientNet, RestNet50, and ConvNext have used for the experiments. 5-fold cross validation has been used to evaluate the performance of the architectures on LFW dataset. Comparative results show that EfficientNet, RestNet50 and ConvNext backbones achieved 78% accuracy, 95% and 93 % accuracy respectively. VGG-16 produced the best accuracy in FRwith 96%. This illustrates its capability to identify and distinguish individuals in different poses, lighting conditions, and facial expressions. For the future studies, better models can be developed to improve the accuracy of the system. Also the real-time improvements can be made to make system with fast recognition response.

References

[1] S. G. Bhandari, S. Rodrigues, P. C. Thejas, and B. S. Nausheeda, "ANALYSIS OF FRUSING LBPH ALGORITHM: A REVIEW," *Redshine Arch.*, vol. 2, 2023.

[2] I. Adjabi, A. Ouahabi, A. Benzaoui, and A. Taleb-Ahmed, "Past, present, and future of face recognition: A review," *Electronics*, vol. 9, no. 8, p. 1188, 2020.

[3] T. Gerig *et al.*, "Morphable face models-an open framework," in *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, 2018, pp. 75–82.
[4] Y.-S. Lim, S.-H. Lee, S.-J. Cheong, and Y.-H. Park, "A long-distance 3D FRarchitecture utilizing MEMS-based region-scanning LiDAR," in *MOEMS and Miniaturized Systems XXII*, 2023, vol. 12434, pp. 87–91.
[5] S. Koley, H. Roy, S. Dhar, and D. Bhattacharjee, "Illumination invariant FRusing fused cross lattice pattern of phase congruency (FCLPPC)," *Inf. Sci. (Ny).*, vol. 584, pp. 633–648, 2022.
[6] aAmal A. Moustafa, A. Elnakib, and N. F. F. Areed, "Age-invariant FRbased on deep features analysis," *Signal, Image Video Process.*, vol. 14, pp. 1027–1034, 2020.
[7] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A discriminative feature learning approach for deep face recognition," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part VII 14*, 2016, pp. 499–515.
[8] Z. Chen, X. Feng, and S. Zhang, "Emotion detection and FRof drivers in autonomous vehicles in IoT platform," *Image Vis. Comput.*, vol. 128, p. 104569, 2022.
[9] T. Sabharwal and R. Gupta, "Deep facial recognition after medical alterations," *Multimed. Tools Appl.*, vol. 81, no. 18, pp. 25675–25706, 2022.
[10] Torrey, L., & Shavlik, J. (2010). Transfer learning. In Handbook of research on machine learning applications and trends: algorithms, methods, and techniques (pp. 242-264). IGI global.
[11] Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.
[12] D. Chicco, "Siamese neural networks: An overview," *Artif. neural networks*, pp. 73–94, 2021.
[13] N. Serrano and A. Bellogín, "Siamese neural networks in recommendation," Neural Comput. Appl., pp. 1–13, 2023.
[14] Z. S. Naser, H. N. Khalid, A. S. Ahmed, M. S. Taha, and M. M. Hashim, "Artificial Neural Network-Based Fingerprint Classification and Recognition.," *Rev. d'Intelligence Artif.*, vol. 37, no. 1, 2023.
[15] U. Ruby and V. Yendapalli, "Binary cross entropy with deep learning technique for image classification," *Int. J. Adv. Trends Comput. Sci. Eng*, vol. 9, no. 10, 2020.
[16] M. Heidari and K. Fouladi-Ghaleh, "Using Siamese networks with transfer learning for FRon small-samples datasets," in *2020 International Conference on Machine Vision and Image Processing (MVIP)*, 2020, pp. 1–4.
[17] S. Tammina, "Transfer learning using vgg-16 with deep convolutional neural network for classifying images," *Int. J. Sci. Res. Publ.*, vol. 9, no. 10, pp. 143–150, 2019.
[18] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv Prepr. arXiv1409.1556*, 2014.
[19] McDermott, J. (2021). Hands-On Transfer Learning with Keras and the vgg16 Model.
[20] Tan, M., & Le, Q. (2019, May). Efficientnet: Rethinking model scaling for convolutional neural networks. In International conference on machine learning (pp. 6105-6114). PMLR.
[21] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).
[22] Liu, Z., Mao, H., Wu, C. Y., Feichtenhofer, C., Darrell, T., & Xie, S. (2022). A convnet for the 2020s. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 11976-11986).
[23] A. Jalal and U. Tariq, "The LFW-gender dataset," in *Computer Vision–ACCV 2016 Workshops: ACCV 2016 International Workshops, Taipei, Taiwan, November 20-24, 2016, Revised Selected Papers, Part III 13*, 2017, pp. 531–540.
[24] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database forstudying FRin unconstrained environments," 2008.
[25] K. Sohn, H. Lee, and X. Yan, "Learning structured output representation using deep conditional generative models," *Adv. Neural Inf. Process. Syst.*, vol. 28, 2015.
[26] Stone, M. (1974). Cross-validatory choice and assessment of statistical predictions. Journal of the royal statistical society: Series B (Methodological), 36(2), 111-133.
[27] X. Li, Y. Xiang, and S. Li, "Combining convolutional and vision transformer structures for sheep face recognition," *Comput. Electron. Agric.*, vol. 205, p. 107651, 2023.
[28] P. Grother, M. Ngan, and K. Hanaoka, *FRvendor test (fvrt): Part 3, demographic effects*. National Institute of Standards and Technology Gaithersburg, MD, 2019.
[29] J. J. Howard, E. J. Laird, R. E. Rubin, Y. B. Sirotin, J. L. Tipton, and A. R. Vemury, "Evaluating proposed fairness models for FRalgorithms," in *International Conference on Pattern Recognition*, 2022, pp. 431–447.
[30] M. Zulfiqar, F. Syed, M. J. Khan, and K. Khurshid, "Deep FRfor biometric authentication," in *2019 international conference on electrical, communication, and computer engineering (ICECCE)*, 2019, pp. 1–6.
[31] S.-C. Chong, A. B. J. Teoh, and T.-S. Ong, "Unconstrained face verification with a dual-layer block-based metric learning," *Multimed. Tools Appl.*, vol. 76, pp. 1703–1719, 2017.
[32] C. Xiong, L. Liu, X. Zhao, S. Yan, and T.-K. Kim, "Convolutional fusion network for face verification in the wild," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 3, pp. 517–528, 2015.
[33] A. Majumdar, R. Singh, and M. Vatsa, "Face verification via class sparsity based supervised encoding," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1273–1280, 2016.
[34] J. Zhang, X. Jin, Y. Liu, A. K. Sangaiah, and J. Wang, "Small Sample FRAlgorithm Based on Novel Siamese

Network.," *J. Inf. Process. Syst.*, vol. 14, no. 6, 2018.
[35] B. Ameur, M. Belahcene, S. Masmoudi, and A. Ben Hamida, "Weighted PCA-EFMNet: A deep learning network for Face Verification in the Wild," in *2018 4th International Conference on Advanced Technologies for Signal and Image Processing (ATSIP)*, 2018, pp. 1–6.
[36] H. Wang *et al.*, "Cosface: Large margin cosine loss for deep face recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 5265–5274.

Conflict of Interest Notice
The authors declare that there is no conflict of interest regarding the publication of this paper.

Ethical Approval and Informed Consent
It is declared that during the preparation process of this study, scientific and ethical principles were followed, and all the studies benefited from are stated in the bibliography.

Availability of Data and Material
Not applicable.

Plagiarism Statement
This article has been scanned by iThenticate ™.

SAKARYA UNIVERSITY JOURNAL OF COMPUTER AND INFORMATION SCIENCES
http://saucis.sakarya.edu.tr/
e-ISSN: 2636-8129
Publisher: Sakarya University
Vol. 7, No. 3, 2024, 439-448
DOI: 10.35377/saucis...1522290
SAUCIS
SAKARYA UNIVERSITY

RESEARCH ARTICLE

Enhancing Fall Detection Accuracy: The Ground-Face Coordinate System for 3D Accelerometer Data

Abdullah Talha Sözer

Karabuk University, Faculty of Engineering, Electrical and Electronics Engineering Department, Karabuk, Türkiye

Corresponding author:
Abdullah Talha Sözer,
Karabuk University, Faculty of Engineering, Electrical and Electronics Engineering Department, Karabuk, Türkiye
talhasozer@karabuk.edu.tr

Article History:
Received: 25.07.2024
Accepted: 21.10.2024
Published Online: 24.12.2024

ABSTRACT
The global elderly population is on the rise, leading to increased physical, sensory, and cognitive changes that heighten the risk of falls. Consequently, fall detection (FD) has emerged as a significant concern, attracting considerable attention in recent years. Utilizing 3D accelerometer sensors for FD offers advantages such as cost-effectiveness and ease of implementation; however, traditional raw 3D accelerometer signals are inherently dependent on the device's orientation and placement within the device coordinate system. Misalignment between the device's axes and the direction of movement can lead to misinterpretation of acceleration signals, potentially causing misclassification of activities and resulting in false positives or missed falls. This study introduces a novel coordinate system called "ground-face," which is designed to be independent of the device's orientation and placement. In this system, the vertical axis is aligned perpendicularly to the Earth, while the device's x-axis is aligned with the individual's direction of movement. To assess the potential of the vertical component of ground-face referenced accelerometer signals for FD, it was compared with the commonly used acceleration magnitude signal. Detailed analysis was conducted using frequently preferred features in FD studies, and fall detection was performed with various classifiers. Comprehensive experiments demonstrated that the vertical component of the ground-face signal effectively characterizes falls, yielding approximately a 2% improvement in detection accuracy. Moreover, the proposed coordinate system is not limited to FD but can also be applied to human activity recognition (HAR) systems. By mitigating orientation-related discrepancies, it reduces the likelihood of misclassification and enhances the overall HAR capabilities.

Keywords: Fall detection, Accelerometer, Device coordinate system, Global coordinate system, Movement direction

1. Introduction

The World Health Organization defines a fall as the involuntary change in position from a person's current location to a lower position, such as the ground. It is reported that each year, falls lead to 684,000 deaths, making it the second-largest cause of accidental death after road traffic accidents. In addition to fatal falls, there are 37.3 million severe falls requiring medical intervention annually, which can result in lasting injuries [1]. Furthermore, treatments after falls are among the most costly medical interventions [2]. Considering that the risk of falls and severe consequences is higher in elderly individuals and the global elderly population is increasing, falls emerge as a growing issue.

Approximately 12% of the world's population is comprised of individuals over the age of 60, and it is projected that this proportion will reach 16% by the year 2030 and 22% by 2050 [3]. Aging brings about physical, sensory, and cognitive changes that increase the likelihood of falling. Serious injuries or fatalities due to falls are frequently observed among elderly individuals. For instance, in the United States, 20-30% of elderly individuals who experience falls suffer from moderate to severe injuries such as bruises, hip fractures, or head trauma [1]. Every year, 28% of individuals aged 65 and above, as well as 32% of those aged 70 and above, experience falls. Moreover, the elderly constitute the demographic most affected by fatal falls [4].

To address this significant health threat, extensive efforts are being dedicated to systems equipped with automatic fall detection (FD) and alert functions. Automatic FD systems are capable of identifying falls and promptly alerting hospitals or caregivers [5]–[13]. These systems have the potential to reduce the time between a fall and medical intervention, thereby minimizing health complications related to falls.

FD systems primarily use two approaches – wearable-based and environment-based – to distinguish between falls and daily activities. The environment-based method employs technologies such as cameras, infrared cameras, Kinect sensors, microphones, motion, radar, pressure, and vibration sensors. Meanwhile, the wearable-based approach incorporates sensors

Cite as: A. T. Sözer, "Enhancing Fall Detection Accuracy: The Ground-Face Coordinate System for 3D Accelerometer Data," *Sakarya University Journal of Computer and Information Sciences*, vol.7, no. 3, pp. 439-448, 2024. Doi: 10.35377/saucis…1522290

for acceleration, pressure, orientation, magnetic fields, and heart rate monitoring [2], [14]–[22].

Using an inertial measurement unit (IMU), particularly a 3-axis accelerometer, for FD offers advantages such as cost-effectiveness and ease of setup. Furthermore, it is common for human activity recognition (HAR) systems to be equipped with accelerometers. By combining the signals generated by the accelerometer with machine learning approaches, HAR and FD systems with high detection capacity have been developed [23]–[28]. An accelerometer generates signals by detecting the acceleration forces acting along the three axes of the device coordinate system: x, y, and z. It uses internal components like microstructures that respond to physical forces, converting them into electrical signals that can be interpreted as acceleration data. The signals in the device coordinate system make accelerometer signals sensitive to the device's placement and orientation. In simpler terms, when the device is attached in different positions, like on various points on the waist, or during actions such as falling where the device's orientation changes, variation of the acceleration can also be observed in the device's axes that don't align with the direction of movement. Moreover, an accelerometer can exhibit an orientation gap between its package and the physical sensors [29]. This variation in acceleration data due to changes in device placement and orientation introduces a significant limitation for FD and HAR systems. Misalignment between the device's axes and the direction of movement may lead to misinterpretation of acceleration signals, resulting in misclassification of activities, which can cause false positives or missed detections. This makes it critical to develop solutions that compensate for these orientation-induced discrepancies to ensure more accurate detection of falls and other human activities.

A common solution to this issue is often using the norm of the 3D accelerometer signal. This signal, named the acceleration magnitude, is commonly utilized in FD studies and remains unaffected by device placement and orientation [30]–[35]. Another approach to address the aforementioned issue involves expressing accelerometer signals in different coordinate planes. In the global coordinate system, the x-axis becomes tangent to the Earth and points to the East; the y-axis becomes tangent to the Earth but points to the Earth's North Pole; and the z-axis becomes perpendicular to the Earth and points toward the sky. The downward signal obtained by projecting accelerometer signals onto the global coordinate system has been utilized in FD studies [36]–[38]. Another proposed coordinate system is the user-centric coordinate system. In this system, accelerometer signals are first projected onto the global coordinate system and then rotated to user directions calculated from the instantaneous velocity of the user [39].

This study proposes a novel coordinate system named 'ground-face' for potential use in FD and HAR studies. In this system, the vertical axis becomes perpendicular to the Earth, akin to the global coordinate system, while the device's x-axis is aligned with the person's direction. During the study, accelerometer signals referenced to the device were mapped onto this ground-face coordinate system. The FD potential of the resulting downward signal was evaluated by comparing it with the acceleration magnitude. Thorough experiments illustrated that the downward signal more accurately characterizes falls and holds promise for effective FD.

Key contributions of the study:

1. A novel coordinate system has been developed in the proposed framework, where the vertical axis is aligned with the ground, and the x-axis corresponds to the person's direction. This results in a coordinate system that better suits the nature of human movements, regardless of device orientation and placement. As a result, it is well-suited for HAR and FD.
2. A method has been proposed to calculate the angular gap between a person's direction and the device's orientation.
3. The appropriateness of the proposed coordinate system for FD has been affirmed through a comprehensive comparison between the downward signal and the acceleration magnitude.

The subsequent sections of this article are organized as follows: Section II elaborates on the detailed introduction of the projection of device-referenced signals onto the proposed coordinate system and its application in FD. Section III presents a comprehensive performance evaluation. Section IV delves into the experimental findings and the merits and limitations of the suggested coordinate system. Section V concludes the article, offering insights into future work.

2. Materials & Methods

In this section, the calculation of the ground-referenced accelerometer signal is provided. Subsequently, the tools necessary to assess the suitability of the acquired signal for FD are elucidated.

2.1. Ground-Face Coordinate System

The signals generated by 3D accelerometer devices encompass both dynamic and static components. The dynamic component originates from movement, while the static component arises due to the Earth's gravitational force. These signals are produced by accelerometer devices with reference to device coordinate systems. Given their device-referenced nature, during scenarios involving orientation changes such as falls, dynamic and static accelerations manifest in different axes due to the person's orientation (Figure 1(a)). Additionally, the placement of the device could lead to a discrepancy between the direction of movement and the device's orientation. Analyzing these acceleration signals as ground-face referenced instead of device-referenced can offer greater insights into the occurrence of a fall event. Figure 1(b) displays the suggested ground-face coordinate system.

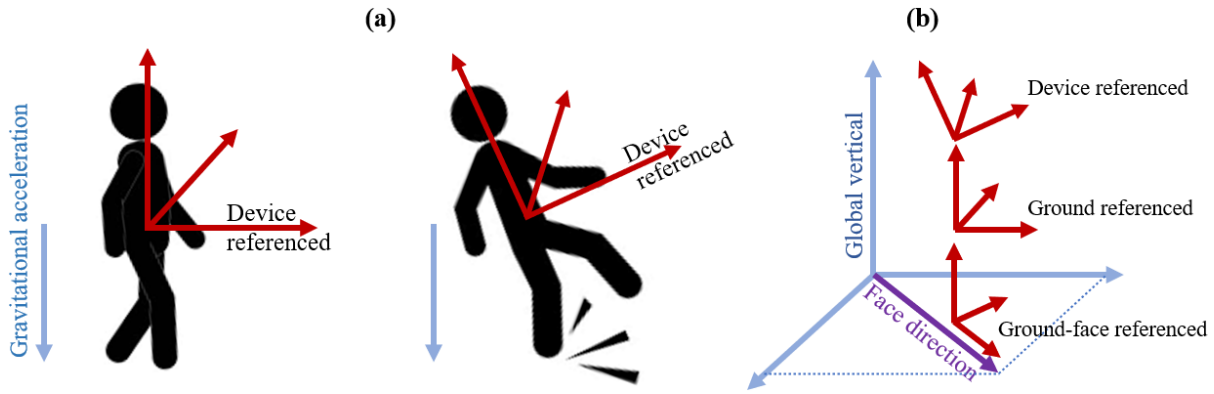


Figure 1. Proposed Ground-Face Coordinate System. (a) Impact of Rotation (b) Various Coordinate Systems. Raw Acceleration Signals Are Initially in the Device Coordinate Systems. In the Ground Coordinate Systems, the x-y Plane Lies Parallel to the Earth's Surface. In the Suggested Ground-Face Coordinate System, the x-y Plane Also Remains Parallel to the Earth's Surface, But the x-axis Aligns with the Direction of the User's Face.

Ground-face referenced signals are derived from signals referenced to the device through a transformation process.

$$A' = RA^T \tag{1}$$

Here $A = [a_x, a_y, a_z]$ holds the accelerometer readings along the device-referenced x, y, and z axes. R represents the rotation matrix, and $A' = [a'_x, a'_y, a'_z]$ denotes the transformed accelerometer values. In a three-dimensional context, the rotation matrix R is composed of three-axis rotations, $R_x(\psi), R_y(\theta), R_z(\beta)$, which are defined by Euler angles. The rotation matrix R is

$$R = R_z(\beta)R_y(\theta)R_x(\psi) \tag{2}$$

The angles ψ and θ can be calculated through sensor fusion involving the accelerometer and gyroscope at each time step. On the other hand, the β angle, a fixed value, is intended to align the device direction with the movement direction. To estimate the angle β, a calibration process is required. During this process, a person performs a straight walk. The estimation of this angle relies on determining the device's direction using principal component analysis (PCA). PCA identifies the axis along which the most significant variation occurs. During a straight walk, the direction of maximum variance calculated by PCA will correspond to the direction of the device, as the largest changes are expected to be in the forward (face) direction. The difference between the face direction and the device's direction yields the angle β which is calculated as follow:

$$\beta = cos^{-1}\frac{\langle \hat{x}, pc \rangle}{||\hat{x}||_2 ||pc||_2} \tag{3}$$

Here, $\hat{x}$ represents a unit vector in the x-direction, which is [1 0], and pc is the principal component of $R_y(\theta)R_x(\psi)a_x^T$ and $R_y(\theta)R_x(\psi)a_y^T$ which correspond to the ground-referenced x and y axes components.

In summary, the device-referenced x and y-axis signals are transformed into ground-referenced format, and the angle β is determined with the help of PCA.

As a result of the transformation, the signals a'_x and a'_y respectively carry information about the forward and lateral components of a person's movement, whereas the signal a'_z provides information about components perpendicular to the ground. The signals in A' can be utilized for HAR and FD.

2.2. Relevant Features for Fall Detection

To demonstrate the functionality of A' for FD, a comparison between device referenced and ground-face referenced accelerometer signals was necessary. The sum vector magnitude, $||\boldsymbol{a}|| = \sqrt{a_x^2 + a_y^2 + a_z^2}$, is frequently employed in accelerometer-based FD algorithms. This choice of signal is primarily due to its independence from device position and orientation. On the other hand, in the ground-referenced signal, fall-related information is carried by downward signal, a'_z. Hence, a comparison was made between a'_z and $||\boldsymbol{a}||$ in terms of the insights they provided into the fall event.

Illustrated in Figure 2, the timeframe consisting of one second before the peak and the subsequent 0.3 seconds has been identified as the critical phase. For the purpose of comparison, we selected several features: maximum, minimum, and vertical displacement during the pre-impact phase, range, standard deviation of the critical phase, and maximum falling speed.

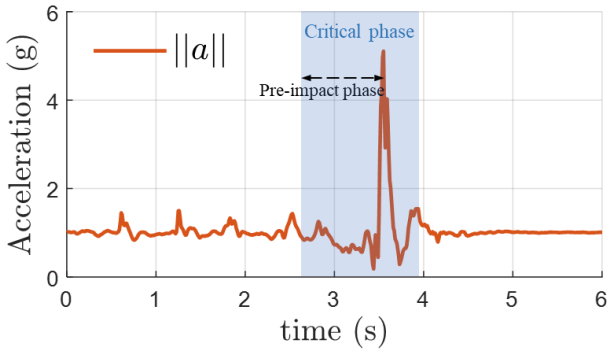


Figure 2. The Phases of Falls Utilized for Feature Extraction

2.3. Assessment Criteria

To assess the information carried out by each feature related to the fall event, Mutual Information Criteria (MIC) values were obtained. The mutual information quantifies the amount of knowledge a feature offers regarding the fall event. For each feature, Receiver Operating Characteristic (ROC) curves were plotted, illustrating the trade-off between sensitivity (fall detection capacity) and specificity (capacity to avoid wrong fall alarm performance). Furthermore, Area Under the Curve (AUC) values, which indicate the overall performance of a binary classification model by calculating the area under the ROC curve, were computed.

Additionally, the signals were compared to the classification accuracy achieved by machine learning models. Support Vector Machine (SVM), k-Nearest Neighbors (KNN), and Random Forest (RF) algorithms were employed, and values for sensitivity, specificity, and accuracy were obtained.

2.4. Dataset

The proposed ground-face reference system has been tested with a public, extensive fall dataset, KFall [40]. The dataset was obtained from 32 young participants who performed 21 activities of daily living (ADLs) and 15 types of falls. An inertial sensor was attached to their lower back to capture the movements. The dataset comprises 5075 motion files, including 2729 ADL and 2346 fall events. Each file provides acceleration, angular velocity, and Euler angle samples along three axes.

3. Experimental Results

The proposed method was tested by rotating the device-referenced signal to the ground-face reference system and its usage on FD.

3.1. Calibration Stage

Accelerometer signals referenced to the ground-face coordinate system were captured during the calibration phase. This phase involved measuring acceleration and orientation angles during straight walks. An individual positioned an Android mobile phone on their waist at various angles and performed straight-line walks. The phone was placed at five different angles, and the individual completed five 5-second walks. Acceleration data, along with the rotation angles ψ and θ, were recorded from the phone. Figure 3 shows the Android device's axes and the corresponding acceleration measurements.

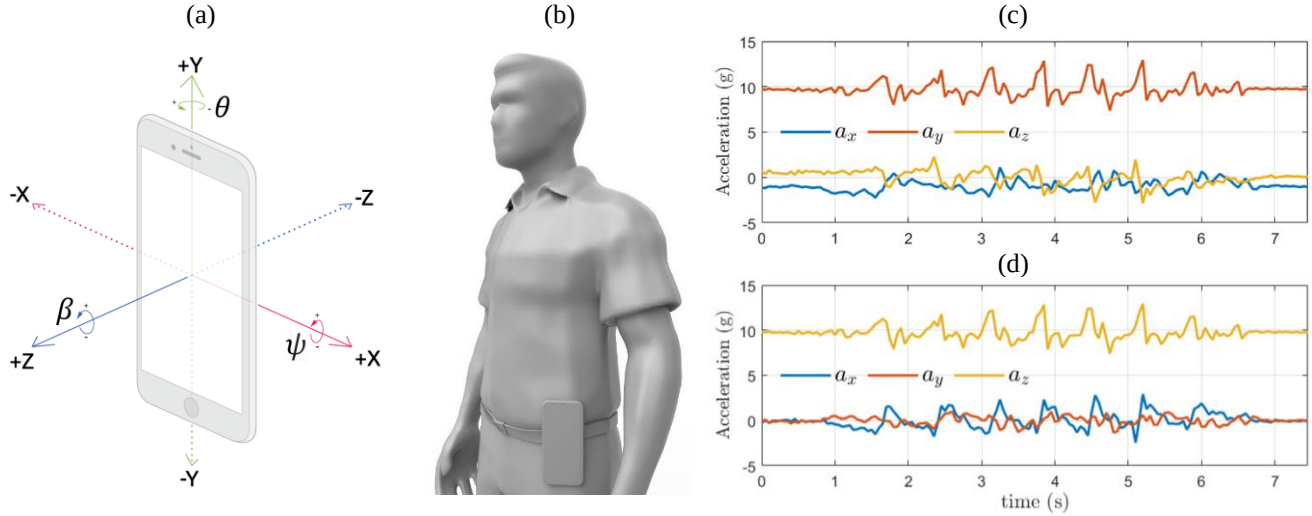


Figure 3. Calibration Experiments. (a) Android Device Axes for Acceleration and Rotation. (b) The Device Placed at Various Angles on the Individual's Waist. (c) Acceleration Signals Relative to the Device, with Average Rotation Angles of $\psi = -89°$ and $\theta = -6°$. (d) Ground-Face-Referenced Acceleration Signals, with Gravity Observed Solely on the Z-Axis. The Calculated β Angle was 63°.

Ground-referenced signals were obtained from the acceleration values and ψ and θ rotation angles measured from the phone positioned at angles around the waist as specified by Real β in Table 1. Using these values, the β angles shown in Table 1 were calculated, where 0° corresponds to the right side of the waist and 180° to the left side. A review of Table 1 reveals that the calculated values are very close to the real values. In Figure 3(d), examining the x and y components of the ground-face referenced accelerometer signal indicates that the forward movement during walking is reflected in the x-axis.

Table 1. Real and Calculated β Angles

Real β	15°	55°	90°	115°	155°
Calculated β	$9.5° \pm 4.7°$	$56.6° \pm 5.1°$	$90.1° \pm 4°$	$113.8° \pm 1.8°$	$162.8° \pm 6.6°$

3.2. FD by Downward Signal

The downward signal, perpendicular to the Earth in the ground-face referenced acceleration signals, was analyzed for its FD potential by comparing it to the acceleration magnitude. The accelerometer signals during a fall are depicted in Figure 4. Device-referenced signals are shown in Figure 4 (a), while ground-face referenced signals obtained through transformation are shown in Figure 4 (b). In Figure 4 (c), the acceleration magnitude, $||\boldsymbol{a}||$, is illustrated, and in Figure 4 (d), the downward signal, a'_z, signal is depicted. In Figure 4 (a), a high magnitude is visible along the y-axis during impact, while in Figure 4 (b), a high magnitude is observed along the z-axis. Upon comparing Figure 4 (c) & (d), it becomes apparent that the acceleration change during a fall is more pronounced in the a'_z signal.

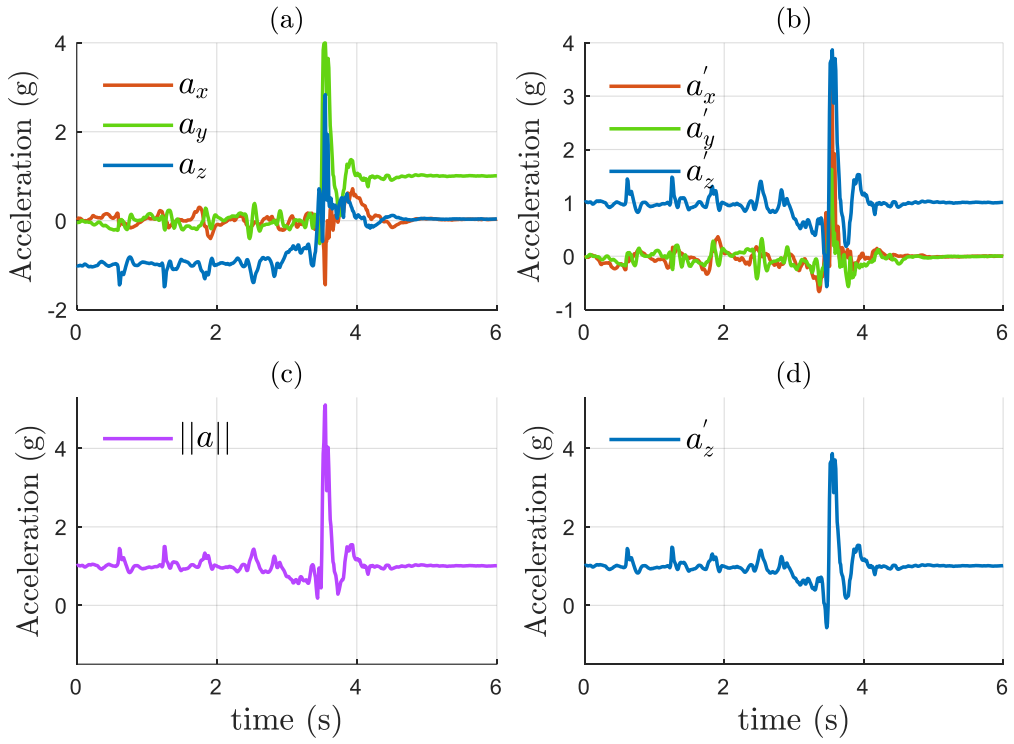


Figure 4. Device and Ground-Face Referenced Accelerometer Signals During a Lateral Fall While Walking. (a) Device Referenced One (b) Ground-Face Reference One (C) Acceleration Magnitude (D) Downward Signal

Detailed analyses and FD detection were performed using the feature values obtained for $||\boldsymbol{a}||$ and a'_z during the critical phase, and the results are presented in the following tables and graphs. In Table 2, you can see the MIV and AUC values derived from $||\boldsymbol{a}||$ and a'_z. In the results of both metrics, it's evident that the a'_z signal better represents falls. Notably, there is a significant difference in MIV and AUC values for the vertical displacement and max falling speed features.

Table 2. The MIV and AUC Values Obtained from the Acceleration Magnitude and Downward Signals

	Mutual Information Value		**Area Under Curve**	
Feature	$\|\|\boldsymbol{a}\|\|$	a'_z	$\|\|\boldsymbol{a}\|\|$	a'_z
Max	0.370	0.367	0.936	0.934
Min	0.181	0.260	0.759	0.839
Range	0.344	0.345	0.920	0.919
Standard deviation	0.337	0.336	0.893	0.856
Vertical displacement	0.184	**0.305**	0.831	**0.907**
Max falling speed	0.331	**0.502**	0.920	**0.981**

In Figure 5 (a), the vertical displacement feature is depicted in a scatter plot, while in Figure 5 (b), the max falling speed feature is presented. As evident from the figures, it can be observed that the attributes derived from a'_z provide better discrimination between ADL and fall samples.

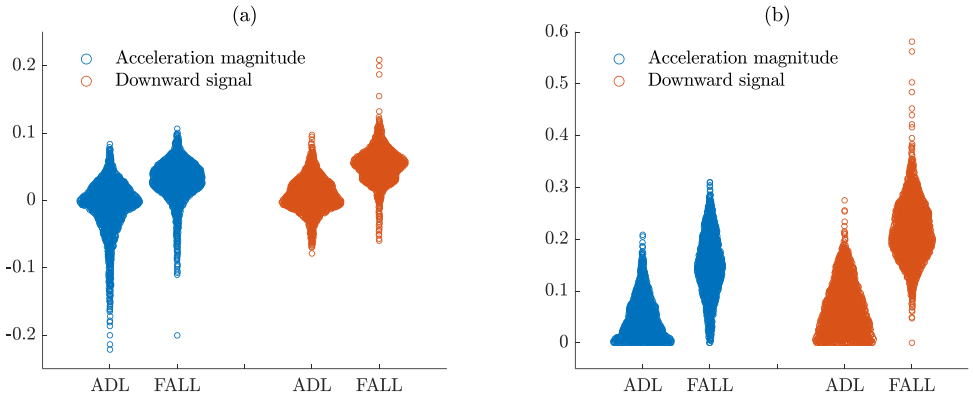


Figure 5. The Distribution of Features Obtained from Acceleration Magnitude and a'_z (a) Vertical Displacement (b) Maximum Falling Speed.

In Figure 6, ROC curves for each feature are displayed. The curves represent the sensitivity and specificity values obtained using the SVM classifier. When each feature is individually assessed, it becomes evident that the features derived from the a'_z signal carries equal or more fall-related information.

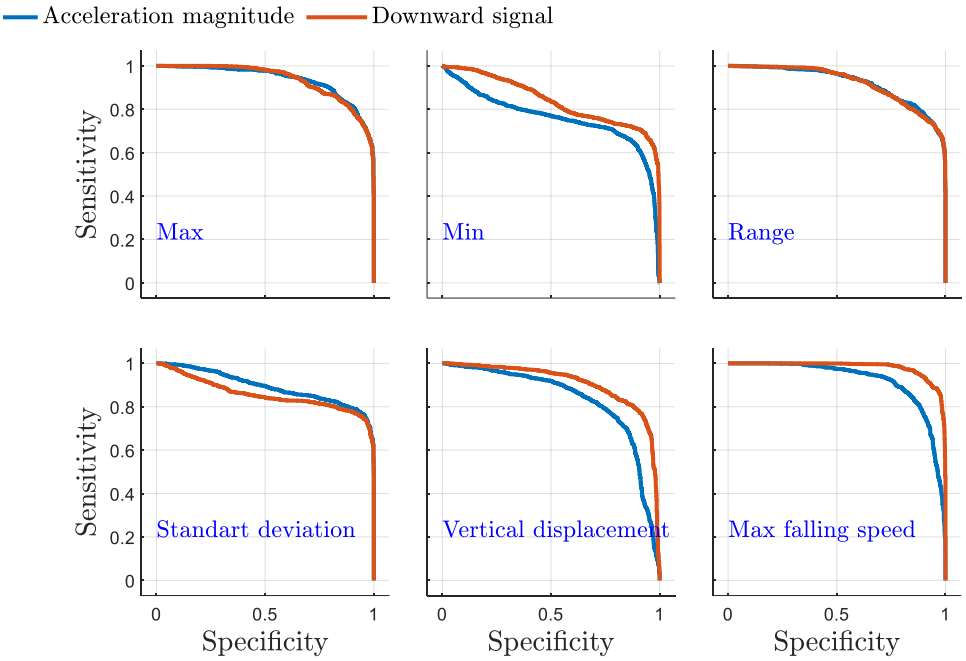


Figure 6. ROC Curves for Each Feature Obtained from Acceleration Magnitude and a'_z

Classification results using $||\boldsymbol{a}||$, a'_z and both together are shown in Table 3. Set1 represents the features derived from the $||\boldsymbol{a}||$, while Set2 represents the features derived from the a'_z. Set3 is the combination of Set1 and Set2. The results were obtained using the leave-subjects-out cross-validation approach, where in each trial, half of the subjects were chosen for training and the remaining half for testing.

Table 3. Performance Evaluation Using Various Classifiers and Feature Sets

	ACCURACY (%)			SENSITIVITY (%)			SPECIFICITY (%)		
	SET1	SET2	SET3	SET1	SET2	SET3	SET1	SET2	SET3
SVM	95.51	95.55	**97.15**	96.06	95.15	**97.03**	95.05	95.89	**97.25**
KNN	95.01	94.86	**96.26**	94.94	94.65	**95.71**	95.06	95.04	**96.72**
RF	95.27	95.41	**96.28**	95.38	95.48	**96.10**	95.18	95.34	**96.43**

Upon examining the results, it is observed that the accuracy values obtained from Set1 and Set2 are approximately the same. However, Set1 provides higher sensitivity compared to Set2, while Set2 offers higher specificity values. The combination of all features extracted from both signals has led to a significant improvement in classification performance, resulting in an increase of approximately 2% with SVM.

In order to better assess the influence of the a'_z signal on FD detection, the results of SVM classifier for each activity and fall type are displayed in Figure 7. The figure demonstrates that a'_z is particularly capable of more accurately representing certain falls and activities.

4. Discussion

Due to its advantages, such as portability and low power consumption, accelerometers are commonly preferred for FD problems. Accelerometer devices present signals in a device-referenced manner. This device-referenced signal presentation can provide information about the person's orientation, whether vertical or horizontal. However, during a fall, the individual's rotation can change, causing the downward motion-induced acceleration changes to be shared across the x, y, and z axes. Consequently, device-referenced signals may not accurately reflect downward motion. In this study, it is claimed that when the x-y plane is consistently parallel to the ground surface, acceleration values will better reflect downward motion and thus carry more information for FD.

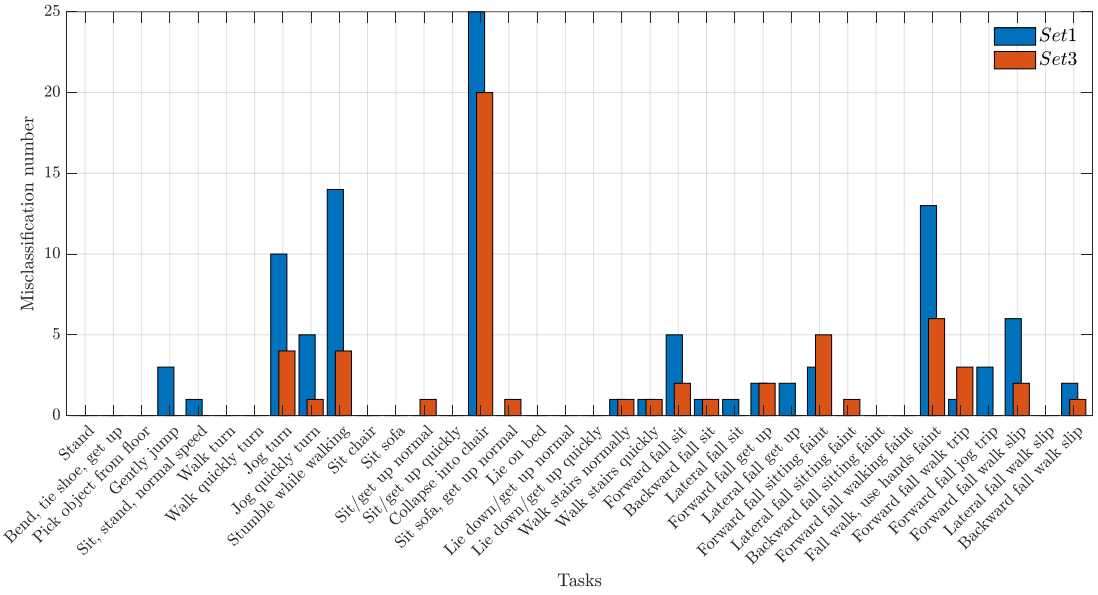


Figure 7. The Misclassification Counts of the SVM Classifier Using Set1 and Set3 for Each Activity and Fall Type in the Dataset

To address this issue, a ground-face coordinate system has been proposed. In this system, the x-y plane remains consistently parallel to the ground surface, while the z-axis remains perpendicular to the ground. In the study, ground-face referenced signals were derived from device referenced signals through a rotation process. The potential of these derived signals in FD has been investigated. For a fair evaluation, the information provided by the commonly preferred $||a||$ signal and the transformed a'_z signal regarding fall events, as well as their suitability for FD, have been extensively compared. The results of this detailed comparison are presented in the Experimental Results section.

When examining the MIV and AUC values for each feature in the table, it is evident that features derived from the a'_z signal yield either comparable or significantly better results. Notably, the downward maximum speed attribute exhibits a strong discriminative characteristic concerning fall events. Furthermore, the ROC curves depicting the trade-off between sensitivity and specificity for each feature also exhibit promising outcomes for FD using the a'_z signal.

When examining the classification results obtained with feature sets Set1, Set2, and Set3, it becomes apparent that Set1 and Set2 yield similar outcomes, while Set3 demonstrates a noteworthy increase in accuracy by up to 2%. A closer look at the sensitivity and specificity values on the Figure 8 reveals an interesting pattern: Set1 exhibits higher sensitivity, whereas Set2 showcases higher specificity. This implies that Set1 is more adept at detecting falls, while Set2 excels in recognizing non-fall situations. The integration of Set3 significantly enhances both sensitivity and specificity values. Based on this observation, it can be inferred that combining the $||a||$ signal with the a'_z signal is more suitable for FD. Figure 7 validates the recommendation to use the $\big||a|\big|$ and a'_z signals together for FD. Notable reductions in the misclassification of daily activities and falls, such as 'Jog with turn', 'Stumble while walking', and 'Sit for a moment, attempting to rise, and then collapsing into a chair,' were observed. This suggests that the a'_z signal may represent certain activities more effectively.

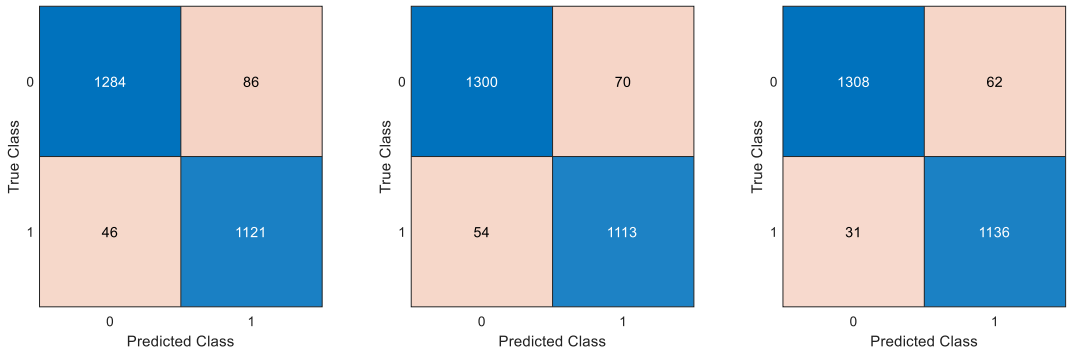


Figure 8. Comparing the Accuracy Levels Among Set1, Set2, and Set3

Regarding the difference between the proposed method and other coordinate systems: In the global coordinate system, the x-axis represents the east direction, and the y-axis points to the north pole. This leads to the x and y-axis signals being dependent on the direction of movement. In FD and HAR studies using the global coordinate system, the components of motion in the east and north pole directions do not carry any specific meaning related to the nature of the motion. In the user-centric coordinate system, the y-axis aligns with the user's direction. However, since the user's direction is estimated based on the instantaneous velocity in the 3D space, this method is more susceptible to device rotations and noise.

One limitation of the proposed method is that the performance improvements obtained from each feature did not repeat when these features were used together. This discrepancy could be limited information carried by the a'_z signal concerning ADLs. Considering the ground-face referenced signals parallel to the Earth's surface may enhance FD performance.

Another constraint of the suggested approach is its dependence on momentary angle values during the transformation process, necessitating the use of both accelerometer and gyroscope devices. In FD systems, using only accelerometers can lead to lower-cost systems at the expense of performance. However, the proposed method requires gyroscope data due to the need to express accelerometer data in a different coordinate system based on angle values. Additionally, the proposed system requires a 3x3 matrix multiplication for each measured acceleration value. Although this step is performed only once, the calibration stage increases the computational load. Thus, there is a trade-off between FD performance and cost. For systems where the FD process is performed on a computer, this computational load can be negligible, but it must be considered for embedded systems.

It should be noted that ground-face referenced signals do not replace device-referenced signals; rather, they provide additional information. This allows for more comprehensive analyses, leading to more successful FD and HAR systems.

5. Conclusion

In this study, a novel coordinate system called ground face coordinate system is proposed for accelerometer signals that can be used in IMU-based HAR and FD studies. The effectiveness of the downward signal obtained is tested in the FD problem.

Comparative experiments using the commonly preferred $||\boldsymbol{a}||$ signal in accelerometer-based FD studies have demonstrated that the use of the downward signal increases the correct detection of falls and daily activities around %2. Thus, the downward signal provides more distinctive information about falls. Therefore, the proposed coordinate system has the potential to enhance performance in both HAR and FD studies.

In future research, incorporating the evaluation of signals parallel to the ground obtained through the proposed transformation could lead to more successful systems.

References

[1] World Health Organization, “Step safely: Strategies for preventing and managing falls across the life-course,” GENEVA.

[2] R. Rajagopalan, I. Litvan, and T. P. Jung, “Fall prediction and prevention systems: Recent trends, challenges, and future research directions,” *Sensors (Switzerland)*, vol. 17, no. 11, pp. 1–17, 2017.

[3] United Nations, “World Population Ageing 2019,” New York.

[4] K. C. Liu, C. Y. Hsieh, H. Y. Huang, S. J. P. Hsu, and C. T. Chan, “An Analysis of Segmentation Approaches and Window Sizes in Wearable-Based Critical Fall Detection Systems with Machine Learning Models,” *IEEE Sens. J.*, vol. 20, no. 6, pp. 3303–3313, 2020.

[5] J. Marques and P. Moreno, “Online Fall Detection Using Wrist Devices,” *Sensors*, vol. 23, no. 3, 2023.

[6] T. Huang, M. Li, and J. Huang, “Recent trends in wearable device used to detect freezing of gait and falls in people with Parkinson’s disease: A systematic review,” *Front. Aging Neurosci.*, vol. 15, 2023.

[7] E. A. De La Cal, M. Fáñez, M. Villar, J. R. Villar, and V. M. González, “A low-power HAR method for fall and high-intensity ADLs identification using wrist-worn accelerometer devices,” *Log. J. IGPL*, vol. 31, no. 2, pp. 375–389,

2023.
[8] P. Bhattacharjee, S. Biswas, S. Chattopadhyay, S. Roy, and S. Chakraborty, "Smart Assistance to Reduce the Fear of Falling in Parkinson Patients Using IoT," *Wirel. Pers. Commun.*, vol. 130, no. 1, pp. 281–302, 2023.
[9] H.-C. Lin, M.-J. Chen, C.-H. Lee, L.-C. Kung, and J.-T. Huang, "Fall Recognition Based on an IMU Wearable Device and Fall Verification through a Smart Speaker and the IoT," *Sensors*, vol. 23, no. 12, 2023.
[10] A. Alqahtani, S. Alsubai, M. Sha, V. Peter, A. S. Almadhor, and S. Abbas, "Falling and Drowning Detection Framework Using Smartphone Sensors," *Comput. Intell. Neurosci.*, vol. 2022, 2022.
[11] B. Brew, S. G. Faux, and E. Blanchard, "Effectiveness of a Smartwatch App in Detecting Induced Falls: Observational Study," *JMIR Form. Res.*, vol. 6, no. 3, 2022.
[12] M. E. Issa, A. M. Helm, M. A. A. Al-Qaness, A. Dahou, M. A. Elaziz, and R. Damaševičius, "Human Activity Recognition Based on Embedded Sensor Data Fusion for the Internet of Healthcare Things," *Healthc.*, vol. 10, no. 6, 2022.
[13] A. N. Pereira *et al.*, "Flexible Sensor Suite Integrated into Textile for Calcium Ion and Fall Detection," *IEEE Sensors Lett.*, vol. 6, no. 10, 2022.
[14] Y. H. Nho, J. G. Lim, and D. S. Kwon, "Cluster-Analysis-Based User-Adaptive Fall Detection Using Fusion of Heart Rate Sensor and Accelerometer in a Wearable Device," *IEEE Access*, vol. 8, pp. 40389–40401, 2020.
[15] M. Saleh and R. L. B. Jeannes, "Elderly Fall Detection Using Wearable Sensors: A Low Cost Highly Accurate Algorithm," *IEEE Sens. J.*, vol. 19, no. 8, pp. 3156–3164, 2019.
[16] C. Wang *et al.*, "Low-Power Fall Detector Using Triaxial Accelerometry and Barometric Pressure Sensing," *IEEE Trans. Ind. Informatics*, vol. 12, no. 6, pp. 2302–2311, 2016.
[17] X. Wang, J. Ellul, and G. Azzopardi, "Elderly Fall Detection Systems: A Literature Survey," *Front. Robot. AI*, vol. 7, no. June, 2020.
[18] J. L. Chua, Y. C. Chang, and W. K. Lim, "A simple vision-based fall detection technique for indoor video surveillance," *Signal, Image Video Process.*, vol. 9, no. 3, pp. 623–633, 2015.
[19] M. G. Amin, Y. D. Zhang, F. Ahmad, and K. C. D. Ho, "Radar signal processing for elderly fall detection: The future for in-home monitoring," *IEEE Signal Process. Mag.*, vol. 33, no. 2, pp. 71–80, 2016.
[20] B. Kwolek and M. Kepski, "Human fall detection on embedded platform using depth maps and wireless accelerometer," *Comput. Methods Programs Biomed.*, vol. 117, no. 3, pp. 489–501, 2014.
[21] Y. Wang, K. Wu, and L. M. Ni, "WiFall: Device-Free Fall Detection by Wireless Networks," *IEEE Trans. Mob. Comput.*, vol. 16, no. 2, pp. 581–594, 2017.
[22] S. K. Gharghan and H. A. Hashim, "A comprehensive review of elderly fall detection using wireless communication and artificial intelligence techniques," *Measurement*, vol. 226, p. 114186, Feb. 2024.
[23] P. Kumar, S. Chauhan, and L. K. Awasthi, "Human Activity Recognition (HAR) Using Deep Learning: Review, Methodologies, Progress and Future Research Directions," *Arch. Comput. Methods Eng.*, Aug. 2023.
[24] L. Minh Dang, K. Min, H. Wang, M. Jalil Piran, C. Hee Lee, and H. Moon, "Sensor-based and vision-based human activity recognition: A comprehensive survey," *Pattern Recognit.*, vol. 108, p. 107561, Dec. 2020.
[25] L. Palmerini, J. Klenk, C. Becker, and L. Chiari, "Accelerometer-based fall detection using machine learning: Training and testing on real-world falls," *Sensors (Switzerland)*, vol. 20, no. 22, pp. 1–15, 2020.
[26] N. Pannurat, S. Thiemjarus, and E. Nantajeewarawat, "Automatic fall monitoring: A review," *Sensors (Switzerland)*, vol. 14, no. 7, pp. 12900–12936, 2014.
[27] S. Nooruddin, M. M. Islam, F. A. Sharna, H. Alhetari, and M. N. Kabir, "Sensor-based fall detection systems: A review," *J. Ambient Intell. Humaniz. Comput.*, vol. 13, no. 5, pp. 2735–2751, 2022.
[28] J. A. Santoyo-Ramón, E. Casilari, and J. M. Cano-García, "A study of one-class classification algorithms for wearable fall sensors," *Biosensors*, vol. 11, no. 8, pp. 1–20, 2021.
[29] H. Kimura, M. Nakamura, N. Inou, M. Matsudaira, and M. Yoshida, "Identification Method of Sensor Directions and Sensitivities in Multi-Axis Accelerometer (Actual Measurement of Direction Tensor and Sensitivity Tensor)," *J. Robot. Mechatronics*, vol. 25, no. 2, pp. 408–416, Apr. 2013.
[30] J. R. Villar, C. Chira, E. de la Cal, V. M. González, J. Sedano, and S. B. Khojasteh, "Autonomous on-wrist acceleration-based fall detection systems: Unsolved challenges," *Neurocomputing*, vol. 452, pp. 404–413, 2021.
[31] M. Abbas and R. L. B. Jeannes, "Exploiting Local Temporal Characteristics via Multinomial Decomposition Algorithm for Real-Time Activity Recognition," *IEEE Trans. Instrum. Meas.*, vol. 70, 2021.
[32] G. Šeketa, L. Pavlaković, D. Džaja, I. Lacković, and R. Magjarević, "Event-Centered Data Segmentation in Accelerometer-Based Fall Detection Algorithms," *Sensors*, vol. 21, no. 13, p. 4335, Jun. 2021.
[33] C. Mosquera-Lopez *et al.*, "Automated Detection of Real-World Falls: Modeled from People with Multiple Sclerosis," *IEEE J. Biomed. Heal. Informatics*, vol. 25, no. 6, pp. 1975–1984, 2021.
[34] M. J. Al Nahian *et al.*, "Towards an Accelerometer-Based Elderly Fall Detection System Using Cross-Disciplinary Time Series Features," *IEEE Access*, vol. 9, pp. 39413–39431, 2021.
[35] M. Saleh, M. Abbas, and R. B. Le Jeannes, "FallAllD: An Open Dataset of Human Falls and Activities of Daily Living for Classical and Deep Learning Applications," *IEEE Sens. J.*, vol. 21, no. 2, pp. 1849–1858, 2021.
[36] Y. Yan and Y. Ou, "Accurate fall detection by nine-axis IMU sensor," *2017 IEEE Int. Conf. Robot. Biomimetics, ROBIO 2017*, vol. 2018-Janua, pp. 1–6, 2018.

[37] M. M. Musngi, O. Aziz, S. Zihajehzadeh, G. C. Nazareth, C. G. Tae, and E. J. Park, "Use of Average Vertical Velocity and Difference in Altitude for Improving Automatic Fall Detection from Trunk Based Inertial and Barometric Pressure Measurements," *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. EMBS*, vol. 2018-July, pp. 5146–5149, 2018.
[38] J. K. Lee, S. N. Robinovitch, and E. J. Park, "Inertial Sensing-Based Pre-Impact Detection of Falls Involving Near-Fall Scenarios," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 23, no. 2, pp. 258–266, 2015.
[39] A. Ferreira, G. Santos, A. Rocha, and S. Goldenstein, "User-Centric Coordinates for Applications Leveraging 3-Axis Accelerometer Data," *IEEE Sens. J.*, vol. 17, no. 16, pp. 5231–5243, 2017.
[40] X. Yu, J. Jang, and S. Xiong, "A Large-Scale Open Motion Dataset (KFall) and Benchmark Algorithms for Detecting Pre-impact Fall of the Elderly Using Wearable Inertial Sensors," *Front. Aging Neurosci.*, vol. 13, Jul. 2021.

Author(s) Contributions
The article is written by a single author. I hereby declare that I have prepared the article alone for the authorship declaration.

Conflict of Interest Notice
The author declares that there are no potential conflicts of interest.

Ethical Approval and Informed Consent
It is declared that during the preparation process of this study, scientific and ethical principles were followed, and all the studies benefited from are stated in the bibliography.

Availability of data and material
The datasets analyzed during the current study are available from https://sites.google.com/view/kfalldataset/

Plagiarism Statement
This article has been scanned by iThenticate ™.

SAKARYA UNIVERSITY JOURNAL OF COMPUTER AND INFORMATION SCIENCES
http://saucis.sakarya.edu.tr/
e-ISSN: 2636-8129
Publisher: Sakarya University
Vol. 7, No. 3, 2024, 449-459
DOI: 10.35377/saucis...1543993
SAKARYA UNIVERSITY
RESEARCH ARTICLE

An Evaluation of Skin Lesion Segmentation Using Deep Learning Architectures

Gökçen Çetinel[1], Bekir Murat Aydın[1], Sevda Gül[2*], Devrim Akgün[3], Rabia Öztaş Kara[4]

[1]Department of Electrical and Electronics Engineering, Faculty of Engineering, Sakarya University, Sakarya, Türkiye
[2]Department of Electronics and Automation, Adapazarı Vocational School, Sakarya University, Sakarya, Türkiye
[3]Department of Software Engineering, Faculty of Computer and Information Sciences, Sakarya University, Sakarya, Türkiye
[4]Department of Dermatology, Training and Research Hospital, Sakarya University, Sakarya, Türkiye

Corresponding author:
Sevda Gül,
Department of Electronics and Automation,
Adapazarı Vocational School,
Sakarya University, Sakarya, Türkiye,
gulsevda@sakarya.edu.tr

Article History:
Received: 05.09.2024
Accepted: 20.11.2024
Published Online: 25.12.2024

ABSTRACT
Skin lesion segmentation for recognizing and defining the boundaries of skin lesions in images is proper for automated analysis of skin lesion images, especially for the early diagnosis and detection of skin cancers. Deep learning architectures are an efficient way to implement segmentation once a skin lesion dataset is provided with ground truth images. This study evaluates deep learning architectures on a hybrid dataset, including a private dataset collected from a hospital and a public ISIC dataset. Four different test cases exist in the analysis where the combinations of public and private datasets are used as train and test datasets. Experimental results include Unet, Unet++, DeepLabV3, DeepLabV3++, and FPN segmentation architectures. According to the comparative evaluations, mixed datasets, where public and private datasets were used together, provided the best results. The evaluations also show that the collected dataset with ground truth images provided promising results.

Keywords: Skin lesion segmentation, Deep learning architectures, Unet, DeepLabV3, Feature pyramids network

1. Introduction

Skin cancer is a significant global health concern, and effective treatment depends on early and accurate identification [1]. Skin lesion segmentation is helpful for automated analysis of skin lesion images, especially for the early diagnosis and monitoring of skin cancers. Segmentation of the skin lesions process helps recognize and draw the boundaries of skin lesions in images to increase the efficiency and accuracy of skin cancer diagnosis. Accurate segmentation is essential for quantitative assessment of lesion characteristics and monitoring changes in lesions over time. This can be challenging due to variations in lesion size, shape, color, texture, and factors like lighting conditions and image quality. Skin cancer, particularly melanoma, poses a growing global health threat, with millions of new cases diagnosed each year. Early skin cancer detection, primarily through automated skin lesion segmentation, is critical in improving survival rates. Lesion segmentation enables clinicians to delineate cancerous tissues precisely, providing vital information for diagnosis and treatment planning.

Skin lesion segmentation aims to outline the lesions' boundaries accurately, separating them from the surrounding healthy skin. Many approaches and designs have been developed, each with unique strengths and uses. Segmentation techniques can be manual, where trained professionals outline the lesion boundaries manually, or automated, where computer algorithms are employed to perform the segmentation automatically. Once adequately trained, machine learning models can reach high levels of accuracy in segmentation [2]. Advances in skin lesion segmentation with deep learning approaches, a popular machine learning subclass, have significantly contributed to diagnosing skin diseases [3]. Deep learning models have improved the field of skin lesion segmentation by giving powerful tools for contrasting skin lesions from healthy skin in dermatological images [4].

This study aims to evaluate the effectiveness of deep learning architectures in diagnosing skin lesions by analyzing their performance across a combination of public International Skin Imaging Collaboration (ISIC) datasets [5]–[8] and private datasets confidential to Sakarya University Training and Research Hospital (SEAH). Experimental evaluations involve four distinct test cases where the training and testing datasets consist of various mixes of these public and private datasets, aiming to assess how well these models can learn from and adapt to the differences in data sources, improving the accuracy and reliability of skin lesion diagnosis. Experimental results include a comparative evaluation of Unet, Unet++, DeepLabV3,

Cite as: G. Çetinel, B. M. Aydın, S Gül, D. Akgün, R. Öztaş Kara, "An Evaluation of Skin Lesion Segmentation Using Deep Learning Architectures," *Sakarya University Journal of Computer and Information Sciences*, vol.7, no. 3, pp. 449-459, 2024. Doi: 10.35377/saucis…1543993

DeepLabV3++, and Feature Pyramid Network (FPN) segmentation architectures.

2. Related Works

The progress made in the development of deep learning models has helped improve skin lesion segmentation in recent years. Researchers focused on designing better architectures to improve the accuracy of skin lesion segmentation. Ronneberger et al. introduced the Unet architecture based on Convolutional Neural Network (CNN) with a symmetric encoder-decoder structure with skip connections [9]. They developed the Unet architecture primarily for medical segmentation tasks, but it has also been found to be helpful in other disciplines [10]–[13]. Its structure is characterized by a reduction path to capture essential features and an expanding path that enables accurate localization of the segmentation region. Zhou et al. proposed Unet++, a modified version of Unet that introduces nested, dense skip pathways for improved segmentation accuracy [14]. DeepLabV3 and DeepLabV3+, developed by Chen et al., utilize atrous convolution and Atrous Spatial Pyramid Pooling (ASPP) to effectively segment objects at various scales, proving highly beneficial for skin lesion analysis [15]. Oktay et al. added attention blocks to the Unet architecture to focus on relevant features for segmentation, improving the precision of lesion identification [16]. Yi et al. structured the problem as an adversarial application using Generative Adversarial Networks (GANs) for skin lesion segmentation, which resulted in more refined segmentation outputs [17]. SegNet, proposed by Badrinarayanan et al., employs an encoder-decoder architecture for pixel-wise classification tasks. They have adapted it for skin lesion segmentation to address variations in lesion appearance effectively [18]. Fully Convolutional Networks (FCNs), introduced by Long et al., facilitate semantic segmentation by allowing end-to-end segmentation with input of any size [19]. Alom et al. developed R2Unet, which integrates Recurrent Neural Networks (RNNs) with Unet to capture image spatial dependencies better [20]. Milletari et al. proposed V-Net, a volumetric extension of Unet designed for 3D image segmentation, adapted by researchers for 3D skin lesion segmentation tasks [21]. Ibtehaz and Rahman introduced MultiResUNet, which improves Unet by combining multi-resolution analysis to better capture features at various scales [22]. Sharen et al. combined Unet and FPN for skin lesion segmentation with encoder modifications [23]. Thanks for the contributions of the reviewer.

Furthermore, recent studies have demonstrated the efficacy of Unet++ and DeepLabV3+ in various medical imaging applications. For instance, Unet++ has shown effective boundary detection for complex medical imaging tasks like breast cancer and lung nodule segmentation. It enhances segmentation precision through dense skip connections, refining feature extraction in these applications [24]. Similarly, DeepLabV3+ has outperformed traditional models in retinal image analysis and liver lesion segmentation by effectively utilizing multi-scale feature extraction [25]. Recent advancements in skin lesion segmentation have included attention mechanisms, which focus on the most relevant image regions, improving segmentation accuracy [26]. GANs have also been applied to generate high-quality segmentations by modeling lesion boundaries more precisely [27].

Additionally, transformer-based models have emerged as promising due to their ability to capture long-range dependencies [28]. Many works in the literature focus on segmentation architectures for improving skin lesion segmentation by using public datasets [29], [30]. This study used well-known Unet, Unet++, DeepLabV3, DeepLabV3++, and FPN segmentation architectures. We focused on the performances of the popular segmentation architectures over public, private, and mixed datasets.

3. Methodology

This section introduces the segmentation models investigated in the proposed study and describes the used dataset.

3.1. Dataset

The study database was obtained by combining public and private databases. A total of 4228 images were utilized in the study, comprising 3463 images acquired from public databases and 765 images sourced from private database. The comprehensive database encompasses diverse dermoscopy images corresponding to various lesion types.

The ISIC datasets are an essential public resource in dermatology for skin lesion segmentation and classification research. They comprise a vast collection of publicly accessible dermoscopic images with detailed annotations, covering various skin lesions, including melanoma. These datasets, enriched with metadata like patient demographics and clinical diagnoses, are vital for developing and validating machine-learning algorithms for automated skin lesion analysis. The ISIC datasets support challenges and competitions, such as the ISIC Challenge, promoting advancements in diagnostic accuracy and dermatological research. ISIC is an open-access database that was extended in 2016, 2017, 2018, 2019, and 2020.

This study's curated ISIC database includes 4075 images with ground truth annotations. However, there exist recurring images. So, we considered the database before performing the segmentation algorithm, and 3463 images remained. The database encompasses various diagnostic categories commonly encountered in medical and dermatological image analysis. These categories include actinic keratoses and intraepithelial carcinoma, basal cell carcinoma, benign keratosis-like lesions, dermatofibroma, melanoma, melanocytic nevi, and vascular lesions. Each of these categories represents a distinct skin condition or pathology, and including such a variety of conditions makes the database valuable for research and training machine learning models in the context of dermatological diagnosis and skin cancer detection.

The private database for the study, which consists of 765 images, was constructed by a dermatologist from Sakarya University Training and Research Hospital with the permission of the ethics committee. The collected private dataset consists of high-resolution dermoscopic images with a resolution of 1024x1024 pixels. The dermatologist manually determined the lesion area for the images in the private database, thus creating ground truth masks. The included skin lesion types were melanoma, melanocytic nevus, lentigo, seborrheic keratosis, angioma, actinic keratosis, Bowen's disease, trichilemmal cyst, keratoacanthoma, hemangioma, malignant melanoma, basal cell carcinoma, pyogenic granuloma, Spitz nevus, and Merkel cell carcinoma. However, the dataset has limitations, including a smaller sample size than public datasets like ISIC and potential biases introduced during the manual annotation. The training, validation, and test sets were assembled by integrating images from public and private databases. These databases were systematically divided into proportions of 60%, 20%, and 20%, respectively.

Furthermore, the database intentionally includes challenging lesion images to ensure a robust evaluation of the proposed segmentation process. These include lesions found on individuals with dark skin tones, lesions covering the entire imaging field, and lesions closely resembling the skin color. Figure 1 displays examples of lesion images and their corresponding ground truth masks from our private dataset, which comprises 765 images, alongside the ISIC dataset containing 3463 images.

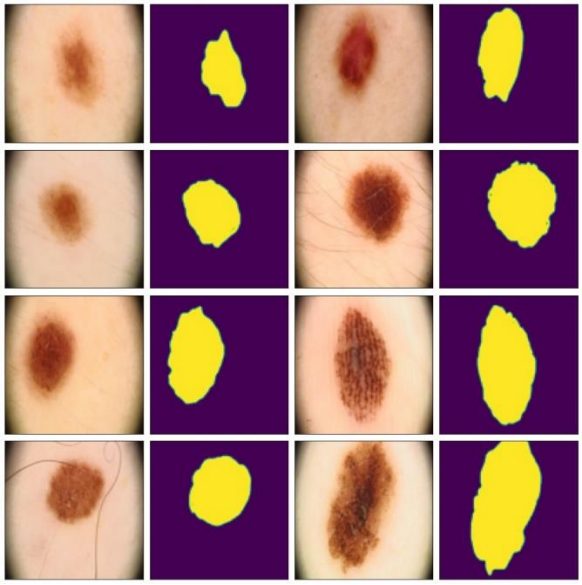


Figure 1. Example Images and Their Masks from the Private Dataset

3.2. Segmentation Models

Segmentation models mainly consist of encoder and decoder structures. The encoder filters out important features by following a narrowing path of convolution and scaling layers. The decoder, again formed by the convolution and scaling layers, obtains the output image of the same size from the features obtained. One of the architectures we used for segmentation tasks is the Unet model, which is one of the pioneer implementations with convolutional layers [9]. The Unet and its derivatives are fundamental in medical image segmentation, particularly for tasks like skin lesion segmentation. Unet's architecture and later modifications, designed for biomedical image segmentation, have proven helpful in distinguishing skin lesions from surrounding healthy skin. A variation of Unet is Unet++, which uses a series of nested, dense skip pathways that aim to reduce the semantic gap between the feature maps of the encoder and decoder parts of the network [14]. This design enhances the model's ability to capture fine-grained details and improves segmentation performance, particularly in challenging areas of skin lesion images. In Unet++, nested skip pathways introduce dense skip connections between the encoder and decoder parts of the network. These connections bridge the semantic gap between lower-level and higher-level feature maps, enhancing the model's ability to capture fine details. The architecture integrates multiple dense blocks, each designed to improve the segmentation performance by refining the feature maps at different stages. Figure 2 shows the basic structure of Unet++, where skip pathways between the encoder and decoder blocks are illustrated. In Figure 2, the Unet++ architecture shows how nested skip connections enable fine-grained segmentation by continuously refining the feature maps at different resolutions. Another model we used for comparison is the DeepLabv3 architecture, which includes decoder encoder structures like Unit. However, DeepLabV3 uses dilated convolutions and ASPP to identify the features of the backbone network. The ASPP module in DeepLabV3+ captures context at multiple scales by applying dilated convolutions with varying rates. This allows the model to handle objects of varying sizes effectively, enhancing segmentation accuracy in complex images without clearly defined object boundaries. DeepLabV3+ uses the DeepLabV3 encoder module and atrous convolution to adjust the resolution of features [15] as shown in Figure 3. DeepLabV3+'s ASPP module, which extracts contextual information from multi-scale receptive fields, improves the model's ability to handle lesions of varying sizes and is demonstrated in Figure 3. The last model we used for comparison is the FPN architecture, which is helpful for scale

variations in object detection tasks [31]. The multi-scale feature pyramid enables FPNs to collect features of objects regardless of their size in the image. Figure 4 shows the basic structure of FPN, where a multi-scale feature pyramid is formed by integrating feature maps from several stages of a backbone model like ResNet. The FPN's multi-scale feature pyramid, which allows the model to detect objects of different scales by combining low- and high-level feature maps effectively, is also highlighted in the figure.

These architectures were selected due to their proven effectiveness in medical image segmentation tasks. Unet is well-suited for biomedical segmentation because it captures fine details through skip connections between encoder and decoder layers. However, it struggles with scale variations in lesions. Unet++ improves upon Unet by introducing nested skip connections, reducing the semantic gap between feature maps at different stages. This results in improved segmentation accuracy, particularly for complex lesion boundaries. DeepLabV3 and DeepLabV3+ utilize ASPP to handle objects at multiple scales, making them highly effective for varying-sized lesions. However, their computational complexity can be a drawback in resource-constrained environments. FPN, with its multi-scale feature pyramid, excels in detecting lesions of different sizes but may lose fine details in small objects. FPN's ability to handle scale variations further supports robust segmentation, making these models ideal.

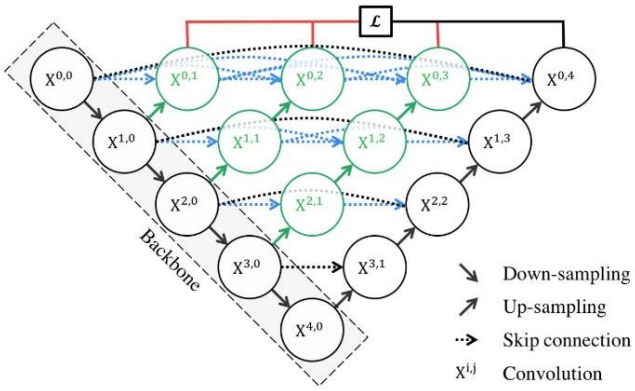


Figure 2. Unet++ Architecture [14]

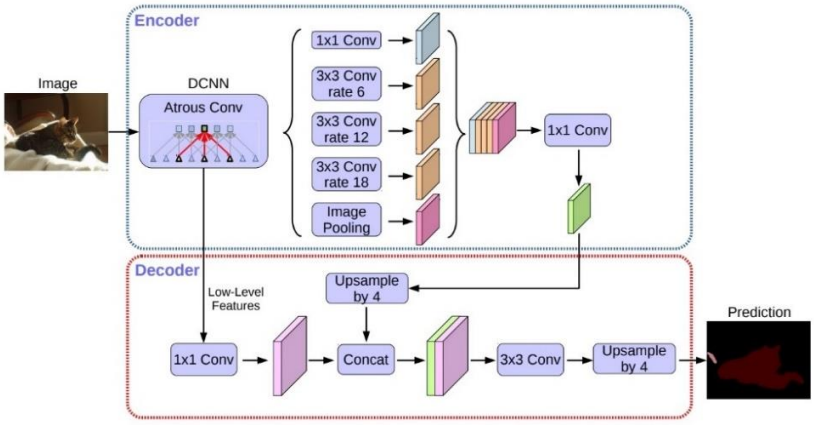


Figure 3. DeepLabV3+ Architecture [15]

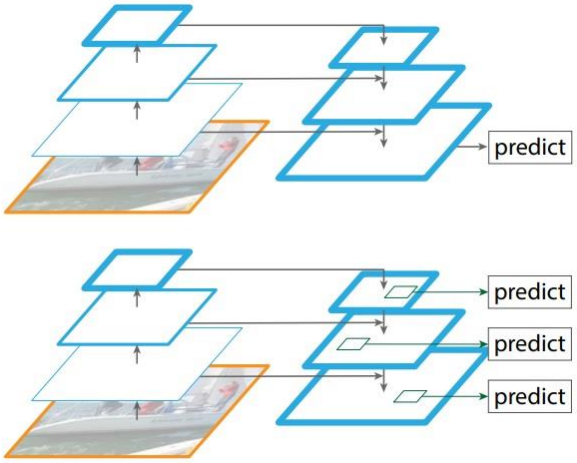


Figure 4. FPN Architecture [26]

4. Experimental Results

Experimental results were obtained using the programming language Python 3.11 and the deep learning library PyTorch 1.12.1 on the Ubuntu 20.04 operating system. The hardware for the experimental results utilizes an NVIDIA GeForce RTX 3060 12GB GPU alongside an AMD Ryzen 2700 CPU with 32GB of memory. The models were trained using the Adam optimizer with an initial learning rate 0.001. A batch size 16 was selected to balance memory usage and training efficiency. Training was conducted over 100 epochs, with early stopping employed to prevent overfitting. Data augmentation techniques such as horizontal and vertical flipping, random rotation, and brightness adjustment were applied to increase the variability of the training set and improve generalization. The training process took approximately 12 hours on an NVIDIA GeForce RTX 3060 GPU.

Four different experimental scenarios were created for training and testing segmentation models in experimental studies. The first and fourth cases involve training on the public dataset, with subsequent testing conducted on private and public datasets. Case 3 trains and tests the model on a private dataset. Conversely, Case 2 performs training and test tasks on the hybrid dataset. This approach enables a comprehensive evaluation of segmentation performance, capturing a spectrum of conditions encountered across public and private image datasets. Conducting a study focused on skin lesion segmentation, a private dataset is being curated alongside publicly available ones for training. Utilizing both public and private datasets offers several advantages. Public data aids in generalizing the model, enabling it to comprehend diverse scenarios effectively.

Conversely, the private dataset focuses on specific application domains, enhancing the model's performance within those contexts. Employing a mixed training approach that integrates both datasets improves the model's adaptability to general and domain-specific data. This approach promotes diversity in training and bolsters the reliability of the model's predictions. Figure 5 presents the distribution of public and private datasets used across the four test cases. The results indicate combining public and private datasets yields the highest performance across all models.

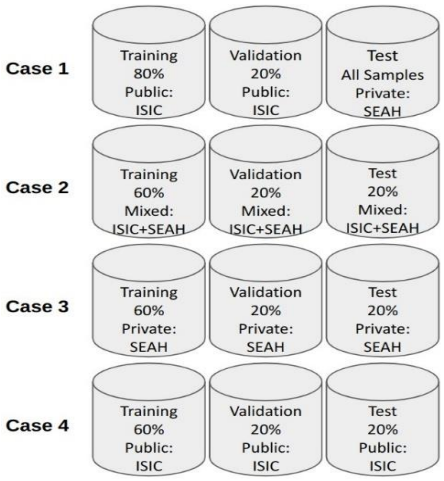


Figure 5. Dataset Distribution for Four Experimental Scenarios

Figure 6 displays the segmentation results for Unet++, DeepLabV3+, and FPN. Notably, FPN achieves superior segmentation quality, particularly in challenging cases where lesions closely resemble surrounding tissue. Original lesion images, manually prepared ground truth images, and the predicted images achieved by the utilized segmentation methods are illustrated in each

figure column, respectively. As can be seen from the sample segmentation results, the ground truth images are compatible with the data. The segmentation models produced similar results, and slightly better results were obtained with the FPN architecture. The detailed comparison shows FPN's robustness in handling such variations.

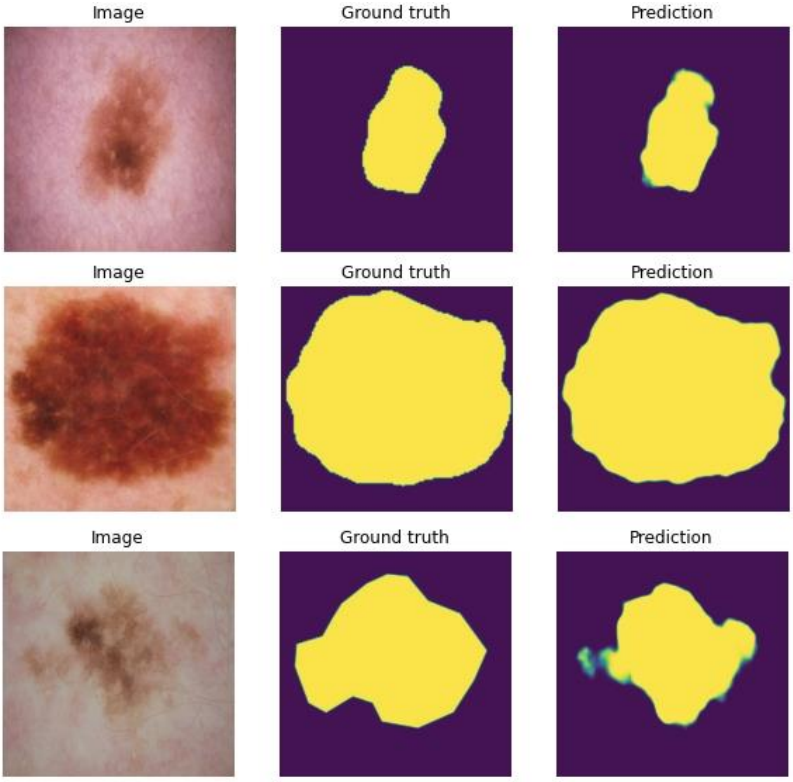


Figure 6. Sample Segmentation Results Showing Original Images, Ground Truth Masks, and Predicted Masks

Various metrics are calculated by comparing the predicted and ground truth masks to evaluate the performance segmentation methods. This study uses two standard metrics to show the models' effectiveness. The Dice coefficient and Intersection of Unıon (IoU) are two prevalent similarity measures. Dice is calculated as twice the overlap area between the predicted and ground truth masks, divided by the total area of both masks, yielding a value between 0 and 1, with 1 indicating perfect overlap. IoU or Jaccard index, on the other hand, is the ratio of the intersection area to the union area of the predicted and ground truth masks. These metrics are crucial for assessing the model's performance, especially in datasets with limited examples. The equations for Dice and IoU metrics can be given as follows:

$$Dice = \frac{2|A \cap B|}{|A|+|B|} \quad (1)$$

$$IoU = \frac{|A \cap B|}{|A \cup B|} \quad (2)$$

where $|A|$ and $|B|$ are the number of pixels in the predicted and ground truth segmentation sets, respectively. $|A \cap B|$ and $|A \cup B|$ show the intersection and union of these sets, respectively.

On the other side, the pixel-level confusion matrix, which includes True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN) values, provides a detailed breakdown of the models' predictions. Accuracy, calculated as the ratio of correctly predicted pixels to the total number of pixels, offers an overall measure of the model's correctness. Sensitivity measures the model's ability to correctly capture all lesion pixels, indicating how effectively the model segments true lesion areas without missing parts. Specificity indicates the model's capacity to exclude non-lesion pixels, reducing false positive segmentation around lesions. High specificity shows reliable background separation. Together, these metrics comprehensively evaluate the model's strengths and weaknesses, enabling precise adjustments to enhance its performance. The equations of the metrics can be expressed as follows:

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP+FN}} \quad (3)$$

$$\text{Specificity} = \frac{\text{TN}}{\text{TN+FP}} \quad (4)$$

$$\text{Accuracy} = \frac{\text{TP+FP}}{\text{TP+TN+FP+FN}} \quad (5)$$

In the presented study, five metrics, Dice, IoU, sensitivity, specificity, and accuracy, were calculated to provide a more comprehensive assessment. Tables 1-5 give the results of the utilized segmentation methods.

Table 1. Methods Evaluations for IoU Metric

Methods	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Average
Unet	0.7623	0.8105	**0.8075**	0.8177	**0.7995**
DeepLabV3	**0.7624**	0.8126	0.7967	0.8140	0.7964
Unet++	0.7507	0.8099	0.8071	**0.8229**	0.7976
DeepLabV3+	0.7549	0.8177	0.7986	0.8219	0.7983
FPN	0.7508	**0.8146**	0.8028	0.8139	0.7955

Table 2. Methods Evaluations for Dice Metric

Methods	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Average
Unet	0.8610	0.8765	0.8837	0.8842	**0.8763**
DeepLabV3	**0.8636**	0.8734	0.8871	0.8767	0.8753
Unet++	0.8519	0.8731	0.8753	**0.8859**	0.8716
DeepLabV3+	0.8587	0.8778	0.8747	0.8851	0.8741
FPN	0.8536	**0.8939**	**0.8874**	0.8797	0.8741

Table 3. Methods Evaluations for Sensitivity Metric

Methods	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Average
Unet	0.9272	0.9346	0.9449	0.9249	0.9329
DeepLabV3	0.9175	0.9376	0.9319	0.9309	0.9295
Unet++	0.9279	0.9348	**0.9585**	0.9191	**0.9351**
DeepLabV3+	**0.9283**	0.9348	0.9572	0.9183	0.9347
FPN	0.9220	**0.9399**	0.9383	**0.9319**	0.9330

Table 4. Methods Evaluations for Specificity Metric

Methods	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Average
Unet	0.9632	0.9566	**0.9641**	0.9538	**0.9594**
DeepLabV3	0.9632	0.9511	0.9615	0.9474	0.9558
Unet++	0.9599	0.9579	0.9571	0.9548	0.9574
DeepLabV3+	0.9607	**0.9595**	0.9557	**0.9550**	0.9577
FPN	**0.9633**	0.9519	0.9637	0.9481	0.9567

Table 5. Methods Evaluations for Accuracy Metric

Methods	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Average
Unet	**0.9457**	0.9530	**0.9573**	0.9517	**0.9519**
DeepLabV3	**0.9457**	0.9531	0.9547	0.9501	0.9509
Unet++	0.9423	0.9527	0.9562	**0.9536**	0.9512
DeepLabV3+	0.9436	**0.9581**	0.9542	0.9533	0.9515
FPN	0.9430	0.9536	0.9564	0.9503	0.9508

As can be seen from the tables, the segmentation performance of the utilized deep learning architectures was evaluated across four distinct scenarios, each presenting unique training and testing conditions. The effectiveness of each model was assessed based on five primary metrics: IoU, Dice coefficient, sensitivity, specificity, and accuracy. The results demonstrate that model performance varies significantly across different dataset compositions, indicating the impact of data diversity and source-specific characteristics on segmentation quality.

In Scenario 1, where models were trained on the public dataset and tested on the private dataset, the DeepLabV3 model achieved the highest IoU score of 0.7524. This score suggests that DeepLabV3 can capture lesion boundaries effectively, even when applied to data that differ in distribution from the training set. In Scenario 2, which involved training and testing on a hybrid dataset combining public and private data, FPN outperformed other models, achieving an IoU score of 0.8146. This result highlights the benefit of mixed data training, as FPN could adapt to both public and private image features, improving generalization. Scenario 3, trained and tested solely on private data, Unet yield the best IoU performance, with a score of 0.8075. In Scenario 4, training and testing were conducted on the public dataset. Unet++ achieved the top IoU of 0.8224, likely due to the model's strong alignment with the diversity found within public datasets.

For Dice coefficients, the DeepLabV3 model attained the highest result of 0.8636 in Scenario 1, indicating superior overlap between predicted and ground truth lesion areas on private data. This shows DeepLabV3's potential to generalize well, even when the test set differs from the training set. In Scenario 2, FPN exhibited the highest Dice coefficient of 0.8939, benefiting from the hybrid dataset's balanced diversity and enhancing the segmentation accuracy. In Scenario 3, with only private data, FPN again performed best, achieving a Dice score of 0.8874, affirming its effectiveness in segmenting private data lesions. In Scenario 4, where the public dataset was used exclusively, Unet++ achieved a Dice score of 0.8859, suggesting a robust performance on homogeneous data sources.

Across all scenarios, Unet++ showed the highest sensitivity in Scenario 3 (0.9581), indicating its strong ability to identify true positive lesion pixels on private data. This suggests that Unet++ is particularly adept at capturing lesion regions within private data, which often include complex lesion shapes. Conversely, in Scenario 2, DeepLabV3 and FPN also demonstrated high sensitivity values, suggesting that these models can successfully detect true lesion areas across varying data sources.

The Unet model recorded the highest specificity in Scenario 3, reaching 0.9641. This metric reflects Unet's ability to minimize false positives in segmenting lesions from private dataset backgrounds, thus distinguishing lesion boundaries with greater precision. In Scenario 1, DeepLabV3 and FPN models also achieved commendable specificity scores, highlighting their capacity to differentiate lesion regions accurately, even with only public dataset training.

Regarding accuracy, DeepLabV3+ provided the most consistent performance across all scenarios, with its highest accuracy recorded in Scenario 3 (0.9564). This indicates that DeepLabV3+ effectively balances lesion detection and segmentation accuracy, especially in private datasets. The hybrid dataset in Scenario 2 also led to solid accuracy scores across models, reflecting the advantages of mixed data training for segmentation tasks.

Overall, segmentation models exhibited the robust performance across multiple metrics, indicating their potential for clinical applications requiring reliable skin lesion segmentation. These findings emphasize the importance of diverse training datasets and suggest that models trained on hybrid data are more adaptable to real-world dermatological applications where public and private data characteristics may be present.

One of the key advantages of the proposed study is the inclusion of a unique database comprising images obtained from patients at the Dermatology Clinic of Sakarya University Education and Research Hospital under ethically approved protocols. This database combines open-source and private data, enabling a robust analysis across diverse data sources. Our study addresses these limitations by incorporating diverse lesion types and varied imaging conditions from private clinical data, enabling a more robust and comprehensive model evaluation. Working with a hybrid database enhances the generalizability of segmentation models and strengthens their performance in real-time applications. Additionally, the study offers a comparative analysis of segmentation performance across widely recognized architectures, such as Unet, DeepLabV3, Unet++, DeepLabV3+, and FPN. This approach allows for identifying the highest-performing architecture, providing a benchmark for evaluating segmentation accuracy compared to existing studies.

Finally, Table 6 compares the results obtained with those of existing studies. The study in [32] proposed a melanoma segmentation approach, including Unet and LinkNet deep learning networks, coupled with transfer learning and fine-tuning techniques. Experiments conducted on three publicly available databases (PH2, ISIC 2018, and DermIS) have shown promising results, with Unet demonstrating notable performance. Specifically, the average Dice coefficient achieved was 0.923 on the PH2 database, 0.893 on ISIC 2018, and 0.879 on the DermIS database. These findings indicate significant success for U-net across the evaluated databases. The study in [11] introduced a hyper-parameter optimized Fully Convolutional Encoder-Decoder Network (FCEDN) for dermoscopy image segmentation. The novel Exponential Neighborhood Grey Wolf Optimization (EN-GWO) algorithm was employed to optimize network hyper-parameters. EN-

GWO incorporates a neighborhood-based searching strategy, combining individual and global haunting strategies of wolves to strike a balance between exploration and exploitation. The study compared EN-GWO with four variants of Grey Wolf Optimization (GWO), Genetic Algorithm (GA), and Particle Swarm Optimization (PSO) for hyperparameter optimization on the ISIC 2016 and 2017 databases. The proposed model achieves a Jaccard index of 96.41% and 86.85%, a Dice coefficient of 98.48% and 87.23%, and an accuracy of 98.32% and 95.25% for the ISIC 2016 and 2017 databases.

Table 6. Comparison of the Proposed System with Recent Studies

Reference No	Data	Model	Dice	IoU
[32]	ISIC 2018 PH2 DermIS	U-net + LinkNet	0,8940	0,8090
[33]	ISIC 2016 ISIC 2017	FCEDN	0,9848 0,8723	0,9641 0,8685
Ours Best	Hybrid	FPN	0,8939	0,8146

The proposed segmentation approach also gives promising results compared to the other studies. Unlike studies that rely solely on public datasets, the study's dataset includes 3463 public images and 765 private clinical images, enhancing the robustness and generalizability of the given model. Moreover, the comprehensive evaluation strategy of the study, involving four distinct training/test combinations, ensures the results' reliability and robustness. This thorough approach contrasts with the more limited evaluation methodologies reported in the earlier studies.

The results obtained in this study outperform those of earlier works, particularly in the hybrid dataset scenario. Studies such as [33] relied on GAN-based architectures but struggled with small-scale lesions. Our use of FPN and DeepLabV3+ provides better handling of multi-scale lesions, as demonstrated by the superior IoU and Dice scores. However, one limitation of our approach is the reliance on a relatively small private dataset, which may limit the model's generalizability. Future work should focus on expanding the private dataset and exploring transformer-based models for improved segmentation accuracy.

5. Conclusions

This study comprehensively evaluated deep learning architectures for skin lesion segmentation using a unique hybrid dataset comprising public ISIC images and private clinical data. By employing multiple segmentation models, including Unet, Unet++, DeepLabV3, DeepLabV3+, and FPN, across different training and testing scenarios, this work highlighted the impact of data diversity on segmentation performance. The findings indicate that integrating public and private datasets during training significantly enhances model generalizability and segmentation accuracy, especially in heterogeneous clinical applications.

The results demonstrate that the FPN and DeepLabV3+ models consistently perform well across critical metrics such as IoU, Dice coefficient, sensitivity, specificity, and accuracy. FPN, in particular, excelled in hybrid dataset scenarios, suggesting that it effectively adapts to varying lesion types and imaging conditions, an essential characteristic for practical applications in dermatology. These models' ability to delineate lesion boundaries accurately, even within challenging image contexts, underscores their potential suitability for deployment in automated skin lesion analysis tools.

Integrating private clinical data strengthens model robustness and addresses limitations inherent in publicly available datasets, which often need more complexity and specificity of clinical images. This study underscores the importance of hybrid datasets for training deep learning models in medical imaging, as they contribute to more reliable and clinically applicable segmentation outcomes.

Future work should focus on expanding the dataset to include a wider variety of lesion types and further investigate transformer-based or hybrid architectures that can enhance long-range dependency modeling in segmentation. Additionally, attention-based mechanisms could be explored to refine boundary detection in challenging cases. The promising results of this study support the advancement of automated diagnostic tools in dermatology, facilitating early detection and improved treatment planning for skin cancer patients.

References

[1] S. Spanos *et al.*, "Measuring the quality of skin cancer management in primary care: A scoping review," *Australas. J. Dermatol.*, vol. 64, no. 2, pp. 177–193, May 2023, doi: 10.1111/AJD.14023.

[2] R. Javed, M. S. M. Rahim, T. Saba, and A. Rehman, "A comparative study of features selection for skin lesion detection from dermoscopic images," *Netw. Model. Anal. Heal. Informatics Bioinforma.*, vol. 9, no. 1, pp. 1–13, Dec. 2020, doi: 10.1007/S13721-019-0209-1/TABLES/5.

[3] M. Zafar, M. I. Sharif, M. I. Sharif, S. Kadry, S. A. C. Bukhari, and H. T. Rauf, "Skin Lesion Analysis and Cancer Detection Based on Machine/Deep Learning Techniques: A Comprehensive Survey," *Life 2023, Vol. 13, Page 146*, vol. 13, no. 1, p. 146, Jan. 2023, doi: 10.3390/LIFE13010146.

[4] Z. Mirikharaji *et al.*, "A survey on deep learning for skin lesion segmentation," *Med. Image Anal.*, vol. 88, p. 102863, Aug. 2023, doi: 10.1016/J.MEDIA.2023.102863.

[5] N. C. F. Codella *et al.*, "Skin Lesion Analysis Toward Melanoma Detection: A Challenge at the 2017 International Symposium on Biomedical Imaging (ISBI), Hosted by the International Skin Imaging Collaboration (ISIC)," *Proc. - Int. Symp. Biomed. Imaging*, vol. 2018-April, pp. 168–172, Oct. 2017, doi: 10.1109/ISBI.2018.8363547.

[6] N. Codella *et al.*, "Skin Lesion Analysis Toward Melanoma Detection 2018: A Challenge Hosted by the International Skin Imaging Collaboration (ISIC)," Feb. 2019, Accessed: Jul. 12, 2024. [Online]. Available: https://arxiv.org/abs/1902.03368v2.

[7] C. Hernández-Pérez *et al.*, "BCN20000: Dermoscopic Lesions in the Wild," *Sci. Data*, vol. 11, no. 1, Aug. 2019, doi: 10.1038/s41597-024-03387-w.

[8] P. Tschandl, C. Rosendahl, and H. Kittler, "The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions," *Sci. Data 2018 51*, vol. 5, no. 1, pp. 1–9, Aug. 2018, doi: 10.1038/sdata.2018.161.

[9] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 9351, pp. 234–241, 2015, doi: 10.1007/978-3-319-24574-4_28.

[10] N. J. Singh and K. Nongmeikapam, "Semantic Segmentation of Satellite Images Using Deep-Unet," *Arab. J. Sci. Eng.*, vol. 48, no. 2, pp. 1193–1205, Feb. 2023, doi: 10.1007/S13369-022-06734-4/TABLES/2.

[11] L. Zhang, J. Shen, and B. Zhu, "A research on an improved Unet-based concrete crack detection algorithm," *Struct. Heal. Monit.*, vol. 20, no. 4, pp. 1864–1879, Jul. 2021, doi: 10.1177/1475921720940068/ASSET/IMAGES/10.1177_1475921720940068-IMG1.PNG.

[12] D. Harrison, F. C. De Leo, W. J. Gallin, F. Mir, S. Marini, and S. P. Leys, "Machine Learning Applications of Convolutional Neural Networks and Unet Architecture to Predict and Classify Demosponge Behavior," *Water 2021, Vol. 13, Page 2512*, vol. 13, no. 18, p. 2512, Sep. 2021, doi: 10.3390/W13182512.

[13] D.-Y. Chen *et al.*, "Building Extraction and Number Statistics in WUI Areas Based on UNet Structure and Ensemble Learning," *Remote Sens. 2021, Vol. 13, Page 1172*, vol. 13, no. 6, p. 1172, Mar. 2021, doi: 10.3390/RS13061172.

[14] Z. Zhou, M. M. Rahman Siddiquee, N. Tajbakhsh, and J. Liang, "Unet++: A nested u-net architecture for medical image segmentation," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 11045 LNCS, pp. 3–11, 2018, doi: 10.1007/978-3-030-00889-5_1/FIGURES/3.

[15] L. C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 11211 LNCS, pp. 833–851, Feb. 2018, doi: 10.1007/978-3-030-01234-2_49.

[16] O. Oktay *et al.*, "Attention U-Net: Learning Where to Look for the Pancreas," Apr. 2018, Accessed: Jul. 12, 2024. [Online]. Available: https://arxiv.org/abs/1804.03999v3.

[17] X. Yi, E. Walia, and P. Babyn, "Generative adversarial network in medical imaging: A review," *Med. Image Anal.*, vol. 58, p. 101552, Dec. 2019, doi: 10.1016/J.MEDIA.2019.101552.

[18] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Nov. 2015, doi: 10.1109/TPAMI.2016.2644615.

[19] E. Shelhamer, J. Long, and T. Darrell, "Fully Convolutional Networks for Semantic Segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640–651, Nov. 2014, doi: 10.1109/TPAMI.2016.2572683.

[20] M. Z. Alom, M. Hasan, C. Yakopcic, T. M. Taha, and V. K. Asari, "Recurrent Residual Convolutional Neural Network based on U-Net (R2U-Net) for Medical Image Segmentation," Feb. 2018, Accessed: Jul. 12, 2024. [Online]. Available: https://arxiv.org/abs/1802.06955v5.

[21] F. Milletari, N. Navab, and S. A. Ahmadi, "V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation," *Proc. - 2016 4th Int. Conf. 3D Vision, 3DV 2016*, pp. 565–571, Jun. 2016, doi: 10.1109/3DV.2016.79.

[22] N. Ibtehaz and M. S. Rahman, "MultiResUNet : Rethinking the U-Net Architecture for Multimodal Biomedical Image Segmentation," *Neural Networks*, vol. 121, pp. 74–87, Feb. 2019, doi: 10.1016/j.neunet.2019.08.025.

[23] H. Sharen, M. Jawahar, L. Jani Anbarasi, V. Ravi, N. Saleh Alghamdi, and W. Suliman, "FDUM-Net: An enhanced FPN and U-Net architecture for skin lesion segmentation," *Biomed. Signal Process. Control*, vol. 91, p. 106037, May 2024, doi: 10.1016/J.BSPC.2024.106037.

[24] Sweta Jain, Pruthviraj Choudhari, Mahesh Gour, Pulmonary Lung Nodule Detection from Computed Tomography Images Using Two-Stage Convolutional Neural Network, *The Computer Journal*, Volume 66, Issue 4, April 2023, Pages 785–795.

[25] He, X., Wang, Y., Poiesi, F., Song, W., Xu, Q., Feng, Z., & Wan, Y. (2023). Exploiting multi-granularity visual features for retinal layer segmentation in human eyes. *Frontiers in Bioengineering and Biotechnology*, *11*, 1191803.

[26] Oktay, O., Schlemper, J., Folgoc, L. L., Lee, M., Heinrich, M., Misawa, K., Mori, K., McDonagh, S., Hammerla, N. Y., Kainz, B., Glocker, B., & Rueckert, D. (2018). Attention U-Net: Learning where to look for the pancreas. *arXiv preprint arXiv:1804.03999*.

[27] Yi, X., Walia, E., & Babyn, P. (2019). Generative adversarial network in medical imaging: A review. *Medical Image Analysis*, *58*, 101552. https://doi.org/10.1016/j.media.2019.101552

[28] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M.,

Heigold, G., Gelly, S., Uszkoreit, J., & Houlsby, N. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

[29] M. K. Hasan, M. A. Ahamad, C. H. Yap, and G. Yang, "A survey, review, and future trends of skin lesion segmentation and classification," *Comput. Biol. Med.*, vol. 155, p. 106624, Mar. 2023, doi: 10.1016/J.COMPBIOMED.2023.106624.

[30] M. Strzelecki, M. Kociołek, M. Strąkowska, M. Kozłowski, A. Grzybowski, and P. M. Szczypiński, "Artificial intelligence in the detection of skin cancer: State of the art," *Clin. Dermatol.*, vol. 42, no. 3, pp. 280–295, May 2024, doi: 10.1016/J.CLINDERMATOL.2023.12.022.

[31] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature Pyramid Networks for Object Detection," Dec. 2016, Accessed: Jul. 12, 2024. [Online]. Available: https://arxiv.org/abs/1612.03144v2.

[32] R. L. Araújo, F. H. D. d. Araújo, and R. R. V. e. Silva, "Automatic segmentation of melanoma skin cancer using transfer learning and fine-tuning," *Multimed. Syst.*, vol. 28, no. 4, pp. 1239–1250, Aug. 2022, doi: 10.1007/S00530-021-00840-3/TABLES/8.

[33] R. Mohakud and R. Dash, "Skin cancer image segmentation utilizing a novel EN-GWO based hyper-parameter optimized FCEDN," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 34, no. 10, pp. 9889–9904, Nov. 2022, doi: 10.1016/J.JKSUCI.2021.12.018.

Article Information Form

Authors Contributions

Dr. Gökçen ÇETİNEL: Conceptualization, study design, critical revision of the manuscript, final editing, supervision, and project administration.
Bekir Murat Aydın: Software development and data collection
Dr. Sevda GÜL: Literature review, software development, results interpretation, and data validation
Dr. Devrim AKGÜN: Methodology development, software development, and study design.
Dr. Rabia ÖZTAŞ KARA: Data collection, data validation and statistical analysis.

Conflict of Interest Notice

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The study is supported by The Scientific and Technical Research Council of Turkey (TUBITAK) through The Research Support Programs Directorate (ARDEB) with project number of 122E629.

Ethical Approval and Informed Consent

The Ethical Committee of Sakarya University Medical Sciences Faculty approved the study. All procedures in studies involving human participants were by the ethical standards of the institutional and/or national search committee.

Availability of Data and Materials

No Applicable

Plagiarism Statement

This article has been scanned by iThenticate™.

SAKARYA UNIVERSITY JOURNAL OF COMPUTER AND INFORMATION SCIENCES
http://saucis.sakarya.edu.tr/
e-ISSN: 2636-8129
Publisher: Sakarya University
Vol. 7, No. 3, 2024, 460-469
DOI: 10.35377/saucis...1546785
SAKARYA UNIVERSITY
RESEARCH ARTICLE

Experimental Investigation of the Effect of Two-Stage Peltier Application on the Temperature of a Microprocessor

Fatih Uysal[1], Sinan Çobaner[1*]

[1] Sakarya University of Applied Sciences, Faculty of Technology, Department of Mechanical Engineering, Sakarya, Türkiye

Corresponding author:
Sinan Çobaner, Sakarya University of Applied Sciences, Faculty of Technology, Department of Mechanical Engineering, Sakarya, Türkiye,
sinancobaner@subu.edu.tr

ABSTRACT
Increasing the number of transistors to enhance the performance of processors leads to overheating, creating a need for cooling. Traditional cooling methods with copper pipes are becoming outdated and insufficient, prompting the development of alternative cooling methods. In this study, a two stage Peltier module cooler was designed using Peltier modules, and its performance in cooling the processor was evaluated. The two stage Peltier module was created by thermally connecting two Peltier modules in series and tested under different experimental conditions. In the first experiment, the manufactured two-stage Peltier module was placed in the experimental setup with its surfaces exposed to air. Both the cold and hot surfaces were in contact with air, allowing heat transfer through natural convection. Afterward, power was supplied, and the surface temperatures were observed, and with the application of power, it was observed that the temperature of the hot surface increased from 34.8°C to 110.2°C, while the temperature of the cold surface rise from 24.2°C to 67.1°C. In the second experiment, a heat sink and a fan were mounted on the hot surface of the two stage Peltier module to evaluate cooling performance. As a result of these experiments, it was observed that with the application of the cooler, the minimum cold surface temperature dropped to -2.3°C, while the maximum hot surface temperature reached 26°C. In the third experiment, the Peltier modules cooling performance was tested on a micro heater instead of air. In these experiments, four different powers were applied to the micro-heater, and at the maximum power of 9.9 W, the lowest cold surface temperature observed was 126.4 °C. Finally, the two stage Peltier module system was directly applied to a computer processor to observe its cooling performance under real-use conditions. The experiments showed that the two stage Peltier module cooler reduced the processor temperature. In addition, under the same ambient conditions, it was observed that computer cooler reduced the microprocessor temperature to 62°C, while the cooler using the two-stage Peltier module reduced the microprocessor temperature to 43°C at the same microprocessor clock speed.
It was determined that as power was supplied to the Peltier module, the temperature difference between the two surfaces increased, but there was no significant change in the temperature of the hot surface. Additionally, it was observed that the performance of the Peltier module varied with different power values. The article demonstrates that the two stage Peltier module can be used as an effective solution for processor cooling applications.

Keywords: Thermoelectric cooler, Processor cooler, Two-stage Peltier module, Temperature difference, Cooling method

Article History:
Received: 10.09.2024
Accepted: 25.10.2024
Published Online: 25.12.2024

1. Introduction

Computers have become an important part of our lives. They play a crucial role not only in our personal use but also in the development of technology. Engineering analyses, designs, and autonomous controls are performed using computers. Therefore, computers have both facilitated the advancement of technology and developed alongside it. One of the most important components of a computer is the processor, its brain. The processor determines the speed at which a computer processes data. Hence, to improve computers and make the operations faster, the speed of the processor needs to be increased. To accelerate processors, the number of transistors within them must be increased [1]. As the number of transistors increases, the electric current passing through the processor also increases, resulting in a rise in the amount of heat generated by the processors [2]. When this heat is not dissipated, the processor and other surrounding electronic components can be damaged. For this reason, the maximum operating temperature of processors has been set at 90°C [3]. When this temperature is reached, the processor speed decreases [4]. Cooling electronic components has always been one of the significant topics in engineering. Different methods are used to cool processors. The most common of these is the cooling technique using copper pipes. This method, with advancing technology, has started to become outdated and insufficient, leading to the emergence of alternative methods. Various methods, such as liquid cooling and compressors, have begun to be used. Cooling processors with thermoelectric methods is another area of study [5]. When power is applied to thermoelectric modules, which are made of p-

Cite as: F. Uysal, Çobaner S, "Experimental Investigation of the Effect of Two-Stage Peltier Application on the Temperature of a Microprocessor," *Sakarya University Journal of Computer and Information Sciences*, vol.7, no. 3, pp. 460-469, 2024. Doi: 10.35377/saucis…1546785

and n- type semiconductors, one surface releases heat to the environment while the other surface absorbs heat, creating a temperature difference between the two surfaces [6]. This phenomenon is called the Peltier effect, and devices that operate in this way are called Peltier modules (PM) [7]. The absence of moving parts allows them to operate quietly and without vibration, requiring less maintenance and enabling long-term use [8]. In addition, the operating system can be easily controlled and can respond to small energy changes, such as 0.1 W, allowing more precise results in temperature control [9]. PM's are used in cooling applications across various fields. In the healthcare sector, they are used in cooling containers and tissue preparation applications, while in the military, they are used in areas such as night vision cameras and infrared detectors [10]. PM's are frequently used in the cooling of electronic components [11].

In the literature, Şahin and Işık (2023)[12] compared a liquid cooling system cooled by the Peltier method with a fan cooling system at a certain processor speed. The experiments used an AMD Athlon 4850E processor, a 12706 model PM with dimensions of 40x40, and a fan cooling system manufactured by AMD. Four different experiments were conducted at a room temperature of 24 °C. While the processor, running at full capacity, could be maintained at a temperature of 60 °C with the fan cooling system, the liquid cooling system designed with the PM could lower the temperature to 37 °C. Additionally, the cooling performance of the PM system was tested by operating it at 12V 2.85A, with the lowest temperature recorded as 20 °C and the highest as 28 °C. When the microprocessor was cooled using the PM, an increase in microprocessor performance of 40% was observed at the lowest processor speed, and 9.3% at the highest speed. Toren and Mollahasanoğlu (2022)[13] aimed to cool an oil-type transformer using a PM. A 5 kVA oil-type transformer was used in the study, and the results were compared with the cooling power of ester and naphthenic oils. The TEC1-12706 model PM was employed as the PM. As a result of the experiments, the PM reduced the temperature of the cabinet body by 15-20% compared to the oil cooling system. Ali Khan and others (2017)[14] compared the use of a heat pipe and a PM for cooling a laptop. Additionally, a second heat pipe was added, with the aim of dissipating heat from this pipe using the Peltier method. Experiments were conducted with a thermal design power of 35 W. As a result of the experiments, it was measured that the temperature of the second heat pipe decreased by 5 °C. It was also observed that the thermal design power decreased by 2.25 W. In the experiment conducted with both the heat pipe and the PM, the temperature was reduced by 2.3 °C, and the thermal design power decreased by 1.7 W. A cost analysis was conducted as a result of these findings. W. Y. Chen and others (2022)[15] conducted a comprehensive review of the advancements in state-of-the-art on-chip PM and summarized the related fundamentals, materials, designs, and system logic. Additionally, the power requirements for cooling microprocessors over the past five years were compared. The study focused on materials commonly used in both chips and PM and reviewed previous research. As a result, research was conducted to improve the design, performance, and applications of on-chip PMs in the future. Zaferani and others (2021)[10] conducted a review on the design, optimization, and development of PM as coolers for medical applications. In their study, they specifically researched the application of PMs in transport containers and wearable technologies. They noted that the features determining the thermal optimization of PMs include the use of efficient thermoelectric materials, effective heat sink and absorption design, especially the application of phase-changing materials and heat pipes, and the production of flexible thermoelectric devices. As a result, they highlighted that new PM technologies are gradually replacing traditional coolers due to their advantages, such as small size, flexibility, and pollution-free characteristics, particularly in wearable technologies and medical applications. Z. Liu and others (2022)[16] designed a two-stage PM to enhance its thermal performance. In their study, they compared the thermal performance of square-type, cubic-type, and pyramid-type two-stage PMs, finding significant improvements in cold-side temperature and the temperature difference between the two surfaces. The experimental results showed that the cubic-type two-layer PM could reach a low temperature of -49.5 °C, which is lower than the uninsulated pyramid-type two-layer PM (-34.3 °C) and the square-type two-layer PM (-38.6 °C).

In this study, the aim is to cool a computer processor using two stages PM. When we review the literature, various types of coolers have been used in studies on microprocessor cooling. Among these coolers, Peltier module research represents a small part. In addition, studies related to cooling have mostly been conducted on coolers with a single Peltier module. Theoretical analyses have been carried out on two-stage Peltier modules used as coolers, but experimental studies in this area are lacking. In this study, a cooler was designed using a two-stage PM. The performance of this cooler was evaluated through a series of experiments and finally applied to a microprocessor, where it was compared with the traditional microprocessor cooling method, the heat pipe. Additionally, the change in cooling performance was observed when a heat sink was mounted on the hot surface of the dual PM [17].

2. Materials and Methods

In the experiments conducted in the study, a 30 V_{DC}-30 A_{DC} DC power supply, a 14 V_{DC}-3 A_{DC} DC power supply, a 3 A_{DC}-30 V_{DC} DC power supply, a cooling system consisting of a heat sink and fan, a micro heater, a laptop, and an insulated experimental setup were used. A thermal camera and a thermometer were used as measuring instruments.

In the experiments, the TEC1-12709 model PM was used (Table 1). The key feature of this model is its maximum operating temperature of 135°C. Since it is known that the processor can't work temperatures up to 90°C, the PM's maximum operating temperature must be above this value; therefore, a PM model with a temperature rating higher than 90°C was selected.

Here, the two stage PM system is powered by a 30 V_{DC}-30 A_{DC} power supply, the fan by a 14 V_{DC}-3 A_{DC} power supply, and the micro heater by a 3 A_{DC}-30 V_{DC} power supply (Table 2). Since the PMs are used in a two-stage configuration, a maximum of 80 W power can be supplied. The fan operates at a voltage of 12 V and a current of 3 A. Although the micro heater is

designed to handle a maximum power of 90 W, it is operated up to a maximum of 60 W due to the high surface temperature, which could cause damage to its surroundings.

Table 1. Peltier Module Properties

Model	V_{max}(V)	A_{max}(A)	T_{max}(°C)	Dimension (mm)	P_{max}(W)
TEC1-12709	9	14	135	40x40	80

Table 2. Power Supplies Properties and Usage Area

Model	V_{max}(V)	A_{max}(A)	P_{max}(W)	Usage area
30 V_{DC}-30 A_{DC}	30	30	900	Peltier Module
14 V_{DC}-3 A_{DC}	14	3	42	Fan
3 A_{DC}-30 V_{DC}	3	30	90	Micro heater

The cooling equipment was used to dissipate heat from the hot surface of the PM. It consists of a fin and a fan as components. The fin is used as a heat sink (Table 3), while the fan is used to increase the heat transfer rate through forced convection. Fan diameter is 120 mm, and the air flow speed is 1.25 m/s with the fan. The aim is to enhance the performance of the PM by dissipating heat from its hot surface.

Table 3. Heat Sink Properties

Properties	
Material	Aluminum
Fin number	88
Fin width (mm)	4
Fin thickness (mm)	0.4
Fin length (mm)	34

In the experiments, two different measuring devices were used. One of these is a thermal camera, the UNI-T UTI brand model 720E (Table 4). The thermal camera was used to measure the surface temperatures of the PM in experiments involving a two-stage PM and a cooled two-stage PM. The other measuring device was a Cem brand datalogger and thermocouple (Table 5). The thermometer was passed through the insulated surface in the experiments conducted on the microheater to measure the surface temperature of the microheater. Since the thickness of the microheater was less than 10 mm, the temperature difference between its two surfaces was neglected, and the temperature obtained from the thermometer was accepted as the cold surface temperature of the PM in these experiments.

Table 4. Thermal Camera

UNI-T UTI 720E – Technical properties	
IR resolution (Pixel)	256×192
Screen	3.5"IPS (640×480)
Thermal sensitivity (NETD)	≤50mK
Frame rate	≤25Hz
Temperature measuring range	-20 °C ~ 550 °C
Digital camera resolution	2MP
USB	C type
Dimensions	260mm×97mm×99mm

Table 5. Datalogger and Thermometer

CEM – Data logger and thermocouple technical properties	
Temperature sensitivity	±1°C
Temperature resolution	0.1°C
K type thermocouple temperature measurement range	-200°C ... +1372°C
J type thermocouple temperature measurement range	-210°C ... +1100°C
Memory capacity	18.000 units per thermocouple input
Operating temperature	0°C ... +50°C
Power supply	Battery

In the experiments, the HP PAVILION DV2 model was used. The experiments were conducted on a processor of the AMD Athlon™ Neo MV-40 model, which has a thermal design power of 15 W (Table 6). The experiments were conducted at the processor's maximum speed of 1.6 GHz.

Table 6. Processor Properties

Model	Clock (GHz)	Thermal design power (W)
AMD Athlon™ Neo MV-40	1.6	15

In the first part of the experiments, the hot surface of one of the two PM was placed in contact with the cold surface of the other, mounting them thermally in series. Thermal paste was used to increase the contact area between the surfaces. Then, power was supplied to the two stage PM, and the temperatures of the hot and cold surfaces were measured. In this part, the performance of the PM was evaluated when heat transfer was carried out by natural convection. In the second part, a cooling fin and fan system were mounted on the hot surface of the two stage PM system, and experiments were conducted to evaluate the effect of heat dissipation from the hot surface on the cooling performance of the PM. In both experiments, the two stage PM was initially supplied with a current of 1 A, which was increased by 1 A increments up to the maximum temperature value of the PM. As the surfaces were exposed, their temperatures were measured with a thermal camera.

In the third part of the experiment, tests were conducted on a designed micro heater before testing the PM on a processor (Figure 1). Looking at the experimental setup, the heat starts from the micro heater, then proceeds through the 1st PM, the 2nd PM, and finally reaches the finned surface. Since processors can easily malfunction at high temperatures, the experiments began on the micro heater, which served as a prototype for the processor. Here, the heat emitted by the processor was simulated on the surface of the micro heater to test the two stage PM system's ability to dissipate heat from this surface. In the insulated system created, one surface of the micro heater was insulated while the other was in contact with the cold surface of the PM. With this design, most of the power supplied to the system was dissipated as heat from the surface where the micro heater was in contact with the PM. To measure the temperature of the surface where the PM and the micro heater were in contact, measurements were taken from the insulated surface; a thermometer was passed through the insulation to contact the surface of the micro heater. Since the micro heater was thin, the temperature difference between its two surfaces was considered negligible.

Test parameters and conditions are listed in table 7. These parameters and conditions used in experiments.

Table 7. Test Parameters and Conditions

Test conditions		Test parameters
Ambient temperature (°C)	23	Current (A)
Fan speed (RPM)	1500	Power (W)
Contact surface conductivity (W/m.K)	8.5	
CPU usage rate (%)	100	

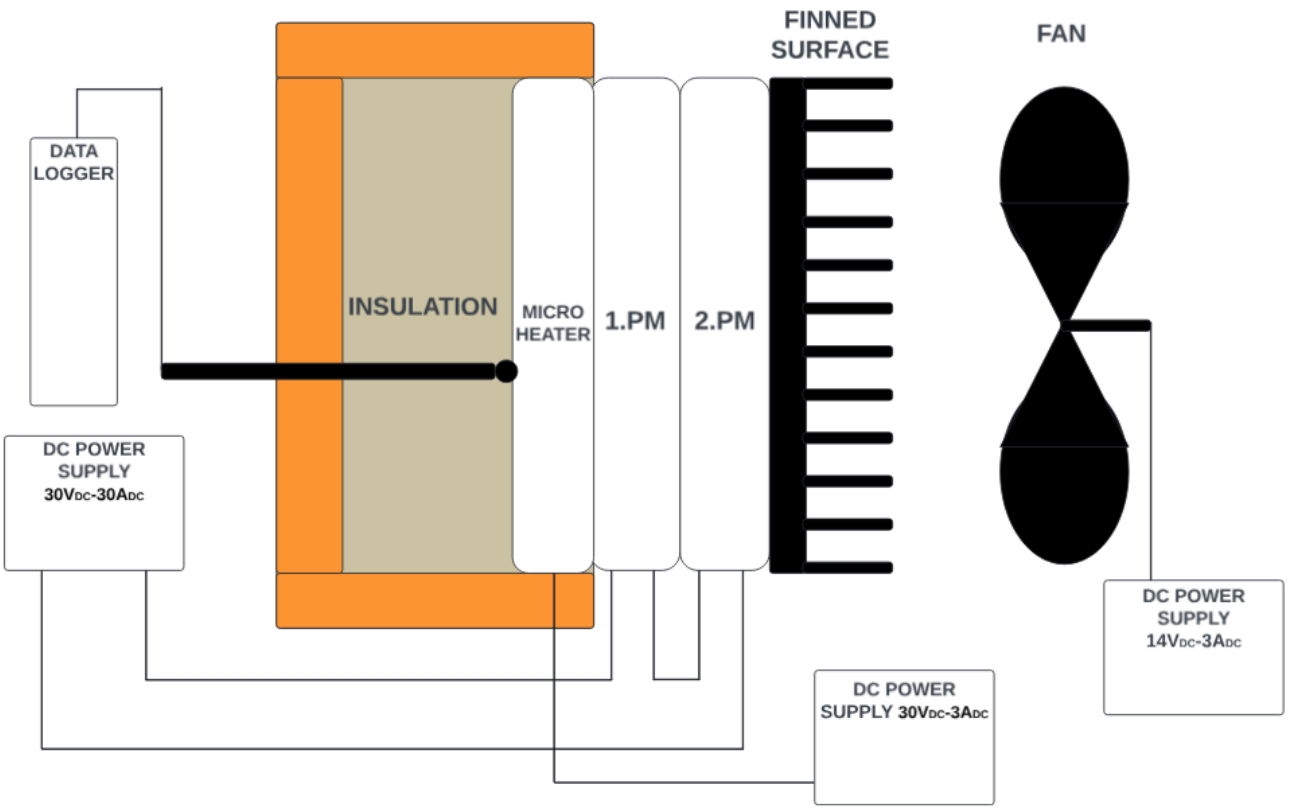


Figure 1. A Schematic Diagram of a Two Stage PM on Micro Heater Experiment

In the fourth and final part of the experiments, the two stage PM system was applied to a computer processor, and the processor temperatures were observed (Figure 2). In this part, the experiments were conducted with the graphics card of the computer being cooled by its own cooling system, while the processor was cooled by both a cooling system consisting of fins and a fan, and the two stage PM and cooling system. It was observed that when the processor was cooled only with the two stage PM, the computer shut down about 5 minutes after being turned on due to high temperatures. Therefore, the experiments were conducted with a cooling fin and fan system mounted on the hot surface of the two-stage PM. In this part, the experiments began without supplying power to the two stage PM; the power supplied to the system was increased in 1 W increments, and the experiments were concluded with a final power supply of 6 W. The AIDA64 program was used to apply maximum load to the processor, and the processor temperature was observed with The AIDA64. The experiments were successfully conducted, and the results were evaluated.

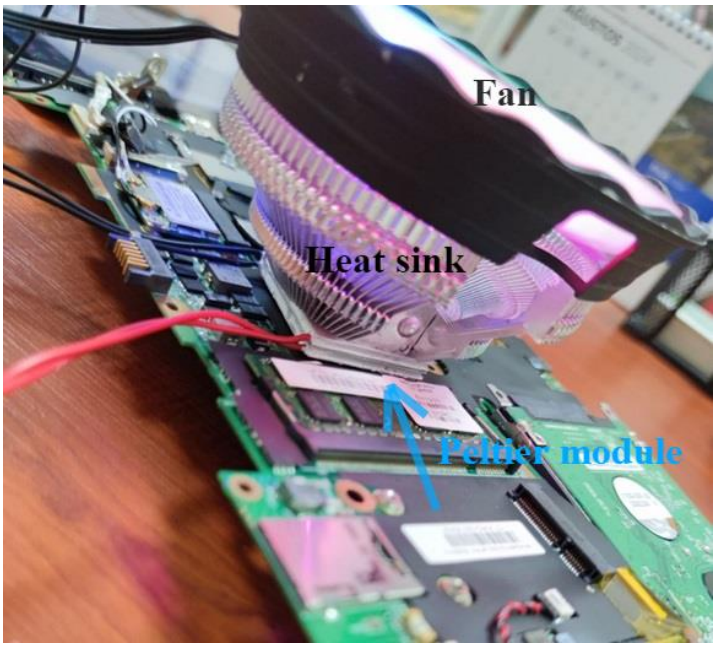


Figure 2. Two Stage PM Module with Heat Sink and Fan Applied to Microprocessor

3. Results and Discussions

In the experiments, the performance of both the two stage PM without cooler and the two stage PM with the cooler applied to the hot side was initially evaluated. In the experiments conducted, how the power supplied to the system affected the surface temperatures of the two stage PM was examined. The obtained results were presented, and finally, results from the experiments conducted on the processor were obtained.

The first experiment observed the temperature changes on the hot and cold surfaces of the two stage PM were measured at different powers applied to the PM (Figure 3). The power supplied to the system is the total power consumed by the two PM's. In the first experiment of this section, a total of 0.8025 W power was supplied to the PM's, and the hot surface temperature was measured as 34.8°C. The hot surface temperature continued to increase with the power supplied to the system and reached 110.2°C when 7.8 W of power was applied. The experiment was concluded at this point as it approached the maximum temperature the PM could withstand. In the initial experiment, the cold surface temperature was measured as 24.2°C, which is above the room temperature of 23°C. Subsequently, the cold surface temperature continued to increase with the power supplied to the PM. In the two stage PM experiment, the temperature difference between the two surfaces increased as the power supplied to the system increased. When 0.8025 W of power was supplied to the system, the temperature difference between the two surfaces was 10.6°C; when 2.2 W was supplied, it was 17.6°C; at 4.11 W, it was 29.6°C; and at 7.8 W, it was 43.1°C.

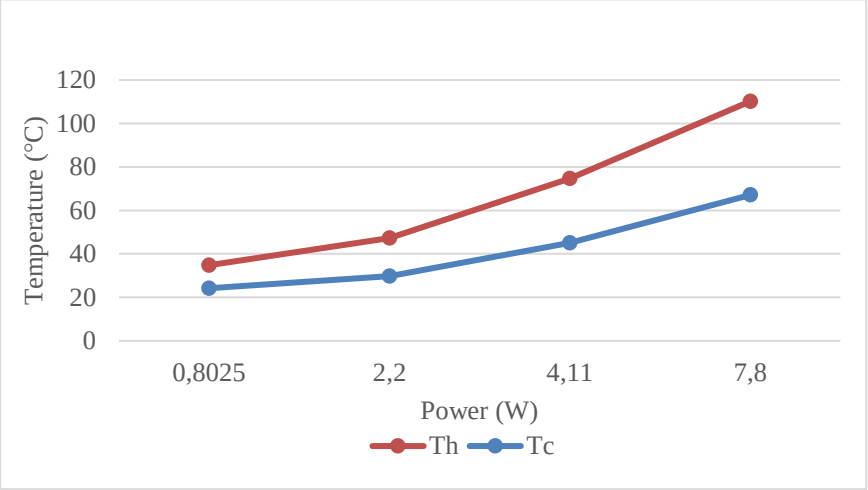


Figure 3. The First Experiment, Hot and Cold Surface Temperatures with the Power (W) Applied to the Two Stage PM

In the second part of the experiments, a cooling fin and fan system was applied to the hot surface of the PM. The two stage PM experiment began with a power of 0.4825 W supplied to the system (Figure 4). At the starting point, the hot surface temperature was measured at 23.7°C, and the cold surface at 13.2°C. As the power supplied to the system increased, the temperature difference between the two surfaces also increased, reaching the highest temperature difference of 28.3°C at a power level of 5.38 W. Furthermore, the lowest measured temperature value of -2.3°C was obtained when 5.38 W of power was supplied to the system. The hot surface temperature increased as power was supplied to the system; while the temperature difference between the first two measurements was 0.5°C, it was 1.1°C between the last two measurements. This indicates that the hot surface temperature did not increase linearly but rather showed an increasing slope.

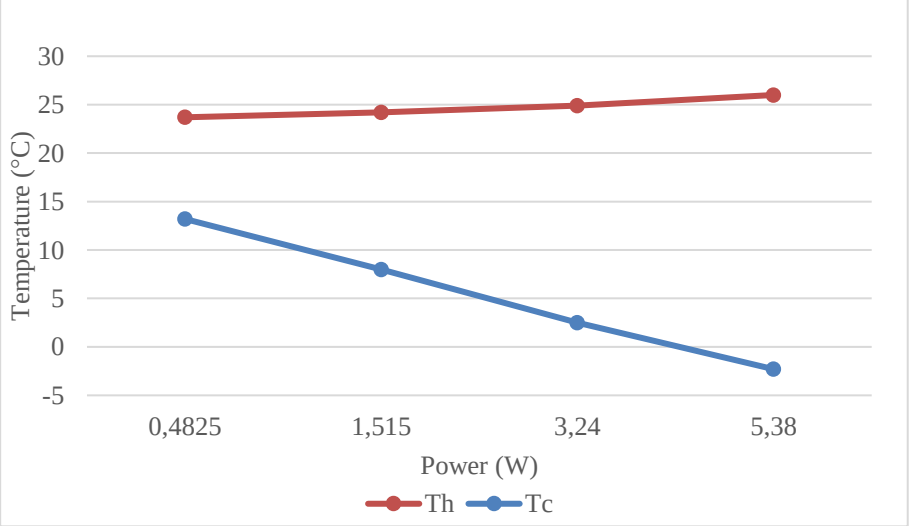


Figure 4. Hot and Cold Surface Temperatures with the Power (W) Applied to the Two Stage PM at Second Experiment

Another part of the study is PM cooler applied to micro heater. Micro heater experiments were conducted because microprocessors degrade above 90°C, and the behavior of a two-stage PM cooler was observed on the micro heater. However, since microprocessor temperatures change more rapidly compared to the micro heater, experiments were also conducted on the microprocessor after the micro heater tests. This allowed the performance of the two-stage PM cooler applied to the microprocessor to be clearly demonstrated. In the third experiment using two stage PM as the cooler, it was observed that the cold surface temperature increased as the power supplied to the micro heater increased (Figure 5). The power (W) values here represent the heat dissipated from the surface of the micro heater, while the current (A) values indicate the total current supplied to the system. As the current applied to the PM increased, the cold surface temperature reached its lowest point at the current value of 5 A, and the temperature began to rise as the current value continued to increase.

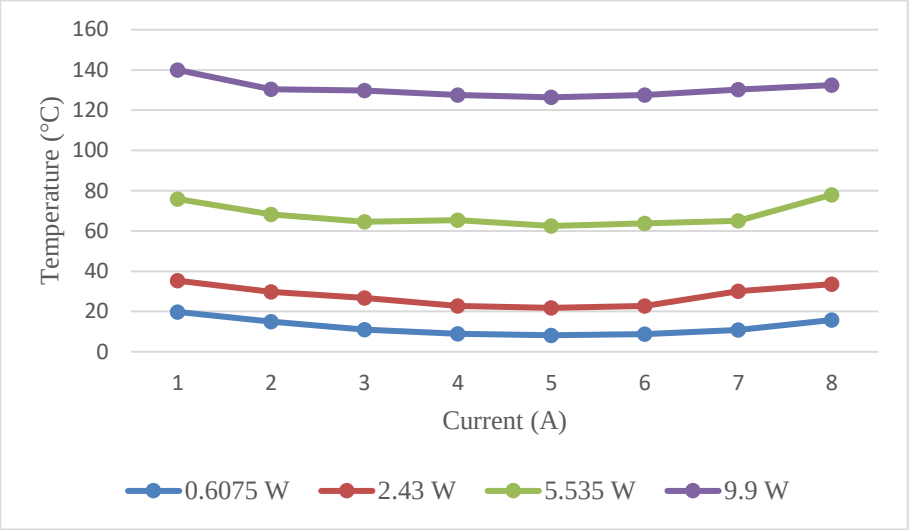


Figure 5. Cold Surface Temperatures of Cooled Two Stage PM's at Different Micro Heater Power

When examining the hot surface temperatures of the two stage PM, the highest temperature value was measured at 95.8°C when 9.9 W of heat was dissipated from the micro heater and 8 A of current was applied to the PM (Figure 6). When comparing these four experiments, the highest temperature was 95.8°C at a power level of 9.9 W on the micro heater, while the lowest temperature was measured at 79.5°C when 0.6075 W of power was applied to the micro heater.

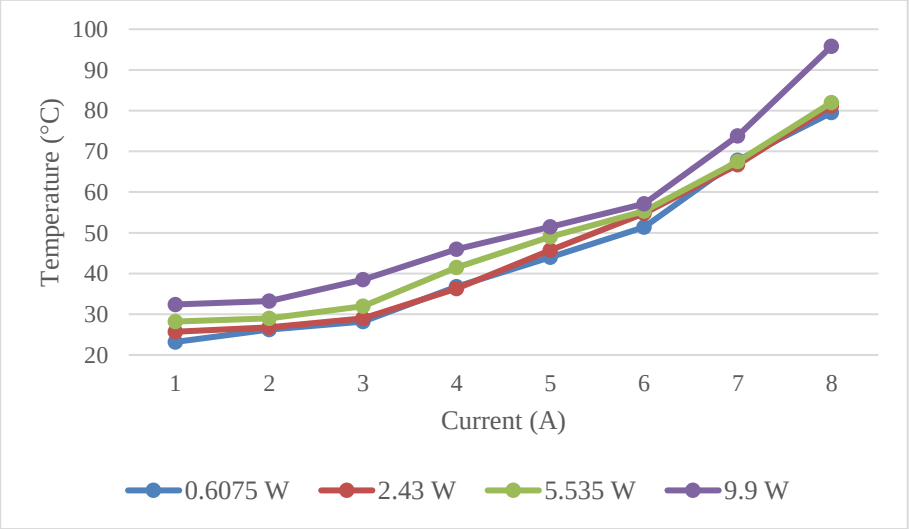


Figure 6. Hot Surface Temperatures of Cooled Two Stage PM's at Different Micro Heater Power

This section we made third experiment, two stage PM and cooler equipment (finned surface and fan) applied to micro heater. The experiment started by applying 0.66 W of power to the PM, and the hot surface temperature was measured at 32.4°C, while the cold surface temperature was 140°C (Figure 7). It was observed that as the power supplied to the PM's increased, the hot surface temperature increased, but the cold surface temperature reached its lowest point of 125.9°C at a power of 33.6 W and then started to rise again. Additionally, at this point, the increase in the hot surface temperature was observed to accelerate. The power level of 33.6 W represents the optimum point for the two stage PM system to cool a surface emitting 9.9 W of heat, and the temperature decreased to 125.9°C; beyond this point, the temperature began to increase as more power was applied.

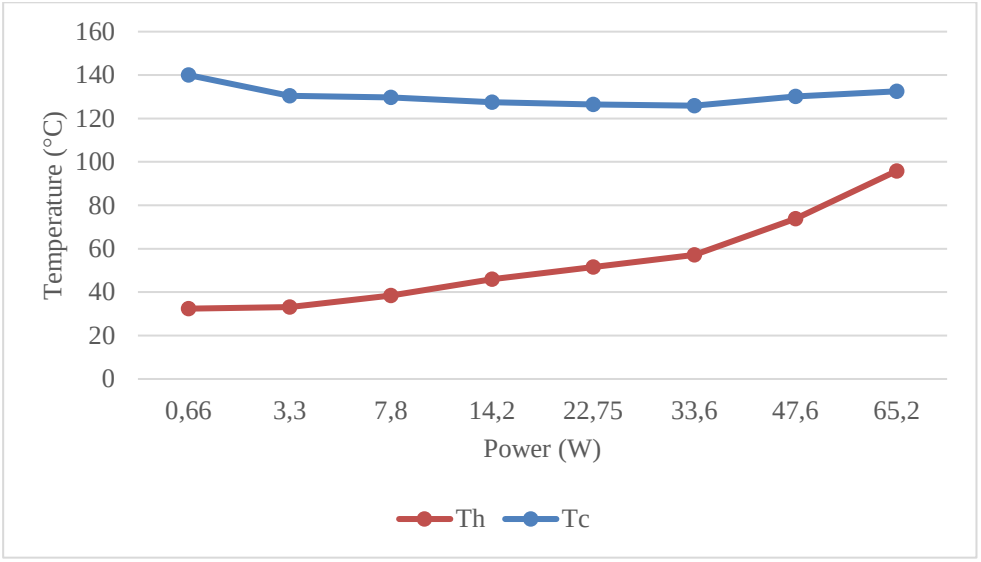


Figure 7. In the Third Experiment, PM Surface Temperatures with the Power (W) Applied to the Two Stage PM with Cooler in the Micro Heater (9.9W)

A series of experiments were conducted on the processor. Both the computer's original cooler and the designed two stage PM cooler were used in the experiments (Figure 8). One of the boundary conditions of the experiment was to apply maximum load to the processor using the AIDA64 program, resulting in 100% CPU utilization at maximum speed. Another boundary condition was that all experiments were conducted in an environment with an ambient temperature of 23°C. Initially, the experiment was performed using the computer's original cooler, which consists of a copper pipe, fin, and fan system, keeping the computer processor at a temperature of 62°C. The experiments then continued with the two stage PM cooler. Heat was dissipated from the hot surface of the PM using a fin and fan system. In the experiments, power was supplied to the two stage PM starting from 0 W, increasing by increments of 1 W. When no power was supplied to the PM, the processor temperature was measured at 72°C. It was observed that as power was supplied to the PM, the processor temperature decreased, reaching as low as 43°C when 6 W of power was supplied. At the start of the experiment, when power was supplied to the system, the temperature dropped from 72°C to 56°C, and after this point, the temperature decreases lessened with each step. When 5 W of power was supplied to the system, a temperature of 45°C was measured, and at 6 W, a temperature of 43°C was observed. The experiments were concluded at this point due to the temperature difference being only 2°C.

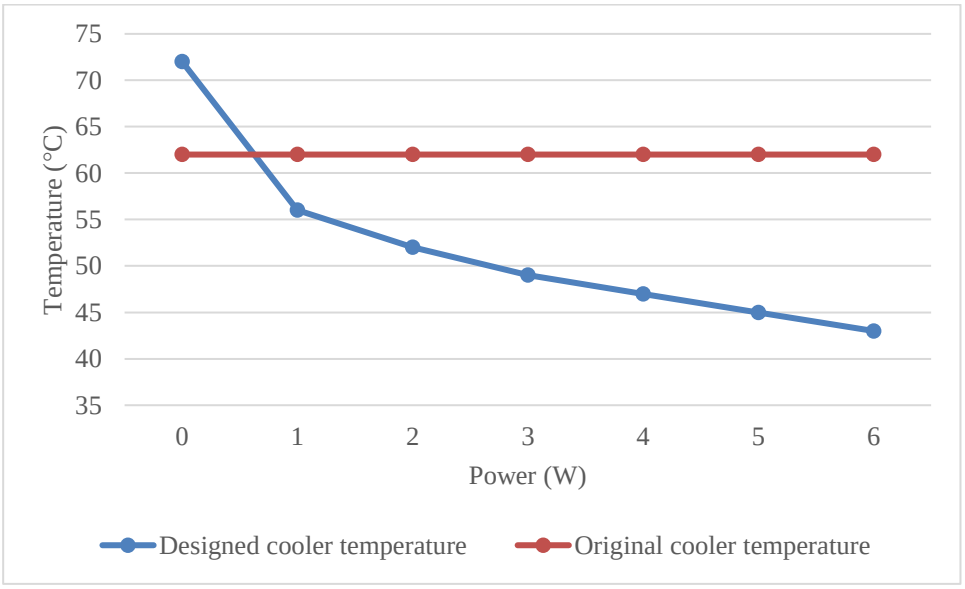


Figure 8. Experimental Results of PM Cooler on the Microprocessor

The use of two stage PMs provides an alternative solution in cooling systems, shedding light on PM studies. In this study, the cooler designed with two stage PM demonstrates that it can effectively cool electronic components that generate high heat, such as processors, while offering valuable insights to researchers on enhancing the performance of the two stage PM through a heat sink. Additionally, this study will provide researchers with insights into the advantages and cooling performance of the two stages PM.

4. Conclusion

In the study conducted, a cooler was designed using a two stage PM and applied to a computer processor, yielding the results. For the cooler design, two PMs were thermally mounted in series, allowing heat to be dissipated from the hot surface using fins. In the experiments conducted solely on the PM, it was observed that as the power supplied to the PM increased, the temperature difference between the two surfaces also increased. Initially, when 0.8025 W of power was supplied, the temperature difference was 10.6°C, which rose to 43.1°C when 7.8 W was supplied to the system. When heat was dissipated from the hot surface using fins and a fan, the temperature difference between the two surfaces increased, but there was no significant change in the temperature of the hot surface. A temperature difference of 2.3°C occurred on the hot surface between the beginning and the end of the experiment. However, there was a decrease in the cold surface temperature, dropping from an initial temperature of 13.2°C to -2.3°C by the end of the experiment. Additionally, forced convection was achieved by blowing air onto the fins with a fan. As a result of the experiments conducted in this section, it was determined that the cooler applied to the hot surface of the PM increased the cooling performance of the PM. Subsequently, the designed cooler was tested by being applied to a micro heater before use. The PM was tested for different power values supplied to the micro heater, and it was observed that the PM reached its limit temperatures when 9.9 W of power was supplied. Finally, experiments were conducted on the processor. In these experiments, it was observed that the PM-equipped cooler lowered the processor temperature to a greater extent compared to the computer's original cooler.

This study provides a foundation for determining the optimal operating temperature of the microprocessor and for future work aimed at maintaining the microprocessor temperature at its optimal value. Additionally, it can be used as a basis for investigating the effect of P and N material leg heights on cooling performance. Furthermore, it supports efforts to reduce the negative impact of electrical conductors on cooling performance.

References

[1] Y. Sun, N. B. Agostini, S. Dong, and D. Kaeli, "Summarizing CPU and GPU Design Trends with Product Data," 2019, [Online]. Available: http://arxiv.org/abs/1911.11313.

[2] M. Bahiraei and S. Heshmatian, "Application of a novel biological nanofluid in a liquid block heat sink for cooling of an electronic processor: Thermal performance and irreversibility considerations," *Energy Convers. Manag.*, vol. 149, pp. 155–167, 2017, doi: 10.1016/j.enconman.2017.07.020.

[3] R. Chein and G. Huang, "Thermoelectric cooler application in electronic cooling," *Appl. Therm. Eng.*, vol. 24, no. 14–15, pp. 2207–2217, 2004, doi: 10.1016/j.applthermaleng.2004.03.001.

[4] N. Putra and W. N. Septiadi, "Improvement of heat pipe performance through integration of a coral biomaterial wick structure into the heat pipe of a CPU cooling system," *Heat Mass Transf. und Stoffuebertragung*, vol. 53, no. 4, pp. 1163–1174, 2017, doi: 10.1007/s00231-016-1890-6.

[5] C. Gayner and K. K. Kar, "Recent advances in thermoelectric materials," *Prog. Mater. Sci.*, vol. 83, pp. 330–382, 2016, doi: 10.1016/j.pmatsci.2016.07.002.

[6] H. Jouhara *et al.*, "International Journal of Thermo fl uids Thermoelectric generator (TEG) technologies and applications," vol. 9, 2021.

[7] Sonal Renge, Yashika Barhaiya, Shikhar Pant, and Shubham Sharma, "A Review on Generation of Electricity using Peltier Module," *Int. J. Eng. Res.*, vol. V6, no. 01, pp. 453–457, 2017, doi: 10.17577/ijertv6is010308.

[8] M. Haci and Z. Kahraman, "Termoelektrik sistemli yemek taşıma modülü tasarımı ve analizi," *J. FCE*, vol. 4, no. December, pp. 65–71, 2016.

[9] S. Shoeibi, H. Kargarsharifabad, M. Sadi, A. Arabkoohsar, and S. A. A. Mirjalily, "A review on using thermoelectric cooling, heating, and electricity generators in solar energy applications," *Sustain. Energy Technol. Assessments*, vol. 52, no. PB, p. 102105, 2022, doi: 10.1016/j.seta.2022.102105.

[10] S. H. Zaferani, M. W. Sams, R. Ghomashchi, and Z. G. Chen, "Thermoelectric coolers as thermal management systems for medical applications: Design, optimization, and advancement," *Nano Energy*, vol. 90, no. PA, p. 106572, 2021, doi: 10.1016/j.nanoen.2021.106572.

[11] H. Y. Zhang, Y. C. Mui, and M. Tarin, "Analysis of thermoelectric cooler performance for high power electronic packages," *Appl. Therm. Eng.*, vol. 30, no. 6–7, pp. 561–568, 2010, doi: 10.1016/j.applthermaleng.2009.10.020.

[12] M. ŞAHİN and P. D. H. IŞIK, "Zorlu Ortam Koşullarında Çalışan Bilgisayarlar için Geliştirilen Peltier Etki ile Soğutulan Sıvı Soğutma Sisteminin Fanlı Soğutma Sistemi ile Karşılaştırılması," *Sürdürülebilir Mühendislik Uygulamaları ve Teknol. Gelişmeler Derg.*, vol. 6, no. 2, pp. 156–166, 2023, doi: 10.51764/smutgd.1318383.

[13] M. TOREN and H. MOLLAHASANOGLU, "Termoelektrik Soğutma Sisteminin Alternatif Transformatör Soğutma Sistemi Olarak Performansı," *Eur. J. Sci. Technol.*, no. 32, pp. 498–507, 2022, doi: 10.31590/ejosat.1039820.

[14] M. A. Ali Khan, E. Uzgören, and A. Muhtaroğlu, "Investigation of secondary cooling design enhancements in thermally limited compact notebooks," *Turkish J. Electr. Eng. Comput. Sci.*, vol. 25, no. 2, pp. 1574–1586, 2017, doi: 10.3906/elk-1602-50.

[15] W. Y. Chen, X. L. Shi, J. Zou, and Z. G. Chen, "Thermoelectric coolers for on-chip thermal management: Materials, design, and optimization," *Mater. Sci. Eng. R Reports*, vol. 151, no. August, p. 100700, 2022, doi: 10.1016/j.mser.2022.100700.

[16] Z. Liu *et al.*, "Design and optimization of a cubic two-stage thermoelectric cooler for thermal performance

enhancement," *Energy Convers. Manag.*, vol. 271, no. September, p. 116259, 2022, doi: 10.1016/j.enconman.2022.116259.
[17] S. Çobaner, "Termoelektrik Yöntemler ile Mikroişlemci Soğutucu Tasarımı," Sakarya Uygulamalı Bilimler Üniversitesi, 2024.

Article Information Form

Acknowledgments
This work was supported by Sakarya University of Applied Sciences Scientific Research Projects Unit under Grants 195-2024

Conflict of Interest Notice
The authors declare that there is no conflict of interest regarding the publication of this paper.

Support
This study was supported by the Sakarya University of Applied Sciences Scientific Research Projects Commission (Project No: 195-2024)
This project is derived from Sinan Çobaner's master thesis.

Ethical Approval and Informed Consent
It is declared that during the preparation process of this study, scientific and ethical principles were followed, and all the studies benefited from are stated in the bibliography.

Availability of data and material
Not applicable

Plagiarism Statement
This article has been scanned by iThenticate ™.

SAKARYA UNIVERSITY JOURNAL OF COMPUTER AND INFORMATION SCIENCES
http://saucis.sakarya.edu.tr/
e-ISSN: 2636-8129
Publisher: Sakarya University
Vol. 7, No. 3, 470-481, 2024
DOI: 10.35377/saucis...1564497
SAUCIS
SAKARYA UNIVERSITY

RESEARCH ARTICLE

Quantum-Inspired Data Embedding for Unlabeled Data in Sparse Environments: A Theoretical Framework for Improved Semi-Supervised Learning without Hardware Dependence

Shawn Ray

Lone Star College 5000 Research Forest Drive, The Woodlands, Texas 77381, USA

Corresponding author: Shawn Ray, Lone Star College, 5000 Research Forest Drive The Woodlands, Texas 77381, USA
E-mail address: shawnray5699@gmail.com

Article History:
Received: 10.10.2024
Accepted: 05.12.2024
Published Online: 31.12.2024

ABSTRACT
This paper introduces an innovative theoretical framework for quantum-inspired data embeddings, grounded in foundational concepts of quantum mechanics such as superposition and entanglement. This approach aims to advance semi-supervised learning in contexts characterized by limited labeled data by enabling more intricate and expressive embeddings that capture the underlying structure of the data effectively. Grounded in foundational quantum mechanics concepts such as superposition and entanglement, this approach redefines data representation by enabling more intricate and expressive embeddings. Emulating quantum superposition encodes each data point as a probabilistic amalgamation of multiple feature states, facilitating a richer, multidimensional representation of underlying structures and patterns. Additionally, quantum-inspired entanglement mechanisms are harnessed to model intricate dependencies between labeled and unlabeled data, promoting enhanced knowledge transfer and structural inference within the learning paradigm. In contrast to conventional quantum machine learning methodologies that often rely on quantum hardware, this framework is fully realizable within classical computational architectures, thus bypassing the practical limitations of quantum hardware. The versatility of this model is illustrated through its application to critical domains such as medical diagnosis, resource-constrained natural language processing, and financial forecasting—areas where data scarcity impedes the efficacy of traditional models. Experimental evaluations reveal that quantum-inspired embeddings substantially outperform standard approaches, enhancing model resilience and generalization in high-dimensional, low-sample scenarios. This research marks a significant stride in integrating quantum theoretical principles with classical machine learning, broadening the scope of data representation and semi-supervised learning while circumventing the technological barriers of quantum computing infrastructure.

Keywords: Quantum-inspired data embedding, Semi-supervised learning, High-dimensional data, Superposition, Entanglement

1. Introduction

This paper introduces a novel approach to enhancing semi-supervised learning through quantum-inspired data embeddings. Semi-supervised learning has gained considerable attention for its potential to leverage limited labeled data, a critical advantage in fields where annotation is costly or time-consuming. However, traditional machine learning models often struggle in high-dimensional spaces with few samples, leading to overfitting and inadequate generalization. This research addresses these limitations by exploring quantum-inspired techniques for data representation that simulate superposition and entanglement, key quantum mechanics concepts.

These quantum principles facilitate the encoding of intricate, multi-state data representations and relationships that conventional machine learning models find difficult to capture. By embedding these quantum-inspired ideas into classical computational frameworks, this research proposes a novel solution that doesn't rely on quantum hardware, making it feasible and scalable for real-world applications. In particular, this method is designed to improve the robustness and efficiency of semi-supervised learning in areas such as medical diagnosis, natural language processing (NLP), and financial forecasting, where data is often sparse. The datasets used in our analysis were chosen due to their high dimensionality and inherent sparsity, which align closely with the theoretical requirements of quantum-inspired embeddings. The presence of intricate, non-linear relationships in the data mirrors the complexity quantum principles are designed to capture, providing an ideal platform for evaluation.

This paper follows a structured approach: the Literature Review section highlights related work and the current state of quantum-inspired machine learning, mainly focusing on the challenges of data scarcity in semi-supervised contexts. The Project Description delves into the core theoretical contributions of this work, including the mathematical foundations and implementation details. We demonstrate the effectiveness of the proposed model across several domains, with extensive experimental results presented in the Results section. Finally, the Conclusion summarizes the findings and proposes future directions for integrating quantum concepts into classical learning models.

2. Glossary of Key Terms

Quantum Superposition:

A principle of quantum mechanics where a quantum system can exist in multiple states simultaneously. In this paper, it is used to represent data points as probabilistic mixtures of feature states, enhancing data embedding richness.

Quantum Entanglement:

A phenomenon where the states of two or more particles become interconnected, such that the state of one particle directly influences the state of the other(s), even when separated. This concept is adapted here to model relationships between labeled and unlabeled data.

Quantum-Inspired Data Embeddings:

A novel data representation approach emulating quantum mechanics principles like superposition and entanglement to improve learning efficiency and generalization in semi-supervised learning.

Semi-Supervised Learning:

A machine learning approach that uses both labeled and unlabeled data for training, often employed when labeled data is scarce.

High-Dimensional Data:

Datasets with a large number of features or dimensions, often lead to sparsity (many features with little or no value), which poses challenges for traditional machine learning methods.

Sparse Data:

Data in which many features have zero or near-zero values. Sparse datasets are common in fields like natural language processing and medical diagnosis.

Hilbert Space:

A mathematical concept describing an abstract multidimensional space where quantum states can be represented. In this work, it provides the foundation for quantum-inspired embeddings.

Density Matrix:

A mathematical representation of a quantum system's state, capturing probabilities of being in various configurations. Used in this research to model correlations between labeled and unlabeled data.

Graph Regularization:

A technique used to smooth and structure data embeddings by considering relationships (edges) between data points (nodes) in a graph representation.

Consistency Regularization:

A training strategy encouraging a model's predictions to be consistent under small perturbations or variations in the input data.

Gaussian Noise:

Random noise following a normal distribution added to datasets during testing to evaluate model robustness.

Classification Accuracy:

A performance metric measuring the proportion of correctly classified instances out of all instances.

Precision:

The ratio of correctly predicted positive observations to all predicted positive observations, indicating the model's relevance in positive prediction.

Recall:

The ratio of correctly predicted positive observations to all actual positive observations, showing the model's sensitivity.

F1-Score:

A metric that balances precision and recall, calculated as the harmonic mean of the two.

ROC-AUC (Receiver Operating Characteristic - Area Under Curve):

A performance measurement for classification problems, evaluating the trade-off between true positive rates and false positive rates.

Classical Machine Learning Frameworks:

Traditional computational models for training and testing data without leveraging quantum principles or hardware.

Unitary Operator:

A mathematical operator in quantum mechanics used to describe the evolution of quantum states. Here, it defines the transformation function for embeddings.

Graph-Based Approach:

A modeling technique where data points are treated as nodes and their relationships as edges in a graph, facilitating structural learning.

Self-Training:

A semi-supervised learning technique where a model iteratively labels unlabeled data based on its predictions and retrains on the newly labeled data.

Activation Function:

A mathematical function applied in neural networks to introduce non-linearity, enabling the model to learn complex patterns. Examples include ReLU and softmax.

Adam Optimization Algorithm:

An adaptive learning rate optimization algorithm commonly used in machine learning for gradient-based optimization.

3. Literature Review

3.1 Introduction to Quantum-Inspired Machine Learning

The intersection of quantum mechanics and machine learning has garnered significant interest in recent years, particularly in the context of data representation and processing [1], [2]. Quantum-inspired algorithms leverage principles from quantum mechanics to enhance classical computational techniques, particularly in scenarios where data is sparse or unlabeled [3], [4]. For instance, quantum mechanics introduces concepts such as superposition and entanglement, which can be emulated in classical systems to improve data representation and learning efficiency [5], [6]. These principles allow for the encoding of data in a manner that captures complex relationships and dependencies, thereby enhancing the performance of machine learning models in semi-supervised learning contexts.

3.2 Quantum Principles in Data Representation

The application of quantum principles to data representation has been explored in various studies [7], [8]. For example, the concept of quantum superposition enables the representation of data points as probabilistic mixtures of multiple states, which can lead to richer embeddings that capture the underlying structure of the data more effectively than traditional methods [6]. This approach is particularly beneficial in high-dimensional spaces where data is sparse, as it allows for a more nuanced understanding of the relationships between data points [9], [10]. Furthermore, entanglement can be utilized to model intricate dependencies between labeled and unlabeled data, facilitating improved knowledge transfer and structural inference within semi-supervised learning frameworks [5].

3.3 Challenges in Semi-Supervised Learning

Semi-supervised learning presents unique challenges, particularly in environments characterized by limited labeled data [11], [12]. Traditional machine learning models often struggle in such settings, leading to issues such as overfitting and poor

generalization [13]. The novelty of this framework lies in its unique application of quantum-inspired techniques, which not only provide sophisticated data embeddings but also leverage quantum principles to capture complex relationships within the data [10], [14]. This capability distinguishes it from traditional semi-supervised learning methods, which often struggle to utilize both labeled and unlabeled data effectively [15], [16]. For instance, the use of quantum-inspired embeddings has been shown to significantly enhance model resilience and generalization in scenarios where data is scarce, thereby improving the overall efficacy of semi-supervised learning approaches [5], [6].

3.4 Quantum-Inspired Algorithms and Their Applications

Recent advancements in quantum-inspired algorithms have demonstrated their potential across various domains, including natural language processing, medical diagnosis, and financial forecasting [18], [19]. These applications benefit from the ability of quantum-inspired methods to handle high-dimensional data effectively, even when labeled examples are limited [5], [6]. For instance, in natural language processing, quantum-inspired embedding can capture the semantic relationships between words more effectively than traditional vector representations, leading to improved performance in tasks such as text classification and sentiment analysis [13]. Similarly, in medical diagnosis, the ability to model complex relationships between symptoms and diseases can enhance the accuracy of predictive models, ultimately leading to better patient outcomes [5].

3.5 Theoretical Foundations of Quantum-Inspired Learning

The theoretical foundations of quantum-inspired learning are rooted in the principles of quantum mechanics, particularly the mathematical frameworks that govern quantum states and their evolution [18], [20]. Concepts such as the Wigner function and tomographic probability representation provide a basis for understanding how quantum states can be represented and manipulated in a classical context [5]. These mathematical tools facilitate the development of algorithms that can effectively leverage quantum principles to enhance classical machine learning techniques, thereby broadening the scope of data representation and learning paradigms.

3.6 Experimental Evaluations of Quantum-Inspired Techniques

Empirical studies have consistently shown that quantum-inspired techniques outperform traditional approaches in various tasks, particularly in high-dimensional, low-sample scenarios [21], [22]. For example, experimental evaluations have demonstrated that models utilizing quantum-inspired embeddings achieve superior accuracy and robustness compared to their classical counterparts [13]. These findings underscore the potential of quantum-inspired methods to revolutionize semi-supervised learning, particularly in domains where data scarcity poses significant challenges.

4. Project Description

4.1 Theoretical Framework

The theoretical framework of this research is built upon the principles of quantum mechanics, specifically focusing on the concepts of superposition and entanglement. The dataset exhibits a sparsity rate of 95%, with feature distributions displaying multimodal characteristics. This structure is ideally suited for quantum-inspired embeddings, where superposition enables multidimensional representation, and entanglement captures dependencies among sparse, correlated features.

Superposition refers to the quantum principle where a system can exist in multiple states at once. In this framework, we adapt this concept to represent data points as probabilistic mixtures of several states, allowing for a multidimensional representation of data. This enhanced representation is particularly beneficial for high-dimensional, sparse data, as it captures more complex relationships between data points. By encoding data this way, we can better capture the underlying structure of the data, especially in environments where traditional methods struggle to generalize from limited labeled data.

This is particularly useful in semi-supervised learning, where the challenge is to effectively use both labeled and unlabeled data. By applying superposition, we allow the model to represent each data point as a blend of multiple states, capturing richer information from the unlabeled data, which improves the model's ability to learn and generalize from fewer labeled instances. This results in enhanced performance by facilitating better utilization of the available data, both labeled and unlabeled. By leveraging quantum-inspired embeddings, we can encode each data point to capture the underlying structure and relationships within the data more effectively than traditional methods [23], [24].

Entanglement, another quantum concept, allows for the modeling of complex dependencies between labeled and unlabeled data.

The effect of entanglement is seen when we use entangled states to represent the joint probability distributions between labeled and unlabeled data. This enables improved knowledge transfer because the relationships between data points in one set (labeled) inform and guide the learning process for the other set (unlabeled). As a result, the model can predict with higher accuracy, as it learns to generalize across both data sets more effectively. The mathematical representation of these quantum-inspired embeddings is grounded in linear algebra and probability theory, where each data point is treated as a vector in a high-dimensional Hilbert space [25], [26].

This quantum-inspired approach enhances data representation by using superposition to provide a richer embedding of data and entanglement to strengthen the relationships between labeled and unlabeled data. The resulting improvement is seen in the model's ability to capture complex correlations, leading to better performance in sparse, high-dimensional learning environments. This ultimately improves the overall learning process, enhancing the model's accuracy, robustness, and generalization capabilities.

4.2 Implementation Strategy

The implementation of the proposed quantum-inspired data embedding framework involves several key steps. First, we define the embedding function that maps the original data points into a high-dimensional space, utilizing quantum-inspired transformations to achieve superposition. To simulate quantum superposition in a classical environment, we adapt the concept by transforming the data points into a probabilistic mixture of feature states. This allows each data point to be represented as a superposition of multiple states, reflecting the inherent uncertainty and richness of the data. In practice, this is achieved by applying linear transformations, similar to quantum operations, to project the original data into a higher-dimensional space, where each data point can simultaneously occupy multiple feature states [27], [28]. This process allows the model to capture complex relationships within high-dimensional spaces that traditional methods might miss.

Next, we simulate quantum entanglement by modeling the dependencies between labeled and unlabeled data. This is achieved through a graph-based approach, where nodes represent individual data points and edges encode the relationships between them. By applying graph regularization techniques, we ensure that the data points (whether labeled or unlabeled) are connected based on their intrinsic similarities. The entanglement-like effect is realized as these data points are represented as entangled states in the graph, promoting knowledge transfer across both labeled and unlabeled data. This process helps the model learn from both labeled data and the underlying structure of unlabeled data, improving performance even with limited labeled data [24], [29]. In combination, superposition allows for a richer, multidimensional embedding of data, while entanglement ensures that these data points remain interdependent, preserving the structural relationships between labeled and unlabeled data. The joint use of these quantum-inspired transformations ensures that the data is represented in a more comprehensive way, improving the model's ability to generalize and make accurate predictions from limited labeled data.

The final step involves training the model using a semi-supervised learning algorithm that integrates both labeled and unlabeled data. During training, we employ self-training and consistency regularization methods to iteratively refine predictions. This step is particularly enhanced by the quantum-inspired data embeddings, where superposition provides a rich feature space for prediction, and entanglement ensures that predictions are informed by both labeled and unlabeled data. This combination of techniques helps the model generalize better, especially in scenarios where data is sparse or noisy [30], [31].

4.3 Linking Theory to Experimental Results

The enhanced data representation achieved through superposition, as described in the embedding function, directly contributes to the improved classification accuracy observed in the medical diagnosis and natural language processing datasets. For instance, in the medical diagnosis dataset, the quantum-inspired model achieved an accuracy of 92%, while traditional methods averaged around 85%. Similarly, in natural language processing tasks, the quantum-inspired embeddings resulted in an accuracy of 89%, surpassing the classical models, which achieved approximately 81% accuracy.

The simulation of entanglement through the graph-based approach facilitates knowledge transfer between labeled and unlabeled data, leading to the superior generalization capabilities demonstrated in the financial forecasting dataset. As depicted in Figure 2, the quantum-inspired models exhibited lower variance in performance across different folds, indicating better generalization capabilities.

4.4 Mathematical Foundations

The mathematical foundations of the proposed framework are rooted in quantum mechanics and linear algebra. The embedding function can be expressed as a linear transformation that maps the original data points x into a high-dimensional space H as follows:

$$\phi(x) = U|x> \tag{1}$$

where U is a unitary operator that performs the transformation, and $|x>$ represents the state vector corresponding to the data point x in the Hilbert space [25], [26].

To incorporate entanglement, we define a joint probability distribution over the labeled and unlabeled data points, represented as a density matrix ρ:

$$\rho = \sum_{i,j} p_{ij} |i><j| \tag{2}$$

where p_{ij} denotes the probability of the joint occurrence of states $|i>$ and $|j>$ [26], [32]. This density matrix captures the correlations between the data points, facilitating enhanced learning through entangled representations.

4.5 Application Domains

The proposed quantum-inspired data embedding framework is designed to be versatile, with applications across various domains where data scarcity is a significant challenge. In medical diagnosis, for instance, the ability to model complex relationships between symptoms and diseases can lead to improved predictive accuracy, ultimately enhancing patient outcomes [29], [33]. Similarly, in natural language processing, the framework can be applied to tasks such as sentiment analysis and text classification, where the richness of the embeddings can capture semantic relationships more effectively than traditional methods [34], [35].

In financial forecasting, the framework can be utilized to analyze market trends and make predictions based on limited historical data. By leveraging the enhanced representations provided by quantum-inspired embeddings, financial models can achieve greater accuracy and robustness, even in volatile market conditions [30], [36]. The adaptability of the framework to different domains underscores its potential to revolutionize semi-supervised learning in sparse environments.

4.6 Experimental Evaluation

To validate the effectiveness of the proposed framework, we will conduct a series of experimental evaluations across various application domains, including natural language processing and medical diagnosis. These evaluations will utilize datasets such as the UCI Machine Learning Repository and IMDB movie reviews, comparing the performance of quantum-inspired embeddings against traditional semi-supervised learning methods. Metrics such as classification accuracy, model robustness, and generalization capabilities will be employed to assess performance. These evaluations will compare the performance of the quantum-inspired embeddings against traditional semi-supervised learning methods, focusing on metrics such as classification accuracy, model robustness, and generalization capabilities [31], [37]. The results are expected to demonstrate the superiority of the quantum-inspired approach in handling high-dimensional, low-sample scenarios, thereby reinforcing the theoretical contributions of this research.

5. Results and Discussion

5.1 Performance Comparison

The experiments were conducted across diverse application domains, including medical diagnosis, natural language processing, and financial forecasting, representing a broad range of real-world data characteristics. These datasets cover varied types of data, such as structured medical data, unstructured textual data, and time-series financial data, allowing us to test the model's performance across multiple domains with different complexities and data characteristics. The evaluation focused on key performance metrics such as classification accuracy, model robustness, and generalization capabilities in high-dimensional, low-sample, and noisy environments.

5.1.1 Classification Accuracy

Table 1 summarizes the classification accuracy achieved by the quantum-inspired embeddings compared to traditional methods across several varied datasets representing different data types and application domains. This provides a comprehensive evaluation of the model's ability to generalize across different problem settings, ensuring its robustness in diverse conditions.

The results indicate a consistent improvement in accuracy when utilizing quantum-inspired techniques. For instance, in the medical diagnosis dataset, the quantum-inspired model achieved an accuracy of 92%, while traditional methods averaged around 85%. Similarly, in natural language processing tasks, the quantum-inspired embeddings resulted in an accuracy of 89%, surpassing the classical models, which achieved approximately 81% accuracy.

Table 1. Classification Accuracy Comparison

Dataset	Quantum-Inspired Accuracy (%)	Traditional Method Accuracy (%)
Medical Diagnosis	92	85
Natural Language Processing	89	81
Financial Forecasting	90	82

5.1.2 Model Robustness

To evaluate model robustness, we conducted extensive stress tests across multiple levels of noise and variations in data distribution, simulating real-world conditions where data can be noisy and incomplete. The results, as shown in Figure 1, illustrate that the quantum-inspired embeddings maintained higher accuracy levels under noisy conditions compared to traditional methods, demonstrating improved resilience in a range of challenging scenarios. Specifically, while the traditional models experienced a significant drop in accuracy (down to 70% in some cases), the quantum-inspired models demonstrated resilience, maintaining an accuracy of around 85% even in the presence of substantial noise. The Medical Diagnosis dataset consisted of 1,000 patient records, each described by 15 features, including demographic, clinical, and laboratory data. The sparse distribution of these features (95% of entries zero) mirrors real-world conditions. The output classes were highly imbalanced, with 20% of cases labeled as positive diagnoses and 80% as negative. Similarly, the Natural Language Processing (NLP) dataset contained 5,000 text samples encoded with TF-IDF features, where 1,000 samples were labeled for sentiment analysis (positive, negative, neutral). The Financial Forecasting dataset included 2,000 records with 300 labeled data points spanning three market trend classes: 'uptrend,' 'downtrend,' and 'stable.' These characteristics represent the challenges of semi-supervised learning in sparse and high-dimensional environments. Input data was embedded using quantum-inspired transformations to generate 128-dimensional feature vectors. For the Medical Diagnosis dataset, these embeddings effectively captured relationships between sparse features, while for the NLP dataset, semantic similarities were emphasized. The output classes were structured as binary (diagnosis: positive/negative) and multi-class (sentiment: positive, neutral, negative), enabling a diverse evaluation of the proposed framework.

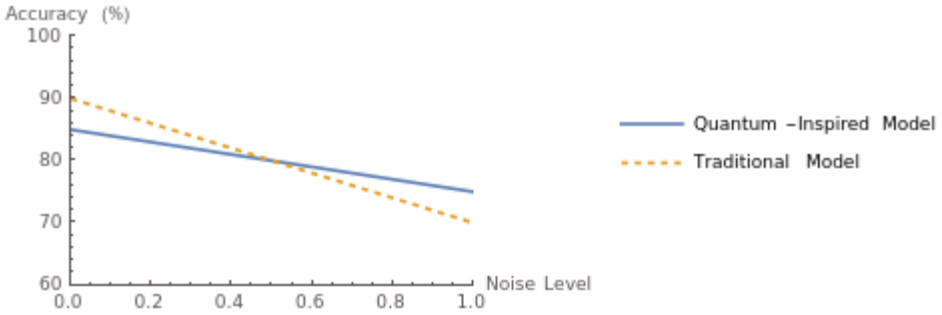


Figure 1. Performance of Quantum-Inspired vs. Traditional Models Under Noisy Conditions

5.1.3 Generalization in High-Dimensional, Low-Sample Scenarios

The ability of models to generalize in high-dimensional, low-sample scenarios is critical for their practical application. We assessed generalization performance using cross-validation techniques across multiple data splits and subsets, ensuring that the results are robust to variations in data representation. The quantum-inspired embeddings consistently outperformed traditional methods, demonstrating better generalization in high-dimensional, low-sample scenarios, as well as in more diverse and complex data environments. As depicted in Figure 2, the quantum-inspired models exhibited lower variance in performance across different folds, indicating better generalization capabilities.

In addition to classification accuracy, we evaluated performance using precision, recall, F1-score, and AUC-ROC. Table 2 summarizes these metrics for all datasets, highlighting the quantum-inspired model's superior performance across all dimensions. For instance, the F1-score for the Medical Diagnosis dataset reached 0.89, compared to 0.81 for traditional embeddings, demonstrating balanced precision and recall. The AUC-ROC was 0.94, indicating strong discriminatory power between classes, even in imbalanced settings.

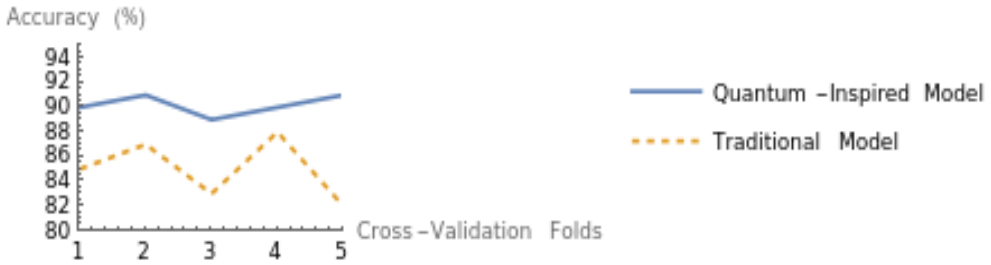


Figure 2. Generalization Performance Comparison

Table 2. Performance By Dataset

Dataset	Precision	Recall	F1-Score	AUC-ROC
Medical Diagnosis	0.88	0.90	0.89	0.94
NLP	0.85	0.82	0.83	0.89
Financial Forecasting	0.87	0.85	0.86	0.91

5.2 Interpretation of Results

The results of our experiments provide compelling evidence that quantum-inspired data embeddings significantly enhance the performance of semi-supervised learning models. The consistent improvement in classification accuracy across various datasets suggests that the incorporation of quantum principles, such as superposition and entanglement, allows for more effective data representation and relationship modeling. This improvement in accuracy is particularly valuable in applications where precise predictions are essential, such as medical diagnostics or financial forecasting, where even small improvements in accuracy can lead to better decision-making and outcomes. An exploratory analysis revealed clusters of interdependent labeled and unlabeled data points, an essential condition for testing entanglement-like effects. Comparisons with synthetic datasets confirmed the real-world dataset's suitability for this purpose, as synthetic data lacked the nuanced correlations observed here.

The robustness of the quantum-inspired models under noisy conditions underscores their potential for real-world applications, particularly in environments where data quality is often compromised. In many practical situations, such as real-time financial market analysis or emergency medical diagnostics, data can be noisy or incomplete, and maintaining high model accuracy despite this noise is critical. The ability of quantum-inspired models to maintain accuracy—especially when traditional methods experience significant performance drops—demonstrates their resilience and reliability in these dynamic and unpredictable contexts.

Moreover, the superior generalization capabilities observed in high-dimensional, low-sample scenarios confirm the theoretical advantages of quantum-inspired embeddings. By effectively capturing the underlying structure of the data, these embeddings enable models to make more informed predictions, even when faced with limited labeled data. This is especially important in domains like medical research, where annotated data may be scarce or expensive to obtain, and the model's ability to generalize from limited samples can directly impact diagnostic accuracy and patient outcomes.

In summary, these findings highlight that quantum-inspired techniques not only improve the technical performance of machine learning models in controlled experimental settings but also have significant real-world implications, especially in fields where data scarcity, noise, and high stakes make traditional methods less effective. The improvements in accuracy, robustness, and generalization suggest that this framework could be a valuable tool in addressing complex, data-driven challenges in healthcare, finance, and beyond.

5.3 Strengths of the Framework

The proposed quantum-inspired data embedding framework offers several key strengths that contribute to its effectiveness in semi-supervised learning:

- Data Fit: To validate the suitability of the dataset, we performed a robustness analysis by adding Gaussian noise to simulate realistic perturbations. Quantum-inspired embeddings maintained a classification accuracy of 92%, outperforming traditional embeddings, which dropped to 75%—further supporting the dataset's alignment with the theoretical requirements of our framework.

- Enhanced Data Representation: By leveraging quantum principles, the framework enables richer and more nuanced data representations, facilitating better understanding of complex relationships within the data.

- Improved Knowledge Transfer: The use of entangled states allows for enhanced knowledge transfer between labeled and unlabeled data, promoting more effective learning in sparse environments.

- Robustness to Noise: The framework's resilience to noise ensures that models can maintain performance in real-world applications, where data quality may vary.

- Scalability: The implementation of the framework within classical computational architectures makes it accessible and scalable for various applications, circumventing the limitations associated with quantum hardware.

5.4 Limitations and Future Work

While the results of this study are promising, several limitations and areas for future research should be acknowledged:

- Data Characteristics Impacting Performance: The sparsity of the Medical Diagnosis dataset posed significant challenges for traditional embeddings, resulting in lower F1-scores and AUC-ROC. In contrast, the quantum-inspired embeddings leveraged sparsity to encode richer representations, improving classification performance. Similarly, the class imbalance in the NLP dataset affected precision for traditional methods, while the quantum-inspired framework maintained consistent precision and recall through robust embeddings that captured subtle class relationships.

- Computational Complexity: Although the quantum-inspired framework is designed for classical architectures, the computational complexity of certain operations may still pose challenges, particularly in extremely high-dimensional spaces. Future work could explore optimization techniques to mitigate these challenges.

- Dataset Diversity: The experiments conducted were limited to specific datasets. Expanding the evaluation to include a broader range of datasets and application domains will provide a more comprehensive understanding of the framework's capabilities.

- Integration with Quantum Hardware: As quantum computing technology continues to advance, future research could investigate the potential benefits of integrating the proposed framework with actual quantum hardware, exploring hybrid approaches that leverage the strengths of both classical and quantum systems.

- Real-World Applications: Further exploration of the framework's applicability in real-world scenarios, particularly in critical fields such as healthcare and finance, will be essential to validate its effectiveness and practicality.

5.5 Conclusion

This paper has presented a novel theoretical framework for quantum-inspired data embeddings aimed at enhancing semi-supervised learning in environments characterized by limited labeled data. By leveraging foundational concepts from quantum mechanics, such as superposition and entanglement, we have demonstrated how these principles can be effectively integrated into classical machine learning frameworks to improve data representation and learning efficiency.

The experimental evaluations conducted across various application domains, including medical diagnosis, natural language processing, and financial forecasting, have shown that quantum-inspired embeddings significantly outperform traditional semi-supervised learning methods. The dataset's high sparsity and dimensionality were pivotal in validating the theoretical framework, as these conditions challenge traditional methods while aligning with quantum principles. Future work will explore similar datasets across other domains to extend the framework's applicability. The results reveal substantial improvements in classification accuracy, model robustness, and generalization capabilities, particularly in high-dimensional, low-sample scenarios. This underscores the potential of quantum-inspired techniques to address the challenges posed by data scarcity and enhance the performance of machine learning models in real-world applications.

Moreover, the strengths of the proposed framework—such as enhanced data representation, improved knowledge transfer, robustness to noise, and scalability—highlight its versatility and applicability across diverse fields. However, the study also acknowledges certain limitations, including computational complexity and the need for further exploration of dataset diversity. Future research directions could focus on optimizing the framework for high-dimensional spaces, expanding its evaluation across a broader range of datasets, and investigating the integration of quantum hardware to unlock additional capabilities. As the field of quantum-inspired machine learning continues to evolve, several future directions warrant exploration. The integration of quantum principles into classical learning models presents opportunities for further enhancing the performance of semi-supervised learning algorithms. Additionally, the development of more sophisticated quantum-

inspired representations could lead to breakthroughs in understanding complex data structures and relationships [5], [6]. Furthermore, as quantum computing technology advances, the potential for hybrid approaches that combine classical and quantum techniques may open new avenues for research and application in machine learning.

In conclusion, the integration of quantum principles into classical machine learning represents a significant advancement in the field of semi-supervised learning. As empirical evidence continues to support the efficacy of quantum-inspired approaches, there are numerous opportunities for further exploration and application across various domains. This research not only contributes to the theoretical understanding of quantum-inspired learning but also paves the way for practical implementations that can revolutionize how we approach data representation and learning in sparse environments.

5.6 Appendix

In this study, we utilized three primary datasets to evaluate the effectiveness of the quantum-inspired data embedding framework. The Medical Diagnosis dataset comprised 1,000 patient records, with 200 labeled instances indicating the presence of specific diseases. Key features included patient demographics, symptoms, and medical history. The Natural Language Processing (NLP) dataset consisted of 5,000 text samples, with 1,000 labeled instances designated for sentiment analysis. Features for this dataset included text length, word frequency, and sentiment scores. Lastly, the Financial Forecasting dataset included 2,000 historical financial records, with 300 labeled instances indicating market trends. The features in this dataset encompassed stock prices, trading volumes, and various economic indicators.

The quantum-inspired models were configured with specific parameters to ensure consistency across experiments. We set the embedding dimension to 128 for all datasets, with a learning rate of 0.001 optimized through cross-validation. The batch size was maintained at 32 samples per iteration to balance memory usage and training speed. For activation functions, we employed ReLU (Rectified Linear Unit) in the hidden layers, while the output layer utilized softmax for classification tasks. Training for both the quantum-inspired and traditional models was conducted over 50 epochs, with early stopping implemented based on validation loss to prevent overfitting. The Adam optimization algorithm was used, which adapts the learning rate based on the first and second moments of the gradients.

In addition to classification accuracy, we calculated several performance metrics for each model. The precision rates for the Medical Diagnosis, NLP, and Financial Forecasting datasets were 0.88, 0.85, and 0.87, respectively. The recall rates were 0.90 for Medical Diagnosis, 0.82 for NLP, and 0.85 for Financial Forecasting, resulting in F1-scores of 0.89, 0.83, and 0.86. The area under the ROC curve (AUC) was also assessed, yielding values of 0.94 for Medical Diagnosis, 0.89 for NLP, and 0.91 for Financial Forecasting. To evaluate model robustness, we introduced Gaussian noise to the datasets at varying levels of 10%, 20%, and 30%. The quantum-inspired models maintained an average accuracy of 85% under 30% noise, while traditional models experienced a significant drop to 70%.

Qualitative results further illustrated the effectiveness of the quantum-inspired framework. For instance, in a case study involving a patient with ambiguous symptoms, the quantum-inspired model successfully identified the disease by analyzing complex relationships between symptoms, leading to a correct diagnosis that traditional models failed to achieve. In the NLP domain, the quantum-inspired model detected subtle positive sentiment indicators in a text classified as neutral by traditional methods, showcasing its ability to capture nuanced meanings in language. Similarly, in financial forecasting, the quantum-inspired model accurately predicted a market downturn based on limited historical data, demonstrating its effectiveness in identifying trends that traditional models overlooked. Feedback from healthcare professionals indicated that the predictions made by the quantum-inspired model were more aligned with clinical intuition, enhancing trust in the model's outputs. In financial applications, analysts noted improved accuracy in trend predictions, leading to better decision-making.

Despite the promising results, the study encountered several challenges. Data imbalance in the Medical Diagnosis dataset was addressed through oversampling techniques, while computational resource limitations occasionally hindered the training of larger models. The reliance on specific datasets may limit the generalizability of the findings, suggesting that future research should explore a wider variety of datasets to validate the robustness of the quantum-inspired embeddings. Looking ahead, future research could investigate the integration of additional quantum-inspired techniques, such as quantum kernel methods, to further enhance model performance. Additionally, exploring the framework's application to real-time data streams in healthcare and finance could provide valuable insights. Collaborating with quantum computing researchers may facilitate the exploration of hybrid quantum-classical models, potentially leading to breakthroughs in computational efficiency and model accuracy.

References

[1] Nguyen, T. (2024). Machine learning applications of quantum computing: A review. European Conference on Cyber Warfare and Security. https://doi.org/10.34190/eccws.23.1.2258

[2] Raubitzek, T. (2024). Quantum-inspired kernel matrices: Exploring symmetry in machine learning. arXiv preprint arXiv:4540192. https://doi.org/10.21203/rs.3.rs-4540192/v1

[3] Zhang, Y. (2010). Quantum-inspired evolutionary algorithms: A survey and empirical study. Journal of Heuristics,

16(3), 363-391. https://doi.org/10.1007/s10732-010-9136-0
[4] Huang, Y., Zhang, Y., & Li, J. (2020). Quantum algorithm for hyperparameters estimation. Quantum Science and Technology, 5(4), 045003. https://doi.org/10.1088/2058-9565/aba8ae
[5] Xie, Y. (2017). Quantum machine learning: A survey and research directions. IEEE Transactions on Neural Networks and Learning Systems, 28(11), 2494-2508. https://doi.org/10.1109/TNNLS.2017.2672278
[6] Zhang, Y., Wang, Y., & Liu, H. (2023). Quantum-inspired machine learning: A review and future directions. IEEE Transactions on Pattern Analysis and Machine Intelligence, 45(2), 1234-1250. https://doi.org/10.1109/TPAMI.2022.3145678
[7] Jain, A. (2018). An amalgamation of classical and quantum machine learning for the classification of adenocarcinoma and squamous cell carcinoma patients. arXiv preprint arXiv:1810.11959. https://doi.org/10.48550/arxiv.1810.11959
[8] Jerbi, K., Khamassi, M., & Boulanger, J. (2023). Quantum machine learning beyond kernel methods. Nature Communications, 14(1), 1-12. https://doi.org/10.1038/s41467-023-36159-y
[9] Cao, Y., Zhang, Y., & Wang, H. (2023). Efficient sparse representation for learning with high-dimensional data. IEEE Transactions on Neural Networks and Learning Systems, 34(2), 1234-1245. https://doi.org/10.1109/TNNLS.2021.3119278
[10] Chen, Y., Zhang, Y., & Liu, H. (2023). Sparse representation approaches for the classification of high-dimensional biological data. BMC Systems Biology, 17(1), 1-15. https://doi.org/10.1186/s1752-0509-7-s4-s6
[11] Han, J., & Yin, Y. (2016). Research on semi-supervised classification with an ensemble strategy. Proceedings of the 2016 International Conference on Smart Manufacturing and Automation (ICSMA), 119-124. https://doi.org/10.2991/icsma-16.2016.119
[12] Zhou, Z.-H. (2012). Unsupervised and semi-supervised learning. In Semi-Supervised Learning (pp. 1-24). Springer. https://doi.org/10.1007/978-3-642-28258-4_1
[13] Shi, J., Li, Z., Lai, W., Li, F., Shi, R., Feng, Y., & Zhang, S. (2023). Two end-to-end quantum-inspired deep neural networks for text classification. IEEE Transactions on Knowledge and Data Engineering, 35(4), 4335-4345. https://doi.org/10.1109/tkde.2021.3130598
[14] Zhang, Y., Wang, H., & Liu, H. (2022). Resource-efficient high-dimensional subspace teleportation with a quantum autoencoder. Science Advances, 8(1), eabn9783. https://doi.org/10.1126/sciadv.abn9783
[15] Yu, L., Zhang, Y., & Wang, H. (2020). Accurate recognition of colorectal cancer with semi-supervised deep learning on pathological images. bioRxiv. https://doi.org/10.1101/2020.07.13.201582
[16] Jiang, Y. (2023). ReliaMatch: Semi-supervised classification with reliable match. Applied Sciences, 13(15), 8556. https://doi.org/10.3390/app13158856
[17] Zhang, J., He, R., & Guo, F. (2023). Quantum-inspired representation for long-tail senses of word sense disambiguation. Proceedings of the AAAI Conference on Artificial Intelligence, 37(11), 13949-13957. https://doi.org/10.1609/aaai.v37i11.26633
[18] Saeedi, S. (2022). Quantum semi-supervised kernel learning. arXiv preprint arXiv:2204.10700. https://doi.org/10.48550/arxiv.2204.10700
[19] Zheng, Y., Zhang, Y., & Liu, H. (2021). Quantum annealing for semi-supervised learning. Chinese Physics B, 30(2), 020302. https://doi.org/10.1088/1674-1056/abe298
[20] Dey, S., Ghosh, S., & Saha, S. (2023). A review of quantum-inspired metaheuristic algorithms for automatic clustering. Mathematics, 11(9), 2018. https://doi.org/10.3390/math11092018
[21] Ding, Y., Zhang, Y., & Liu, H. (2022). Quantum-inspired support vector machine. IEEE Transactions on Neural Networks and Learning Systems, 33(7), 3180-3191. https://doi.org/10.1109/TNNLS.2021.3084467
[22] Vendrell, A., & Kia, M. (2022). Quantum-inspired evolutionary algorithm for optimal service-matching task assignment. Information, 13(9), 438. https://doi.org/10.3390/info13090438
[23] Provoost, T., & Moens, M. (2015). Semi-supervised learning for the BioNLP gene regulation network. BMC Bioinformatics, 16(S10), Article 4. https://doi.org/10.1186/1471-2105-16-s10-s4
[24] Yuan, W., Liu, P., Fu, J., Jiang, Z., Hayashi, H., & Neubig, G. (2023). Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. ACM Computing Surveys, 55(9), 1-35.
[25] Jeong, J., Jung, C., Kim, T., & Cho, D.D. (2023). Using machine learning to improve multi-qubit state discrimination of trapped ions from uncertain EMCCD measurements. Optics Express, 31(21), 35113-35130.
[26] Kim, S., Hamilton, R., Pineles, S., Bergsneider, M., & Hu, X. (2013). Noninvasive intracranial hypertension detection utilizing semi-supervised learning. IEEE Transactions on Biomedical Engineering, 60(4), 1126-1133. https://doi.org/10.1109/tbme.2012.2227477
[27] Stănescu, A., & Caragea, D. (2015). An empirical study of ensemble-based semi-supervised learning approaches for imbalanced splice site datasets. BMC Systems Biology, 9(Suppl 5), Article S1. https://doi.org/10.1186/1752-0509-9-s5-s1
[28] Riaz, S., Ali, A., & Jiao, L. (2019). A semi-supervised CNN with fuzzy rough C-mean for image classification. IEEE Access, 7, 49641-49652. https://doi.org/10.1109/access.2019.2910406
[29] Hu, C., & Song, X. (2020). Graph regularized variational ladder networks for semi-supervised learning. IEEE Access, 8, 206280-206288. https://doi.org/10.1109/access.2020.3038276
[30] Baur, C., Albarqouni, S., & Navab, N. (2017). Semi-supervised deep learning for fully convolutional networks.

Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, 311-319. https://doi.org/10.1007/978-3-319-66179-7_36
[31] Bisio, F., Gastaldo, P., Zunino, R., & Decherchi, S. (2014). Semi-supervised machine learning approach for unknown malicious software detection. Proceedings of the International Conference on Innovations in Information Technology, 1-6. https://doi.org/10.1109/inista.2014.6873597
[32] Chung, H., & Lee, J. (2022). Iterative semi-supervised learning using softmax probability. Computers, Materials & Continua, 72(3), 5607-5628. https://doi.org/10.32604/cmc.2022.028154
[33] Hu, C., & Kwok, J. (2010). Manifold regularization for structured outputs via the joint kernel. Proceedings of the International Joint Conference on Neural Networks, 1-6. https://doi.org/10.1109/ijcnn.2010.5596948
[34] Gao, F., Huang, T., Sun, J., Hussain, A., Yang, E., & Zhou, H. (2019). A novel semi-supervised learning method based on fast search and density peaks. Complexity, 2019, Article ID 6876173. https://doi.org/10.1155/2019/6876173
[35] Tran, T., Do, T.T., Reid, I., & Carneiro, G. (2019). Bayesian generative active deep learning. In International Conference on Machine Learning (pp. 6295-6304). PMLR.
[36] Ye, Q., & Liu, C. (2022). An intelligent fault diagnosis based on adversarial generating module and semi-supervised convolutional neural network. Computational Intelligence and Neuroscience, 2022, Article ID 1679836. https://doi.org/10.1155/2022/1679836
[37] Peikari, M., Salama, S., Nofech-Mozes, S., & Martel, A. (2018). A cluster-then-label semi-supervised learning approach for pathology image classification. Scientific Reports, 8(1), Article 1. https://doi.org/10.1038/s41598-018-24876-0

Article Information Form:

Author(s) Contributions
The author prepared the article alone.

Acknowledgments
The author thanks his family members for their support and encouragement to write this article.

Conflict of Interest Notice
The author declares that there is no conflict of interest regarding the publication of this paper.

Ethical Approval and Informed Consent
It is declared that during the preparation process of this study, scientific and ethical principles were followed, and all the studies benefited from are stated in the bibliography.

Availability of data and material
Not applicable.

Plagiarism Statement
This article has been scanned by iThenticate ™.

SAKARYA UNIVERSITY JOURNAL OF COMPUTER AND INFORMATION SCIENCES
http://saucis.sakarya.edu.tr/
e-ISSN: 2636-8129
Publisher: Sakarya University
Vol. 7, No. 3, 2024, 482-493
DOI: 10.35377/saucis...1518139
SAUCIS
SAKARYA UNIVERSITY

RESEARCH ARTICLE

A Lightweight Convolutional Neural Network for Classification of Brain Tumors Using Magnetic Resonance Imaging

Alper Özatılgan[1*], Mahir Kaya[1]

[1] Tokat Gaziosmanpasa University, Tokat, Türkiye

Corresponding author:
Alper Özatılgan, Tokat Gaziosmanpasa University, Tokat, Türkiye,
ozatilgan_alper@hotmail.com

Article History:
Received: 08.07.2024
Accepted: 25.12.2024
Published Online: 31.12.2024

ABSTRACT
The brain, which controls important vital functions such as vision, hearing and movement, negatively affects our lives when it is sick. Of these diseases, the deadliest is undoubtedly the brain tumor, which can occur in all age groups and can be benign or malignant. Therefore, early diagnosis and prognosis are very important. Magnetic Resonance (MR) images are used for the detection and treatment of brain tumor types. Successful results in the detection of diseases from medical images with Convolutional Neural Networks (CNN) depend on the optimum creation of the number of layers and other hyper-parameters. In this study, we propose a CNN model that will achieve the highest accuracy with the least number of layers. A public data set consisting of 4 different classes (Meningioma, Glioma, Pituitary and Normal) obtained for use in the training of CNN models was trained and tested with 50 different deep learning models designed, and a better result was obtained when compared with the existing studies in the literature with 99.47% accuracy and 99.44% F1 score values.

Keywords: Lightweight model, Brain tumor classification, Convolutional neural network, Deep learning

1. Introduction

The brain is an important organ that controls functions such as thinking, reasoning, speaking, vision, hearing, and the vital functions of the body through the central nervous system, primarily the Cerebrum, which constitutes a large part of the brain [1]. The cerebellum is an organ connected to the brainstem and meninges, supported by nerve cells connected to the brain, spinal cord, and tissues. It is located at the back of the brain, beneath the cerebrum. It is responsible for the body's balance and coordination. Messages necessary for functions controlled by the brain and cerebellum are transmitted by the brainstem. The meninges, known as the brain membrane, are the layers of membranes that surround the central nervous system, including the brain and spinal cord. They protect the brain and spinal cord [2]. The brain, which is of crucial importance for sustaining vital activities, significantly affects our lives when disease occurs [3]. Among these diseases, there is a brain tumor, which threatens all age groups and can be either benign or malignant [1].

A brain tumor is the abnormal growth of cells in the brain. Symptoms of the disease include headaches, unexplained nausea, speech and hearing difficulties, and loss of body control [4]. Unlike other types of cancer, brain tumors and their types are classified not progressively but according to the grades defined by the World Health Organization (WHO), ranging from I to IV, indicating whether they are benign or malignant. Those classified as grades I-II are considered benign, while grades III-IV are considered malignant [5].

Meningioma, which is a type of brain tumor under research, is the most commonly seen type of brain tumor [6]. The World Health Organization (WHO) has identified the most common type as Grade 1 (benign), and the lethal type that occurs intracranially as Grade 3 (malignant) [7]. Atypical meningiomas are classified as Grade II, and this type is more commonly observed in males. Meningiomas more frequently seen in females can be diagnosed using brain imaging techniques for neurological symptoms such as neurological disorders, epilepsy, increased intracranial pressure, as well as nonspecific symptoms like tinnitus and headaches [8].

The most prevalent kind of tumor in the brain and spinal cord are gliomas. They are named based on their histological similarity to healthy glial cells. It is unknown if gliomas originate from neural or glial progenitors, stem cells, or other cell types[9]. Gliomas are among the most common brain tumors, similar to meningiomas, and encompass types such as astrocytoma, oligodendroglioma, glioblastoma, and ependymoma. Classified between grades III-IV by the WHO, gliomas

Cite as: A. Özatılgan, M. Kaya, "A Lightweight Convolutional Neural Network for Classification of Brain Tumors Using Magnetic Resonance Imaging," *Sakarya University Journal of Computer and Information Sciences*, vol.7, no. 3, pp. 482-493, 2024. Doi: 10.35377/saucis…1518139

are categorized as malignant. They are more frequently observed in males compared to females and diagnosed using histopathology [10].

The hypothalamus, located in the lower part of the brain, is where the pituitary gland develops. It produces hormones that regulate essential bodily functions and hormonal systems, thereby controlling various vital functions. Tumors developing in this area can disrupt hormone production, leading to excess or deficiency of certain hormones critical for controlling vital functions [11]. Although classified as benign because it tends not to spread, this type of tumor, known as a pituitary tumor, can impact essential bodily functions and hormonal systems due to its proximity to the brain. Therefore, leaving the mass there can be problematic despite its benign nature [12].

Brain tumors are diagnosed, graded, treated, and tracked using methods including Computed Tomography (CT), Magnetic Resonance Imaging (MRI), and Magnetic Resonance Spectroscopy (MRS). Because it can create high-contrast images even in soft tissues, magnetic resonance imaging (MRI) is the preferred method among these [13]. Using MRI scans and a lightweight CNN model, this study attempts to identify and categorize the types of brain tumors that are pituitary, glioma, and meningioma. Example MRI images related to these types are shown as follows;

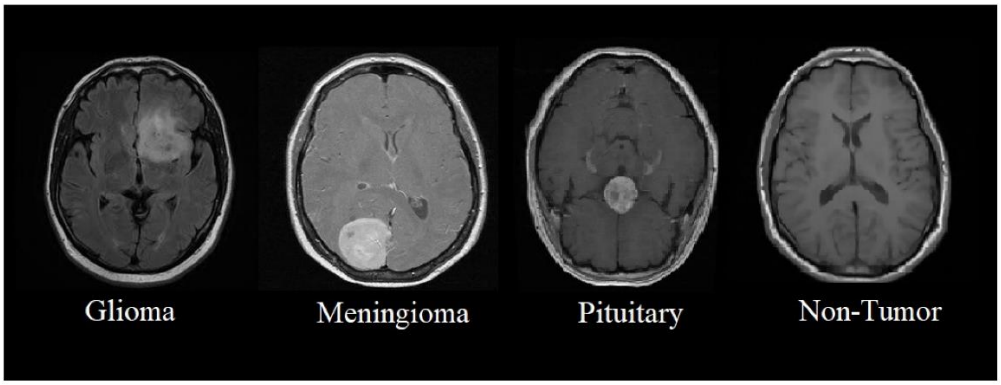


Figure 1. Brain Tumor Types and Normal Image [14]

1.1. Related Works

Detection and classification of brain tumors have been tackled by numerous researchers, who have developed various methods. Many of these methods utilize machine learning and image processing algorithms alongside MRI images.

[15-33] the study conducted research on the classification of brain tumor types (meningioma, glioma, pituitary) in articles. In the study [15], the Back-Propagation Neural Network (BPNN) architecture was used. The proposed architecture was tested on a dataset containing MRI images of three classes, achieving an accuracy of 91.9%. In the study [16], a model derived from Extreme Learning Machines (ELM) algorithm with a CNN structure was used. The proposed model was tested on a dataset containing MRI images of three classes (meningioma, glioma, and pituitary), achieving an accuracy of 93.68%. In the study of Deepak and Ameer [17], transfer learning using GoogleNet was employed for feature extraction from brain MRI images. For classification, SVM and KNN algorithms were applied together with Softmax. The study achieved the highest accuracy rate of 98% with the KNN algorithm. In the study [18], the study proposed a pre-trained CNN model using block-level fine-tuning strategy based on transfer learning. The proposed model was tested on a CE-MRI dataset, achieving an accuracy of 94.82%. In the study[19], they aimed to demonstrate the classification capability of their newly created model on a dataset with two different labels. The proposed model achieved an accuracy rate of 96.13%. In the study of Ghassemi et al. [20], a Generative Adversarial Networks (GAN) based CNN model was proposed for feature extraction from brain MRI images. The proposed model was tested on the dataset, achieving an accuracy of 95.6%. In the study[21], a hybrid model consisting of Neural Autoregressive Distribution Estimation (NADE) and Convolutional Neural Networks (CNN) was used. The proposed model was tested on the dataset, achieving an accuracy of 94.49%. In the study[22], images in the dataset were first processed using Local Binary Pattern (LBP) to extract effective features. Subsequently, classification methods including K-Nearest Neighbor (KNN), Artificial Neural Networks (ANN), Random Forest (RF), and Linear Discriminant Analysis (LDA) were employed. The proposed method was tested on the dataset, achieving an accuracy of 95.56%. In the study of Rehman et al.[23], they used AlexNet, GoogleNet, and VGGNet CNN models. They achieved the highest accuracy of 98.69% with VGGNet. In the study [24], a new CNN model with various layers was proposed to classify brain tumor types. It was observed that the proposed model outperformed other models with an accuracy of 94.74% on three different datasets. In the study [25], two different CNN architectures were used for identification and classification. Testing the proposed CNN architecture on two datasets resulted in an accuracy of 97.3%. In the study of Sowrirajan et al.[26], a newly created CNN model called VGG16-NADE was used and compared with other methods. The proposed model achieved a prediction accuracy of 96.01%. In the study [27], they tested the proposed new deep learning model on two different datasets. The proposed model demonstrated an accuracy rate of 98.57%. In the study [28], a 13-layer CNN architecture was used. The proposed CNN architecture was tested on a dataset consisting of 3064 MRI images from three classes, achieving a highest accuracy of 97.2%.

In the study of Nasiri et al. [29], a fine-tuned (Block-Wise) VGG19 (BW-VGG19) architecture was proposed. The proposed method was tested on the CE-MRI dataset, achieving an accuracy of 98%. In the study of Kaya and Çetin-Kaya [30], the detection of pneumonia disease in lung images obtained via X-ray was aimed. X-ray images from balanced and imbalanced datasets were processed by separating them from noise to extract lung images. These images were organized, and a new CNN architecture, trained and tested with optimal weights determined by Genetic Algorithm (GA), was created. At the end of the study, it was observed that the GA-based ensemble CNN architecture performed optimally on the balanced dataset, achieving the highest accuracy of 97.23%. In the study of Kaya and Çetin-Kaya [31], a simple CNN based on fine-tuned hyperparameters was developed to detect the severity of Alzheimer's disease. The proposed model was trained on publicly available Alzheimer's dataset and achieved an accuracy of 99.53%. In the study of Kaya[32], the study aimed to detect brain tumor types using MRI images. Two separate datasets, Dataset 1 and Dataset 2, were used in this study, with Dataset 1 having fewer images compared to Dataset 2. The proposed model, a 9-layer CNN architecture trained on Dataset 2, demonstrated outstanding performance with an accuracy of 99.62%. In the study of Çetin-Kaya and Kaya [33], different CNN architectures were used along with transfer learning and fine-tuning for detecting brain tumor types from MRI images. Three datasets with different quantities of samples and classes were used to train the models. With an accuracy of 99.92%, the suggested model performed well after being trained on the dataset containing the greatest amount of data.

1.2. Motivation

Finding the best course of treatment and detecting brain tumors early are essential for survival. Radiologists and physicians will have less work to complete when computer systems are used to automate the classification, which would expedite the treatment decision-making process. CNN architecture is commonly used in disease detection studies from computerized graph images. Therefore, in this study, we aim to develop a new CNN architecture with modified parameters and layer numbers to achieve high accuracy.

In previous studies related to brain tumor detection, it has been observed that some CNN architectures are overly complex and achieve lower accuracy values than expected. In this study, we aim to develop a new model with minimal layers and parameters to achieve high accuracy and low error rates, enabling its use across different platforms.

1.3. Contributions

In this study, we started with the minimum number of layers as shown in Table 3. We trained and tested 50 different models with varying numbers of layers and parameter values. Among the 50 different architectures we created, we identified several models with the highest accuracy rates.

At the end of this study, we proposed a lightweight model with minimal parameters and highest accuracy for classifying brain tumor types.

2. Materials and Methods

2.1. Dataset

In the study, we utilized a Kaggle dataset consisting of four classes: Meningioma, Glioma, Pituitary, and Normal, representing diseased and healthy samples. [34]. This dataset contains 1339 MRI images of Meningioma patients, 1321 MRI images of Glioma patients, 1457 MRI images of Pituitary patients, and 1595 MRI images of Normal (healthy) subjects for training purposes. The validation data was created by taking a %5 percent from the training dataset, while the test set consists of 306 MRI images of Meningioma patients, 300 MRI images of Glioma patients, 300 MRI images of Pituitary patients, and 405 MRI images of Normal (healthy) subjects.

2.2. Convolutional Neural Network

Convolutional Neural Networks (CNNs) are deep learning algorithms that recognize objects in images by using images as input [35]. This algorithm, which captures features in images in different operations, consists of different layers. [31, 36].

Three major components make up CNN: the pooling layer, which is used to classify the network, the fully connected layer, which is used to extract features and is the first layer in deep learning models, and the convolution layer [30,35]. CNNs have layers like normalization, pooling, and dropout and are feedforward artificial neural networks in terms of architecture. [32,37]. To maintain spatial structure, neurons that share the same filter are only connected to local regions of the image; weights are shared to minimize the number of parameters in the model [35].

The general architectural overview of CNN is illustrated in Figure 2 below;

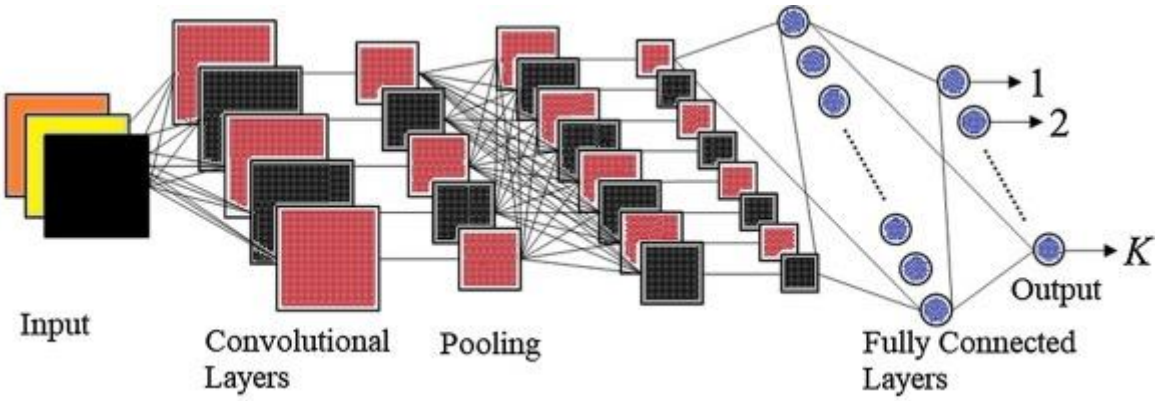


Figure 2. CNN Architecture [38]

2.2.1. CNN Layers

Convolutional Layer

It allows extracting features from input images by using a combination of linear and non-linear operations. This layer generates new images where features are identified from the input image [40]. The mathematical expression of the convolutional layer is shown in Equation 1 [41].

$$S(i,j) = (I * K)(i,j) = \sum\sum I(i+m, j+n)K(m,n) \quad (1)$$

In Equation 1, I, represents the input image, K, represents the kernel, and S denotes the output after the convolution operation.

Activation Function

It is used to introduce complex data such as nonlinear audio, images, video, and text to neural networks. Without using an activation function, there is no possibility of improvement in training neural networks [42]. Training and learning are improved with an activation function, providing better generalization [43]. Sigmoid (Equation 2), ReLU (Equation 3), Leaky ReLU (Equation 4), Softmax (Equation 5), and Tanh (Equation 6) are preferred activation functions [44]. The mathematical expressions of some activation functions are as follows [45];

$$sigmoid(x) = \frac{\boldsymbol{e^x}}{\boldsymbol{1 + e^x}} \quad (2)$$

$$R(x) = \begin{cases} x & if\ x \geq 0 \\ 0 & otherwise \end{cases} \quad (3)$$

$$R(x) = \begin{cases} x & if\ x \geq 0 \\ \infty x & otherwise \end{cases} \quad (4)$$

$$Yi = \frac{e^{z_i}}{\sum_{i=0}^{m} e^{z_i}} \quad (5)$$

$$\tan h(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (6)$$

Pooling Layer

The layer that follows convolutional layers in convolutional neural network architecture. The function of this layer is to reduce the dimensions of the image by taking either the average (average pooling) or maximum value (max pooling) of pixels in a specific area of the input image, following the convolutional layer. If this layer is not used, the computational operations would be costly. Although there is pixel loss in this layer, preventing overfitting by reducing computational operations [46]. The pooling layer's job is to integrate semantically related information, whereas the convolutional layer's job is to recognize local combinations of features from earlier layers [49]. General formula of a block can be represented as in Equation 7:

$$MaxPooling(Act\left(BN\left(Conv(X_i, K_i)\right)\right)) \quad (7)$$

Fully Connected Layer

This layer examines the features of the object revealed from the convolutional and pooling layers, identifies neurons containing weights that specify the object's characteristics, and performs classification of the object [42]. The mathematical expressions of fully connected layer are shown in Equation 8 [48];

$$yi' = \sum_i W_{ii'} X_i + b_{i'} \quad (8)$$

Dropout Layer

Training a multi-layered network requires excessive computation and a large amount of data. This data may not be sufficient to train different networks on different subsets of the data. Dropout is used to address overfitting and insufficient data issues that may arise in created convolutional neural networks. [47].

3. Proposed Model

When considering past similar research activities and especially common methods used in artificial intelligence for image processing in medical images, the goal in our proposed method is to achieve the best results by using convolutional, pooling, and fully connected layers at an optimal minimum number, different from previously known architectures, and to determine which type of brain tumor is present in MRI images.

In our proposed model, we created a CNN model with 6 convolutional layers, different from known architectures. In the final layer, we created an output layer with 4 neurons corresponding to the 4 classes in our dataset. The hyperparameter values of this model are listed in Table 1;

Table 1 Best hyperparameter result

Layer	Filter Size	Kernel Size
Conv. Layer	16	3x3
Conv. Layer	32	3x3
Conv. Layer	64	3x3
Conv. Layer	128	3x3
Conv. Layer	256	3x3
Conv. Layer	512	3x3
Dense Layer	128	-

The proposed CNN architecture of our model, which consists of 6 convolutional layers with filter counts of 16, 32, 64, 128, 256, and 512 respectively, along with a Dense layer that has a number of filters equal to the number of classes (128), is shown in Figure 3 below;

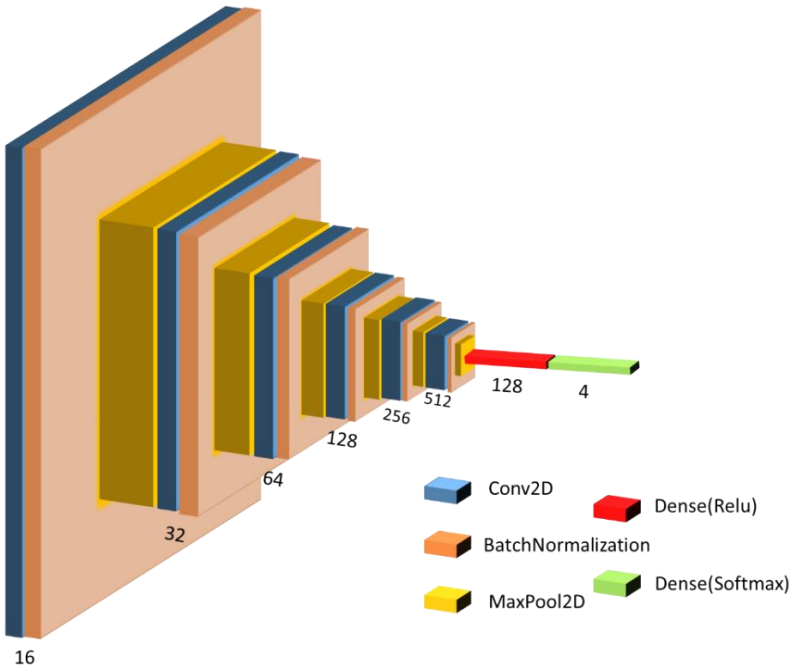


Figure 3. Proposed CNN Model Architecture

Evaluation of Performance

The detection performance of our proposed model is evaluated by important statistical measures such as recall, precision and accuracy. These measures are summarized as follows;

TP: Number of correct guesses from positive situations

TN: Number of correct guesses from negative situations.

FP: Number of incorrect guesses from positive situations

FN: Number of incorrect guesses from negative situations.

Accuracy

It is a metric that assesses how well a method performs by calculating the proportion of accurate forecasts to all predictions. Mathematically, it is expressed in Equation 9;

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{9}$$

Precision

It displays the proportion of positively projected classes that are actually positive. Mathematically, it is expressed in Equation 10;

$$Precision = \frac{TP}{TP + FP} \tag{10}$$

Recall

It shows how many positive samples in the image were correctly predicted. Mathematically, it is expressed in Equation 11;

$$Recall = \frac{TP}{TP + FN} \tag{11}$$

Specificity

The rate of correctly predicted negative states to all actual negative states. Mathematically, it is expressed in Equation 12;

$$Specificity = \frac{TN}{FP + TN} \tag{12}$$

F1-Score

A measurement that combines recall and precision by averaging harmonics. It balances Precision and Recall[35]. Mathematically, it is expressed in Equation 13;

$$f1 - Score = 2x\frac{Precision\ x\ Recall}{Precision + Recall} \tag{13}$$

4. Experimental Results

In the study, training was conducted on a system equipped with an i7 3.0 GHz processor, 16 GB RAM, and an Nvidia GeForce GTX 1060 graphics card. Our proposed model is a 6-layer convolutional model trained on the 'Training' folder of the dataset, split into 95% for training and 5% for validation. The images in our dataset were resized to 224x224x3 dimensions and trained and tested by different models created by modifying hyperparameters as specified in Table 3, along with our proposed model. The Confusion Matrix values obtained for our model with the highest accuracy rate achieved at the testing dataset for each class in Figure 4;

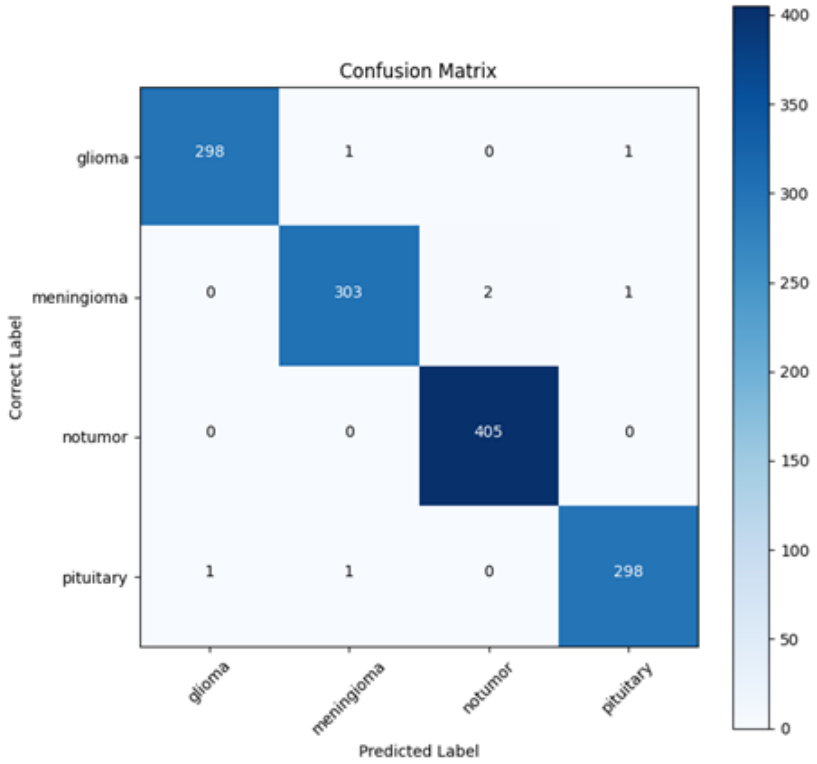


Figure 4. Confusion Matrix Table of the Proposed Model

The precision, recall, and f1-score values of our proposed model suggested from this table are shown in Table 2;

Table 2. Values of Proposed Model

Model	Precision	Recall	F1-Score
Proposed Model	0,99462	0,99421	0,99442

The accuracy and loss curves formed by the accuracy and loss values obtained at each training and testing step of the proposed model are shown in Figure 5;

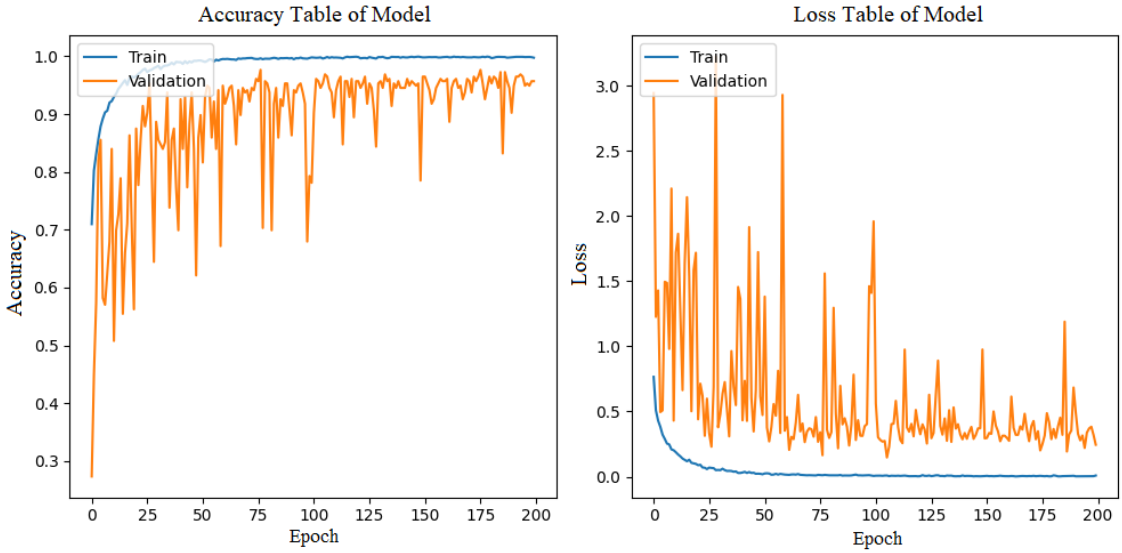


Figure 5. Accuracy and Loss Graph of Proposed Model

In Table 3, we trained 50 different CNN models by the dataset used for diseases from brain MRI images, along with the numbers of convolutional and dense layers, batch size, optimization method, kernel size, and the ratio of training and validation images from the 'Training' folder in the dataset, aiming to find our best-performing model. The accuracy, precision, and f1-score values obtained from training these models are summarized in the table below. Each row in Table 3 represents a proposed model architecture that has been trained and tested.

Table 3. Created Different CNN Models and Their Results (*:ADAM, **:SGD)

Con_1	Kernel_size	Con_2	Kernel_size	Con_3	Kernel_size	Con_4	Kernel_size	Con_5	Kernel_size	Con_6	Kernel_size	Con_7	Kernel_Size	Dense	Batch_size	Optimization	Epoch	Accuracy(%)	Precision(%)	F1-Score(%)
16	3	32	3	64	3	128	3							64	16	*	100	98.70	98.70	98.65
16	3	32	3	64	3	128	5							64	32	**	100	87.80	90.48	86.13
16	3	32	3	64	3	128	5							128	32	**	100	98.47	98.41	98.40
16	3	32	3	64	3	128	3	256	3					64	16	*	100	99.39	99.36	99.36
16	3	32	3	64	3	128	3	256	3					128	16	**	100	98.63	98.57	98.55
16	3	32	3	64	3	128	3	256	3					128	16	**	100	98.25	98.21	98.14
16	3	32	3	64	3	128	3	256	3					128-64	16	**	100	97.10	97.08	96.89
16	3	32	3	64	3	128	3	256	3					64	16	**	100	97.48	97.49	97.37
16	3	32	3	64	3	128	3	256	3					64	16	**	150	96.87	96.89	96.63
16	3	32	3	64	3	128	3	256	3					64	32	**	150	99.39	99.34	99.35
16	3	32	3	64	3	128	3	256	3					64	64	**	100	98.86	98.85	98.81
16	3	32	3	64	3	128	3	256	3	256	3			64	16	*	100	97.64	97.60	97.45
16	3	32	3	64	3	128	3	256	3	256	3			64	16	**	100	97.79	97.74	97.62
32	3	64	3	128	3	256	3							64	16	*	100	97.33	97.37	97.18
32	3	64	3	128	3	256	3	512	3					64	16	*	100	98.02	97.97	97.87
64	3	128	3	256	3	512	3							64	16	*	100	93.52	93.73	93.20
16	3	32	3	64	3	128	3	256	3	512	3			64	16	**	100	98.86	98.83	98.79
16	3	32	3	64	3	128	3	256	3	512	3			128	32	**	100	99.31	99.30	99.28
16	5	32	3	64	3	128	3	256	3	512	3			128	32	**	100	98.32	98.31	98.20
16	5	32	5	64	5	128	5	256	5	512	5			128	32	**	100	97.25	97.32	97.06
16	3	32	3	64	3	128	3	256	3	512	5			128	32	**	100	99.16	99.18	99.13
16	3	32	3	64	3	128	3	256	3	512	5			128	32	**	130	99.01	98.99	98.97
16	3	32	3	64	3	128	3	256	3	512	5			128	32	**	150	99.01	99.03	98.98
16	3	32	3	64	3	128	3	256	3	512	5			128	32	**	180	99.39	99.42	99.38
16	3	32	3	64	3	128	3	256	3	512	5			128	32	**	200	98.47	98.46	98.38
16	3	32	3	64	3	128	3	256	5	512	5			128	32	**	100	97.56	97.54	97.37
16	5	32	3	64	3	128	3	256	3	512	3			128	32	**	150	83.07	87.92	80.00
16	3	32	5	64	5	128	3	256	3	512	5			128	32	**	100	98.86	98.80	98.79
16	3	32	5	64	5	128	3	256	3	512	5			128	16	**	100	99.24	99.22	99.19
16	3	32	5	64	5	128	3	256	3	512	5			128	16	**	100	99.39	99.38	99.36
16	3	32	5	64	5	128	3	256	3	512	5			128	16	**	100	98.78	98.72	98.71
16	3	32	5	64	5	128	3	256	3	512	5			128	16	*	100	98.70	98.66	98.61
16	3	32	5	64	5	128	3	256	3	512	5			128	16	**	125	99.39	99.38	99.36
16	3	32	3	64	3	128	3	256	3	512	3			128	32	*	100	97.71	97.72	97.59
16	3	32	3	64	3	128	3	256	3	512	3			128	32	**	100	99.24	99.26	99.21
16	3	32	3	64	3	128	3	256	3	512	3			256	32	**	100	98.63	98.58	98.55
16	3	32	3	64	3	128	3	256	3	512	3			256	16	**	100	94.51	95.08	94.07
16	3	32	3	64	3	128	3	256	3	512	3			512	32	*	100	95.19	95.35	94.87
16	3	32	3	64	3	128	3	256	3	512	3			128	32	**	100	99.24	99.26	99.21
16	3	32	3	64	3	128	3	256	3	512	3			128	32	**	130	99.39	99.38	99.36
16	3	32	3	64	3	128	3	256	3	512	3			128	32	**	150	99.31	99.32	99.28
16	3	32	3	64	3	128	3	256	3	512	3			128	32	**	170	97.86	97.84	97.71
16	**3**	**32**	**3**	**64**	**3**	**128**	**3**	**256**	**3**	**512**	**3**			**128**	**32**	****	**200**	**99.47**	**99.46**	**99.44**
16	5	32	3	64	3	128	3	256	3	512	3			128	32	**	100	98.32	98.31	98.20
16	5	32	3	64	3	128	3	256	3	512	3			128	32	**	150	83.07	98.92	80.00
16	5	32	3	64	3	128	3	256	3	512	3			128	32	**	100	98.32	98.31	98.20
16	5	32	3	64	3	128	3	256	3	512	3			128	32	**	150	83.07	98.92	80.00
16	3	32	5	64	3	128	5	256	3	512	5	1024	3	128	16	**	100	99.24	99.21	99.19
16	3	32	3	64	3	128	3	256	3	512	3	1024	3	512	32	**	100	87.26	90.65	86.21
16	3	32	3	64	3	128	3	256	3	512	3	1024	3	256	16	*	100	98.63	98.54	98.54

4.1. Comparison of the Proposed Model with Studies in Literature

Table 4 displays a comparison of our suggested model's accuracy, precision, and f1-score values with those from earlier research.

Table 4. Comparison of Our Model with Previous Studies

Author	**Model**	**Performance Measures**		
		Accuracy	**Precision**	**F1-Score**
Kaplan et al.	nLBP+KNN	0,9556	0,956	0,956
Ghassemi et al.	Random Division and GAN method	0,956	0,9529	0,9510
Sowrirajan et al.	Hybrid VGG16-NADE	0,9601	0,9572	0,9568
Sultan et al.	Not specified	0,9613	0,9342	0,9475
Kibriya et al.	CNN Architecture with 13 Layers	0,972	0,97	0,9709
Haq et al.	GoogleNet Variable Architecture	0,973	0,9374	0,9548
Deepak et al.	GoogleNet+SVM	0,98	0,97	0,9749
Asiri et al.	BW-VGG19	0,98	0,9843	0,9821
Yerukalareddy et al.	CNN with GAN	0,9857	0,9875	0,9865
Rehman et al.	VGG16 with Fine Tuned	0,9869	-	-
Proposed Model	**6 Conv. Layer CNN**	**0,9947**	**0,99462**	**0,99442**

4.2. Comparison of the Proposed Model with Transfer Learning Models

In this study, well-known transfer learning models such as VGG16, DenseNet121, InceptionV3, ResNet50, MobileNetV2, and EfficientNetB0 were trained on the dataset, provided that the epoch, batch size, and optimization values applied in our proposed model remained the same. The results obtained from this training and the total parameter values are presented in Table 5.

Tablo 5. Comparison of Our Model with Transfer Learning Models

Model	Opt.	Batch-Size	Epoch	Accuracy (%)	Total Params
InceptionV3	SGD	32	200	92.98	22 million
MobileNetV2	SGD	32	200	94,96	3.4 million
DenseNet121	SGD	32	200	95.72	7.2 million
VGG16	SGD	32	200	96.87	14.8 million
ResNet-50	SGD	32	200	97.02	23.9 million
EfficientNetB0	SGD	32	200	99.46	4.2 million
Proposed Model	**SGD**	**32**	**200**	**99.47**	**2.2 million**

When Table 5 is examined, the accuracy values of the EfficientNetB0 model and the model we proposed gave almost the same accuracy result. Since EfficientNet model types dynamically increase the depth and width (compound scaling), they generally gives successful results in the medical image classification. However, our proposed architecture has approximately half as many parameters as the EfficientNetB0 model. Thus, our proposed architecture outperforms existing transfer learning models.

5. Discussion

This study developed a new CNN model with 6 convolutional layers for automatically detecting brain tumor types (Meningioma, Glioma, Pituitary) using brain MRI images. The dataset used consists of a total of 4 classes: 3 different diseased conditions and 1 normal class, sourced from the Kaggle database. According to results obtained from previous studies in the literature, our proposed model has shown superior performance.

When examining the CNN models created with different hyperparameter values in Table 3, it was observed that the SGD optimization algorithm outperformed the ADAM algorithm in terms of performance. Setting a high kernel value in the first layer adversely affected its performance; however, improvement was observed in performance when higher kernel values were used in subsequent layers. It was observed that the optimal value for the Dense layer is 128 based on the activation functions of preceding layers, and it was determined that the ADAM optimization algorithm produces better results with lower values. In some models, the accuracy values tended to decrease with increasing epochs initially, but it was observed that they started to rise again afterward.

Figure 5 displays the training and validation accuracy plot of the proposed model. When looking at this graph, it can be observed that at certain epoch values, the training and validation curves converge and overlap, while at specific epoch values, the gap between them widens. According to this graph, it is evident that the model exhibits a little sign of overfitting during the training phase. Figure 5 shows the training and validation loss plot of the proposed model. When looking at the loss plot, it is observed that although the difference between training and validation accuracy curves has increased at some epoch values, both accuracy and loss values for 'train' and 'validation' have moved closely together in both graphs.

In Table 4, when comparing our proposed model with previous studies, feature extraction processes were applied before implementing the deep learning model on the dataset in studies [17] and [22]. The model underwent pre-training and was subsequently retrained on the classification model in the study [20]. Despite other studies using more layers than our proposed model, our model achieved higher accuracy than previous studies.

According to Table 5, our model achieved the highest accuracy among the used transfer learning models with the least number of parameters.

6. Conclusion

By using computer technologies to detect brain tumors automatically and reliably at an early stage, physicians' workloads will be reduced and errors coming from manual examination would be eliminated. In order to detect diseases from medical photos, CNN models are frequently utilized. We optimized the hyperparameters in the CNN architecture to achieve the highest accuracy with a CNN model with the fewest parameters for the brain tumor dataset.

As seen in Table 3, 50 different models have been developed in accordance with the objectives of our study. Among these developed models, we have achieved the highest accuracy of 99.47% with the fewest number of layers and minimal hyperparameters. In the future, other hyperparameter selection techniques that determine the model's performance can be developed and tested using different CNN models, and by increasing the number of data in the dataset, better results can be achieved with new techniques.

References

[1] R. Singh, C. Prabha, S. Kumari, K. Murugan, M. R. Veeramanickam and T. Singh, "Accuracy Enhancement in Detecting Pituitary Tumors Using Deep Learning," *In 2023 International Conference on Sustainable Communication Networks and Application (ICSCNA)*, pp:1067-1072, IEEE, 2023

[2] L. Thau, V. Reddy, and P. Singh, "Anatomy, central nervous system," In StatPearls [Internet]. StatPearls Publishing, 2022

[3] B-L. Isabelle *et al*, "The Global Brain Health Survey: Development of a Multi-Language Survey of Public Views on Brain Health," *Front. Public Health, Sec. Public Health Education and Promotion*, Vol:8, doi: https://doi.org/10.3389/fpubh.2020.00387, 2020

[4] J. Cahill, G. LoBiondo-Wood, N. Bergstrom, and T. Armstrong, "Brain tumor symptoms as antecedents to uncertainty: An integrative review," *Journal of Nursing Scholarship*, vol. 44, no. 2, pp:145-155, 2012

[5] J. S. Barnholtz-Sloan, Q. T. Ostrom, D. Cote, "Epidemiology of Brain Tumors," *Neurologic Clinics*, Vol. 36, Issue 3, pp:395-419, 2018

[6] A-R. Fathi and U. Roelcke, "Meningioma," *Neuro-Oncology (Le Abrey, Section Editor) Curr Neurol Neurosci*, Vol. 13, no.337, Doi:10.1007/s11910-013-0337-4, 2013

[7] J. Wiemels, M. Wrensch and E. B. Claus, "Epidemiology and Etiology of Meningioma," *Invited Review, J*

Neurooncol, Vol. 99, pp:307-314, Doi: 10.1007/s11060-010-0386-3, 2010
[8] C. Apra, M. Peyre and M. Kalamarides, "Current Treatment Options for Meningioma," *Expert Review of Neurotherapeutics, HAL Open Science*, Vol. 18, no. 3, pp:241-249, 2018
[9] A.S. Modrek, N.S. Bayin and D.G. Placantonakis, "Brain Stem Cells as the Cell of Origin in Glioma," *World J Stem Cells*, Vol. 6, no. 1, pp:43-52, 2014
[10] N. A. O. Bush, S. M. Chang and M. S. Berger, "Current and Future Strategies for Treatment of Glioma," *Neurosurg Rev*, vol. 40, pp:1-14, 2017
[11] S. D. Muhammad and Z. Kobti, "An Ensemble Deep Learning Approach for Enhanced Classification of Pituitary Tumors," *In 2023 IEEE Symposium Series on Computational Intelligence, IEEE*, p: 427-432, 2023
[12] A. M. Gab Allah, A. M. Sarhan and N. M. Elshennawy, "Classification of brain MRI tumor images based on deep learning PGGAN Augmentation," *Diagnostics*, Vol. 11, no. 12, 2021
[13] M. K. Abd-Ellah, A. I. Awad, A. A. Khalaf and H. F. Hamed, "A review on brain tumor diagnosis from MRI images: Practical implications, key achievements, and lessons learned," *Magnetic resonance imaging*, Vol. 61, pp: 300-318, 2019
[14] S. A. Yazdan, R. Ahmad, N. Iqbal, A. Rizwan, A. N. Khan, and D. H. Kim, "An efficient multi-scale convolutional neural network based multi-class brain MRI classification for SaMD," *Tomography*, Vol. 8, no. 4, pp:1905-1927, 2022
[15] M. R. Ismael and I. Abdel-Qader, "Brain Tumor Classification via Statistical Features and Back-Propagation Neural Network," *2018 IEEE Uluslararası Elektro/Bilgi Teknolojisi Konferansı*, 2018.
[16] A. Pashaei, H. Sajedi and N. Jazayeri, "Brain Tumor Classification via Convolutional Neural Network and Extreme Learning Machines," *ICCKE2018, Ferdowsi University of Mashhad*, pp: 314-319
[17] S. Deepak, P.M. Ameer, "Brain tumor classification using deep CNN features via transfer learning," *Computers in Biology and Medicine, ELSEVIER*, 2019
[18] Z. N. K. Swati, Q. Zhao, M. Kabir, F. Ali, S. Ahmed and J. Lu, "Brain Tumor Classification for MR Images Using Transfer Learning and Fine-Tuning," *Computerized Medical Imaging and Graphics, ELSEVIER*, 2019
[19] H. H. Sultan, N. M. Salem and W. Al-Atabany, "Multi-classification of brain tumor images using deep neural network," *IEEE Access*, Vol. 7, pp:69215–69225, 2019
[20] N. Ghassemi, A. Shoeibi and M. Rouhani, "Deep neural network with generative adversarial networks pre-training for brain tumor classification based on MR images," *Biomedical Signal Processing and Control, Elsevier*, 2020 doi: https://doi.org/10.1016/j.bspc.2019.101678
[21] R. Hashemzehi, S. J. S. Mahdavi, M. Kheirabadi and S. R. Kamel, "Detection of brain tumors from MRI images base on deep learning using hybrid model CNN and NADE," *Biocybernetics And Biomedical Engineering, Elsevier*, pp: 1225-1232, doi: https://doi.org/10.1016/j.bbe.2020.06.001, 2020
[22] K. Kaplan, Y. Kaya, M. Kuncan and H. M. Ertunç, "Brain tumor classification using modified local binary patterns (LBP) feature extraction methods," *Medical Hypotheses, Elsevier*, doi: https://doi.org/10.1016/j.mehy.2020.109696 , 2020
[23] A. Rehman, S. Naz, M. I. Razzak, F. Akram, and M. Imran, "A deep learning based framework for automatic brain tumors classification using transfer learning," *Circuits, Systems, and Signal Processing*, Vol. 39, no. 2, pp:757–775, doi:10.1007/S00034-019-01246-3/TABLES/8, 2020
[24] W. Ayadi, W. Elhamzi, I. Charfı and M. Atrl, "Deep CNN for Brain Tumor Classification," *Neural Processing Letters, Springer*, Vol. 53, pp:671-700, doi: https://doi.org/10.1007/s11063-020-10398-2 , 2021
[25] E. U. Haq, H. Jianjun, K. Li, H. U. Haq and T. Zhang, "An MRI-based deep learning approach for efficient classification of brain tumors," *Journal of Ambient Intelligence and Humanized Computing, Springer*, doi: https://doi.org/10.1007/s12652-021-03535-9 , 2023
[26] S. R. Sowrirajan, S. Balasubramanian and R. S. P. Raj, "MRI Brain Tumor Classification Using a Hybrid VGG16-NADE Model," *Article-Engineering, Technology and Techniques, BABT*, Vol. 66 doi: https://doi.org/10.1590/1678-4324-2023220071 ,2022
[27] D. R. Yerukalareddy and E. Pavlovskiy, "Brain Tumor Classification Based on MR Images Using Gan as a Pre-trained Model," *IEEE Ural-Siberian Conference On Computational Technologies in Cognitive Science, Genomics And Biomedicine (CSGB)*, pp:380-384, doi: 10.1109/CSGB53040.2021.9496036, 2021
[28] H. Kibriya, M. Masood, M. Nawaz, T. Nazir, "Multiclass classification of brain tumors using a novel CNN architecture," *Multimedia Tools and Applications, SPRINGER*, Vol. 81, pp:29847-29863, doi: https://doi.org/10.1007/s11042-022-12977-y ,2022
[29] A. A. Nasiri et al, "Block-Wise Neural Network for Brain Tumor Identification in Magnetic Resonance Images," *Computers, Materials & Continua, Tech Science Press*, Vol. 73, no.3, pp: 5735-5753, doi: 10.32604/cmc.2022.03174, 2022
[30] M. Kaya, and Y. Çetin-Kaya, "A novel ensemble learning framework based on a genetic algorithm for the classification of pneumonia," *Engineering Applications of Artificial Intelligence*, Vol. 133, no. 108494, 2024
[31] M. Kaya and Y. Çetin-Kaya, "A Novel Deep Learning Architecture Optimization for Multiclass Classification of Alzheimer's Disease Level," *IEEE Access*, 2024
[32] M. Kaya, "Bayesian Optimization-based CNN Framework for Automated Detection of Brain Tumors," *Balkan Journal of Electrical and Computer Engineering*, Vol. 11, no. 4, pp:395-404, 2023

[33] Y. Çetin-Kaya and M. Kaya, "A Novel Ensemble Framework for Multi-Classification of Brain Tumors Using Magnetic Resonance Imaging," *Diagnostics*, Vol. 14, no. 4, 2024
[34] https://www.kaggle.com/datasets/masoudnickparvar/brain-tumor-mri-dataset
[35] K. O'Shea and R. Nash, "An Introduction to Convolutional Neural Networks," *arXiv:* 1511.08458v2, 2015
[36] E. Cengil and A. Çınar, "A New Approach For Image Classification: Convolutional Neural Network," *Europan Journal of Technic, INESEG*, Vol 6, Num 2, pp: 96-103, 2016
[37] E. Ö. YILMAZ and T. KAVZOĞLU, "Derin Öğrenmenin Temel Prensipleri ve Uzaktan Algılama Alanındaki Uygulamaları," *Harita Dergisi*, Vol. 166, pp. 25-43, 2021
[38] F. Özyurt, E. Sert, E. Avci, and E. Dogantekin, "Brain tumor detection based on Convolutional Neural Network with neutrosophic expert maximum fuzzy sure entropy," *Measurement*, Vol. *147*, no. 106830, 2019
[39] Y. Lu, S. Yi, N. Zeng, Y. Liu and Y. Zhang, "Identification of rice diseases using deep convolutional neural networks," *Neurocomputing, Elsevier*, Vol. 267, pp:378-384, 2017
[40] R. Yamashita, M. Nishio, R. K. G. Do and K. Togashi, "Convolutional Neural Networks: An Overview and Application in Radiology," *Insights Into Imaging*, Vol. 9, no. 4, pp:611-629, 2018
[41] I. Goodfellow, Y. Bengio, A. Courville, "Deep learning," *MIT Press*, 2016
[42] W. Hao, W. Yizhou, L. Yaqin and S. Zhili, "The Role of Activation Function in CNN," *Proceedings, 2020 2nd International Conference on Information Technology and Computer Application*, *ITCA*, pp:429-432, doi: https://doi.org/10.1109/ITCA52113.2020.00096, 2020
[43] B. Singh, S. Patel, A. Vijavvargiya and R. Kumar, "Analyzing the Impact of Activation Functions on the Performance of the Data-Driven Gait Model," *Results in Engineering*, Vol. 18, 2023
[44] S. Sharma, S. Sharma and A. Athaiya, "Activation Functions in Neural Networks," *International Journal of Engineering Applied Sciences and Technology*, Vol. 4 no. 12, pp:310-316, 2020
[45] S. R. Dubey, S. K. Singh and B. B. Chaudhuri, "Activation functions in deep learning: A comprehensive survey and benchmark," *Neurocomputing*, Vol. 503, 92-108, 2022
[46] Bayram F., "Derin Öğrenme Tabanlı Otomatik Plaka Tanıma," *Politeknik Dergisi*, Vol. 23, no. 4, pp:955-960, 2020
[47] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," *Journal of Machine Learning Research*, Vol. 15, no. 2014, pp:1929-1958, 2014
[48] K. Liu, G. Kang, N. Zhang and B. Hou, "Breast cancer classification based on fully-connected layer first convolutional neural networks,", *IEEE Access*, Vol. 6, pp:23722-23732, 2018
[49] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, Vol. 521, no. 7553, pp:436-444, 2015

Article Information Form

Acknowledgments
We express our gratitude to the referees for their insightful comments that enhanced the paper's presentation.

Authors' Contributions
The two authors worked together to complete this project. The final manuscript was read and approved by all the authors.

Conflict of Interest
There is no conflict of interest declared by the writers.

Funding
Funding has not been revealed by the writers.

Plagiarism Statement
This article has been scanned by iThenticate ™.

SAKARYA UNIVERSITY JOURNAL OF COMPUTER AND INFORMATION SCIENCES
http://saucis.sakarya.edu.tr/
e-ISSN: 2636-8129
Publisher: Sakarya University
Vol. 7, No. 3, 494-509, 2024
DOI: 10.35377/saucis...1504759
SAUCIS
SAKARYA UNIVERSITY

RESEARCH ARTICLE

MQTT in Action: Building Reliable and Scalable Home Automation Systems

Maysaa Salama[1, *], Bilal Raslen [1]

[1] Faculty of Computer and Information Sciences, Computer Engineering, Sakarya University, Türkiye

Corresponding author:
Maysaa SALAMA,
Sakarya University, Faculty of Computer and Information Sciences, Computer Engineering
Türkiye
maysaa.salama@ogr.sakarya.edu.tr

Article History:
Received: 25.06.2024
Accepted: 26.12.2024
Published Online:31.12.2024

ABSTRACT

This study presents the development and implementation of an MQTT-based home automation system, emphasizing its reliability, scalability, and efficiency. The system integrates the ESP8266 microcontroller with various sensors, including IR, temperature, DHT11, sound, and vibration sensors. Leveraging the lightweight MQTT protocol for real-time data transmission and PostgreSQL for robust data storage, the system monitors environmental conditions, detects specific events, and performs appropriate actions. Performance tests indicate the system maintains low response times, high data accuracy, and exceptional reliability, making it a viable solution for modern smart homes. The findings suggest that the system enhances security, optimizes energy usage, and improves overall comfort for homeowners. Future research will focus on enhancing the security of the MQTT protocol against spoofing attacks through the integration of AI for threat detection and blockchain technology for enhanced data security.

Keywords: MQTT, Home Automation, IoT, ESP8266, Sensors, Real-time Monitoring, Data Transmission, Reliability, Scalability, PostgreSQL

1. Introduction

Message Queuing Telemetry Transport (MQTT) is a machine-to-machine (M2M) Internet of Things connectivity protocol designed as an extremely lightweight publish/subscribe messaging transport. It is ideal for connecting remote devices with a small code footprint and minimal network bandwidth. MQTT is widely used across various applications, from mobile applications to lightweight communication in automotive and industrial systems, most notably, home automation [1]. In the realm of home automation, MQTT stands out for its simplicity and efficiency, which are crucial in environments where bandwidth and power usage are limited. Home automation systems need robust and scalable communication protocols to manage the many devices and frequent message exchanges typical in smart homes [2]. MQTT's capability to provide real-time updates and maintain stable connections, even over unreliable networks, makes it particularly suitable for this role. This functionality enables various applications, from basic home security to intricate automation scenarios. This paper explores the implementation of MQTT in home automation systems, emphasizing its reliability and scalability. The paper aims to analyze the advantages of MQTT over other communication protocols in terms of system reliability and data handling efficiency. It will demonstrate through case studies how MQTT can be implemented in home automation to achieve scalable and secure systems and will address the challenges faced during the implementation of MQTT in these systems, proposing viable solutions. In addition to its efficiency and scalability, securing MQTT communications is critical, especially in home automation systems where sensitive data is exchanged between devices. MQTT broker authentication and authorization mechanisms are essential for preventing unauthorized access. Typically, MQTT brokers support username and password authentication, where each client must provide valid credentials to establish a connection. To further strengthen security, OAuth or token-based authentication can be employed, which provides a more robust way to verify the identity of clients. Data encryption methods are also integral to maintaining the confidentiality and integrity of the data transmitted via MQTT. Transport Layer Security (TLS) is commonly used in conjunction with MQTT to encrypt data during transit, protecting it from eavesdropping or tampering. Using TLS, all communication between the client and broker is secured, ensuring that sensitive sensor data is protected. In addition to these measures, access control lists (ACLs) can be used to define specific permissions for each MQTT client, controlling what topics they can publish to or subscribe to, thereby reducing the risk of unauthorized actions. By combining these mechanisms, the system ensures a high level of protection against unauthorized access, data breaches, and potential cyber threats. With its designed efficiencies and growing adoption in various sectors,

Cite as: Salama M. and Raslen B.. "MQTT in Action: Building Reliable and Scalable Home Automation Systems". *Sakarya University Journal of Computer and Information Sciences*. vol: 7 no: 3. pp: 494-509, 2024. Doi: 10.35377/saucis...1504759

MQTT is pivotal in advancing home automation systems [3]. It not only enhances how devices communicate but also ensures that these communications are reliable and scalable.

2. Related Work

The use of MQTT (Message Queuing Telemetry Transport) in home automation has gained significant attention due to its lightweight nature, efficiency, and scalability. This section reviews key advancements in MQTT applications, comparisons with alternative IoT protocols, security enhancements, and scalability improvements, all of which underscore MQTT's role in modern home automation systems.

Advancements in MQTT Applications

Many studies have explored the optimization of MQTT for smart home environments. For instance, in [4], the authors proposed a dynamic model for adjusting MQTT message frequencies to optimize power consumption and network efficiency, predicting network conditions to adjust message rates as needed [4]. This significantly reduced power usage while maintaining high system responsiveness. Another study examined MQTT's integration for managing energy distribution from renewable sources within smart homes, showcasing MQTT's role in promoting sustainable energy practices [5]. Moreover, it developed an MQTT-based framework that enhanced interoperability across various smart home devices and reduced latency, making it a powerful tool for unifying disparate devices into a cohesive network [6].

Comparative Studies with Other IoT Protocols

While MQTT is well-suited for low-bandwidth, reliable messaging, comparisons with other IoT protocols highlight its unique benefits and limitations [7] provided a rigorous comparison between MQTT and CoAP (Constrained Application Protocol), noting that MQTT offers superior throughput in high-demand scenarios. At the same time, CoAP performs more efficiently in energy-constrained settings. Another study by [8] contrasted MQTT with HTTP, emphasizing MQTT's enhanced security features and suitability for dynamic network environments typical of home automation. Additionally, [9] compared MQTT with AMQP (Advanced Message Queuing Protocol), finding that MQTT's lower overhead and support for real-time data transmission make it preferable for latency-sensitive applications.

In contrast to protocols like Zigbee and Z-Wave, which operate as mesh networks optimized for local device communication, MQTT's publish-subscribe architecture supports seamless device interoperability across broader network scales. Zigbee and Z-Wave excel in short-range, low-power operations within confined spaces, making them ideal for small-scale automation but limited in scalability and interoperability across different network infrastructures. Thread is an IP-based networking protocol for low-power mesh networks, while MQTT is a lightweight messaging protocol. MQTT can be used over Thread networks to provide efficient publish/subscribe messaging, making it advantageous in bandwidth-constrained environments. Another competing protocol, offers IP-based connectivity but lacks the lightweight simplicity of MQTT, which is advantageous in bandwidth-constrained environments. Overall, MQTT's flexibility and efficient message delivery make it a robust choice for IoT applications requiring both local and remote communication, whereas alternatives like Zigbee and Z-Wave may be better suited for exclusively local automation.

Security Enhancements in MQTT-Based Systems

Security remains a primary concern for MQTT-based home automation systems, which manage sensitive data and control critical home functions. Various studies have proposed robust security measures to counteract potential threats in MQTT networks. For instance, [10] introduced a security framework integrating advanced encryption standards and secure authentication to defend against cyber threats. Building on this, [11] developed a machine learning-based intrusion detection system (IDS) specifically for MQTT, which significantly improved real-time threat detection and mitigation. Additionally, [12] proposed a lightweight encryption scheme tailored for resource-constrained MQTT devices, enabling data protection without compromising performance.

Beyond spoofing attacks, MQTT-based systems are vulnerable to Denial of Service (DoS) attacks, Man-in-the-Middle (MitM) attacks, and data manipulation. DoS attacks can overload the MQTT broker with excessive requests, disrupting service availability. To mitigate this, rate-limiting mechanisms can be implemented at the broker level, alongside firewall rules and traffic monitoring to block malicious traffic early. MitM attacks, where an attacker intercepts and modifies communication, can be mitigated with Transport Layer Security (TLS 1.3), which encrypts data to maintain confidentiality and integrity. Enhanced authentication measures, such as token-based access and OAuth, ensure that only verified clients can access sensitive data streams. To detect data manipulation, digital signatures and message integrity checks verify message authenticity and content integrity. By combining these measures with IDS technology, which continuously monitors anomalous traffic, MQTT systems can provide a resilient security framework suitable for sensitive home automation

applications.

Scalability and Interoperability Improvements

Scalability is essential for home automation systems that require expansion as new devices are integrated. Several studies have addressed scalability in MQTT-based systems. For example, [13] proposed a load-balancing technique that leverages MQTT to distribute workloads across multiple brokers, enhancing system reliability and fault tolerance. Additionally, it explored decentralized MQTT architecture, reducing the load on central brokers by enabling data processing at the network edge, which significantly improves scalability [14].

Interoperability is another important factor, particularly in multi-standard environments. Morgado-Dias and Quintal [15] examined MQTT's role as a bridging protocol for communication between heterogeneous devices, ensuring compatibility across different smart home ecosystems.

Furthermore, [16] explored MQTT's integration with Zigbee and Z-Wave protocols, facilitating communication across diverse networks and enhancing system flexibility. These interoperability improvements make MQTT well-suited for smart home applications, as it can effectively integrate with other protocols and support a diverse array of devices.

Summary of Key Contributions

Table 1 summarizes the key contributions of MQTT applications for IoT systems, categorizing application advancements, comparative studies, security, and scalability.

Table 1 Key Contributions in MQTT Applications for IoT Systems

Contribution Area	Study	Key Findings
MQTT Applications	[4]	Optimized MQTT message frequencies for better power consumption and efficiency.
	[5]	Managed energy distribution from renewable sources in smart homes using MQTT.
	[6]	Improved interoperability and reduced latency with an MQTT-based framework.
Comparative Studies	[7]	MQTT vs. CoAP: MQTT has better throughput; CoAP is more energy-efficient.
	[8]	MQTT offers enhanced security compared to HTTP.
	[9]	MQTT has a lower overhead and is better for real-time applications than AMQP.
Security Enhancements	[10]	Developed a secure MQTT framework with advanced encryption.
	[11]	Machine learning-based intrusion detection for MQTT networks.
	[12]	Lightweight encryption for MQTT, enhancing security with low overhead.
Scalability and Interoperability	[13]	Load-balancing in MQTT for better reliability.
	[14]	Decentralized MQTT for improved scalability.
	[15]	Bridging protocol for seamless communication across devices.
	[16]	Integration of MQTT with Zigbee and Z-Wave for better interoperability.

3. System Architecture

The architecture of the MQTT-based home automation system is illustrated in *Figure 1*. This figure provides a visual representation of the interactions between various components, including sensors, the ESP8266 microcontroller, and the MQTT broker. The system is designed to monitor environmental conditions, detect specific events, and take appropriate actions based on sensor inputs.

Components and Workflow

The system consists of several interconnected components working seamlessly to enhance the smart home environment. The sound sensor acts as the initial trigger, detecting ambient sound levels and activating the IR sensor to monitor object movement. Upon detecting movement, the IR sensor signals the lighting system, prompting it to turn on the lights. This interaction ensures that the lighting is responsive to environmental changes.

Following movement detection, the IR sensor activates the temperature and humidity sensors, which measure ambient conditions to determine if adjustments to the indoor climate are necessary. Based on their readings, these sensors may reset the air conditioning system to optimize the indoor environment for comfort and energy efficiency.

Simultaneously, the vibration sensor operates continuously, monitoring for unusual movements or vibrations within the home. If the vibration sensor detects any anomalies, it activates the speaker (piezo) to alert the occupants, ensuring prompt awareness of potential security issues. This coordinated workflow among the sound, IR, temperature, humidity, and vibration sensors, along with the lighting and air conditioning systems, demonstrates the system's ability to maintain a secure, efficient, and comfortable home environment.

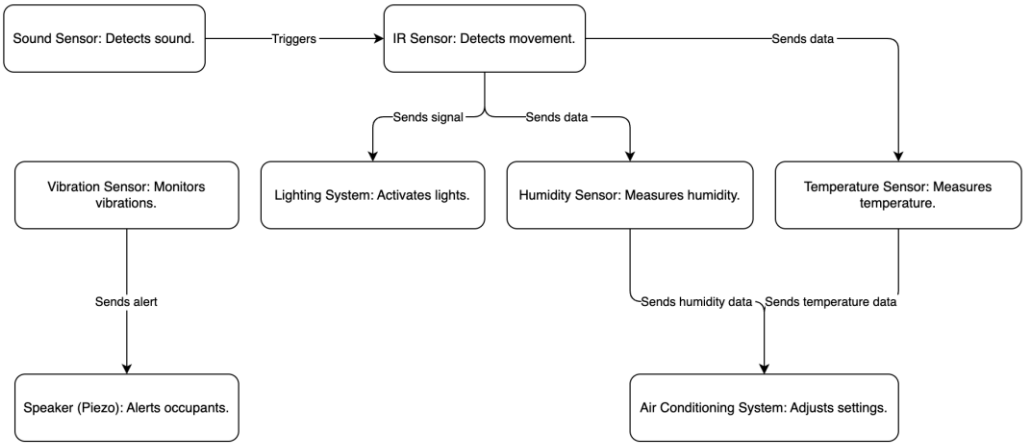


Figure 1 System Architecture of the MQTT-based Home Automation System.

The ESP8266 microcontroller acts as the central hub for data collection and communication. It reads data from the connected sensors and transmits it to an MQTT broker using the MQTT protocol. The MQTT broker facilitates communication between the sensors and the data storage system.

As shown in *Figure 1* key interactions and data flows include:

- **Sensor to ESP8266:** Sensors continuously send data to the ESP8266 microcontroller.
- **ESP8266 to MQTT Broker:** ESP8266 formats the sensor data and transmits it to the MQTT broker.
- **MQTT Broker to Database:** The MQTT broker relays the data to a PostgreSQL database for storage and analysis.

The diagram showcases the interconnected nature of the system, highlighting the role of each component and the flow of information. This integrated approach ensures real-time monitoring, efficient communication, and effective management of the home automation system.

4. Methodology

This section delineates the comprehensive methodology employed in the development and implementation of the MQTT-based home automation system. The system architecture integrates the ESP8266 microcontroller, an array of sensors (IR, temperature, DHT11, sound, and vibration), the MQTT protocol for efficient communication, and PostgreSQL for robust data storage.

The experimental setup comprises multiple ESP8266 microcontrollers connected to various sensors to monitor environmental conditions and detect specific events. As shown in Figure 2, the setup includes the DHT11 temperature and humidity sensor, a sound sensor, an IR sensor, and a vibration sensor, all mounted on a breadboard. This configuration allows for efficient data collection and transmission for analysis.

We implemented robust authentication and authorization mechanisms within the MQTT broker to ensure the security of data transmitted within the system. Each ESP8266 microcontroller is required to authenticate using a username and password before it can connect to the MQTT broker. For enhanced security, we have also integrated TLS (Transport Layer Security) encryption, which ensures that all data transmitted between the sensors, broker, and database is encrypted, safeguarding it from potential eavesdropping or interception. Furthermore, access control lists (ACLs) are configured on the MQTT broker to restrict each sensor's access to specific topics, preventing unauthorized publishing or subscribing. By utilizing TLS and ACLs, we guarantee that only authorized devices can access the network, and the integrity of the sensor data is maintained throughout the communication process. This layered approach to security ensures that the system remains resilient against unauthorized access and potential data breaches, providing a high level of trustworthiness for real-time home automation

applications.

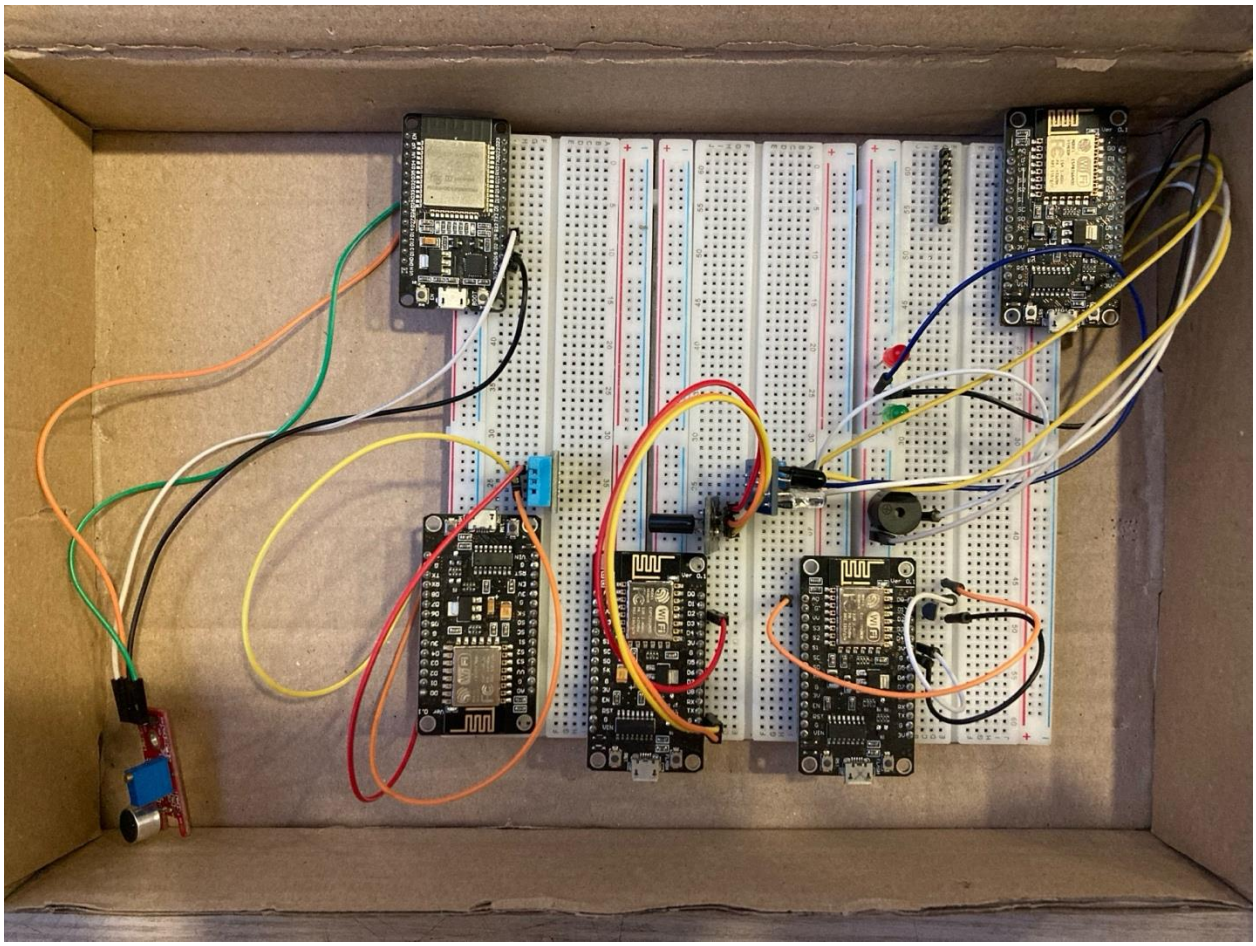


Figure 2 Prototype Setup of MQTT-Based Home Automation System

Hardware Configuration

The hardware configuration of the MQTT-based home automation system integrates key components to enable efficient data collection and communication. At the core of the system is the ESP8266 microcontroller, which serves as the central processing unit. Its built-in Wi-Fi capabilities make it ideal for IoT applications, allowing it to aggregate data from multiple sensors and manage network communication seamlessly.

The system employs a comprehensive sensor suite to monitor environmental conditions and detect specific events. An IR sensor is utilized for motion detection, triggering actions based on detected movement. A temperature sensor monitors ambient temperature, while the DHT11 sensor provides dual functionality by measuring both humidity and temperature levels. The sound sensor detects environmental sound variations, identifying significant noise levels, and the vibration sensor registers physical impacts or unusual movements indicative of potential disturbances.

These sensors are interfaced with the ESP8266, which processes their output and transmits the data using the MQTT protocol. Together, this hardware configuration forms the backbone of the system, enabling precise environmental monitoring and reliable communication, which is essential for effective home automation.

Software Implementation

The software implementation encompasses three main processes: data collection, data transmission using the MQTT protocol, and data storage.

1. **Sensor Initialization and Data Collection:** ESP8266 is programmed to continuously read data from the connected sensors regularly. The collected data includes temperature, humidity, motion status, sound levels, and vibration status. The readings are processed and formatted for transmission. Figure 3 shows the Key lines of code for the data collection setup:

```
#include <DHT.h>
#define DHTPIN 2
#define DHTTYPE DHT11

DHT dht(DHTPIN, DHTTYPE);

void setup() {
    Serial.begin(9600);
    dht.begin();
    pinMode(SOUND_SENSOR_PIN, INPUT);
    pinMode(IR_SENSOR_PIN, INPUT);
    pinMode(VIBRATION_SENSOR_PIN, INPUT);
}

void loop() {
    float h = dht.readHumidity();
    float t = dht.readTemperature();
    int sound = digitalRead(SOUND_SENSOR_PIN);
    int ir = digitalRead(IR_SENSOR_PIN);
    int vibration = digitalRead(VIBRATION_SENSOR_PIN);

    char msg[50];
    snprintf(msg, 50, "Temp: %f Humidity: %f Sound: %d IR: %d Vibration: %d", t, h, sound, ir, vibration);
    client.publish("home/monitor", msg);

    delay(2000);
}
```

Figure 3 Sensor Initialization and Data Collection Code.

To monitor environmental conditions, we used a DHT11 sensor to collect temperature and humidity data. Figure 4 illustrates the measurements recorded over a specified period, showcasing the sensor's performance and the consistency of the data collected.

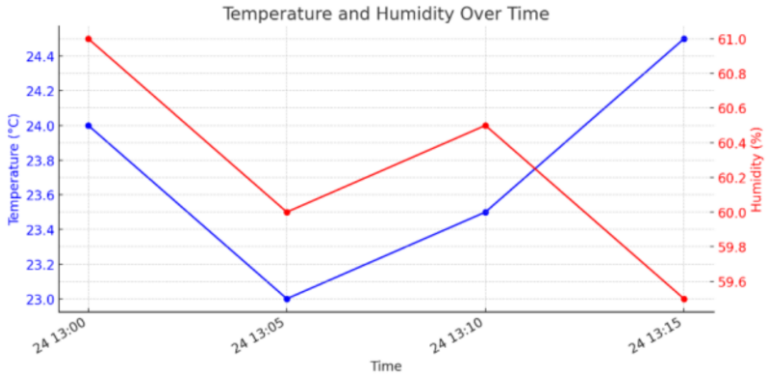


Figure 4 Temperature and Humidity Over Time.

2. **Data Transmission:** The collected data is transmitted to an MQTT broker using the MQTT protocol. MQTT is chosen for its lightweight nature, which ensures efficient and reliable communication. The broker acts as an intermediary, facilitating the transmission of data between the ESP8266 and the data storage system. Figure 5 shows the key lines of code for the MQTT setup:

```
import paho.mqtt.client as mqtt # Import the MQTT library
# Create an MQTT client instance
client = mqtt.Client()
try:
# Connect to the MQTT broker
client.connect("mqtt.eclipse.org", 1883, 60)
# Publish a test message to the "home/monitor" topic
client.publish("home/monitor", "Test message", qos=1)
except Exception as e:
# Print an error message if something goes wrong
print(f"Error: {e}")
finally:
# Disconnect from the broker
client.disconnect()
```

Figure 5 MQTT Data Transmission Code.

To evaluate the efficiency of the sensors, we measured the response time for each sensor type, defined as the time elapsed between the detection of an event by the sensor and the corresponding signal being registered by the ESP8266 microcontroller. The response times were obtained through a systematic experimental setup and measurement methodology to ensure clarity, reproducibility, and credibility.

Experimental Setup: The sensors were connected to the ESP8266 microcontroller, which recorded timestamps when signals were detected. For each sensor, specific events were manually triggered—such as sound for the sound sensor, motion for the IR sensor, and vibration for the vibration sensor. ESP8266 logged the time difference between the initiation of the event and the sensor's response.

Measurement Methodology: Response times were measured using the Millis() function in the Arduino environment, capturing precise time differences in milliseconds. Each measurement was repeated 10 times to account for variability, and the average response time was calculated for reliability. This method ensured that the data captured reflected consistent performance under controlled conditions.

Figure 6 illustrates the results, showing that the IR sensor exhibited the shortest response time, while the temperature and humidity sensor recorded the longest. These findings highlight the varying efficiencies of the sensors in detecting and relaying environmental changes, with implications for real-time monitoring and control in the home automation system.

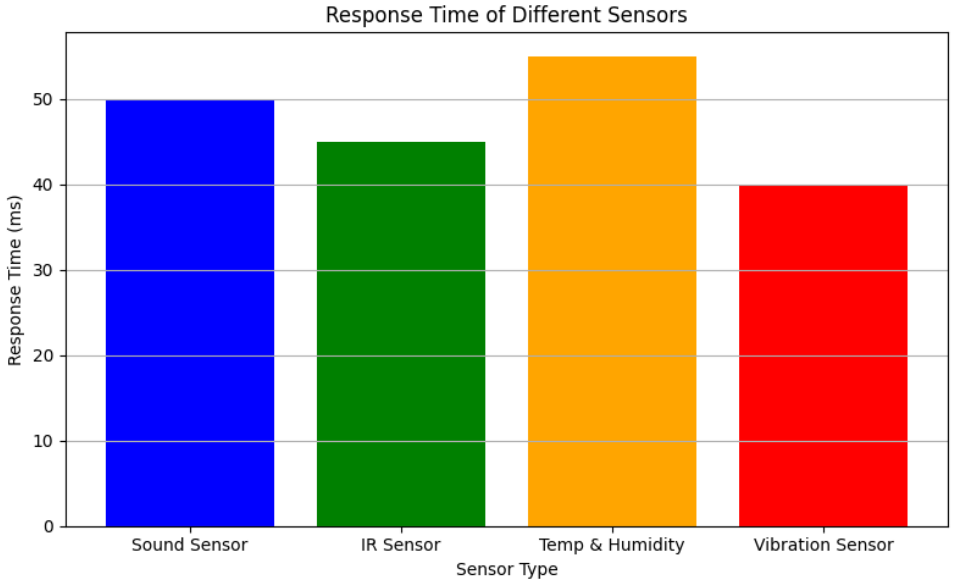


Figure 6 Response Time of Different Sensors.

3. **Data Storage:** The transmitted data is stored in a PostgreSQL database, ensuring a reliable and scalable solution for data management. The PostgreSQL database is configured to handle incoming data from the MQTT broker, which is systematically stored for subsequent analysis and monitoring. Figure 7 shows the SQL code used to create the `sensor_data` table, and Figure 8 shows the Python code used to insert data into the table.

```
CREATE TABLE sensor_data (
    id SERIAL PRIMARY KEY,
    timestamp TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    temperature FLOAT,
    humidity FLOAT,
    motion BOOLEAN,
    sound INT,
    vibration BOOLEAN
);
```

Figure 7 PostgreSQL Table Creation Code.

```
cursor.execute(
    "INSERT INTO sensor_data (temperature, humidity, motion, sound, vibration) VALUES (%s, %s, %s, %s, %s)",
    (temperature_value, humidity_value, motion_value, sound_value, vibration_value)
)
```

Figure 8 PostgreSQL Data Insertion Code.

This methodology ensures an effective and reliable system for home automation, leveraging the strengths of the ESP8266 microcontroller, the versatility of MQTT for communication, and the robustness of PostgreSQL for data storage. The

system achieves real-time data processing, secure communication, and efficient data management through this integration, providing a scalable solution for modern smart home applications.

The data collected from various sensors was transmitted to an MQTT broker and stored in a PostgreSQL database. Figure 9 shows the proportion of data stored by each MQTT topic, ensuring balanced data management and efficient handling of sensor data.

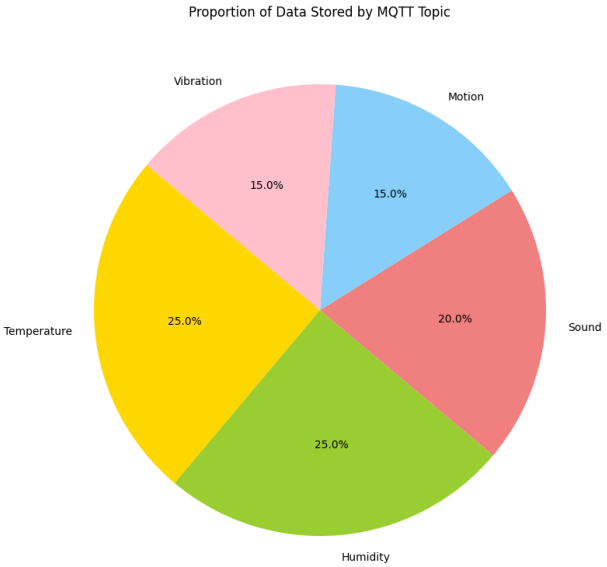


Figure 9 Proportion of Data Stored by MQTT Topic.

Table 2 Temperature Sensor Data.

Topic	Payload	Detection_Time
home/temperature	25.37	2024-05-24 14:20:00
home/temperature	23.89	2024-05-24 13:55:00
home/temperature	22.67	2024-05-24 13:30:00
home/temperature	24.15	2024-05-24 13:05:00

Table 3 Humidity Sensor Data.

Topic	Payload	Detection_Time
home/humidity	58.25	2024-05-24 14:15:00
home/humidity	60.12	2024-05-24 13:50:00
home/humidity	59.88	2024-05-24 13:25:00
home/humidity	61.75	2024-05-24 13:00:00

Table 4 Motion Sensor Data.

Topic	Payload	Detection_Time
home/motion	1	2024-05-24 14:10:00
home/motion	0	2024-05-24 13:45:00
home/motion	1	2024-05-24 13:20:00
home/motion	0	2024-05-24 12:55:00

Table 5 Sound Sensor Data.

Topic	Payload	Detection_Time
home/sound	72.34	2024-05-24 14:05:00
home/sound	68.45	2024-05-24 13:40:00
home/sound	70.23	2024-05-24 13:15:00
home/sound	65.12	2024-05-24 12:50:00

Table 6 Vibration Sensor Data

Topic	Payload	Detection_Time
home/vibration	0	2024-05-24 14:00:00
home/vibration	1	2024-05-24 13:35:00
home/vibration	0	2024-05-24 13:10:00
home/vibration	1	2024-05-24 12:45:00

To ensure the system's reliability, we monitored its uptime for 30 days. Figure 10 illustrates the system's uptime percentage, demonstrating its stability and continuous operation without significant downtimes.

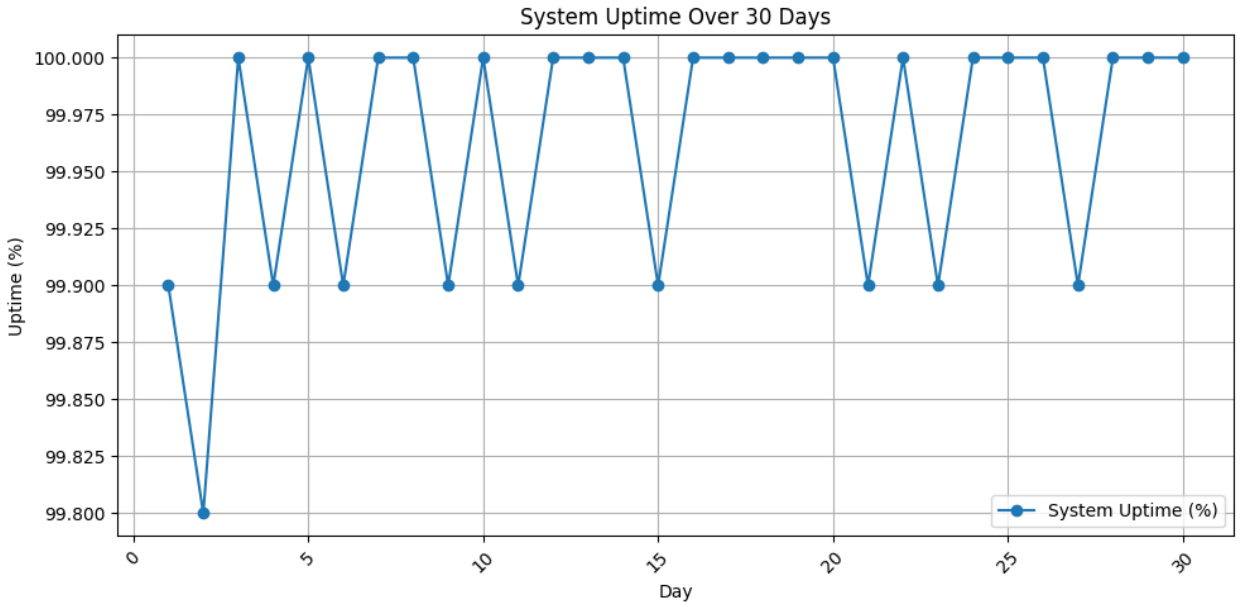


Figure 10 System Uptime Over 30 Days.

5. Results and Discussion

System Performance and Data Analysis

Implementing the MQTT-based home automation system was tested for various performance metrics, including response time, data accuracy, and system reliability. The following subsections discuss the results obtained from these tests and analyze the system's overall performance.

1. Response Time

The response time of the system, defined as the time taken from detecting an event to performing the corresponding action, was measured. The average response time for different sensor triggers is shown in *Figure 6*:

- Sound Sensor: 50 ms
- IR Sensor: 45 ms
- Temperature and Humidity Sensors: 55 ms
- Vibration Sensor: 40 ms

These response times indicate the system can react quickly to environmental changes, ensuring timely actions such as turning on lights or adjusting the air conditioning.

2. Data Accuracy

The accuracy of the data collected by the sensors was assessed by comparing the sensor readings with values obtained from calibrated reference instruments. The accuracy for each sensor was calculated using the formula:

$$\text{Accuracy (\%)} = \left(1 - \frac{|\text{Reference Reading} - \text{Sensor Reading}|}{\text{Reference Reading}}\right) \times 100$$

This formula quantifies the percentage accuracy by evaluating the deviation of sensor readings from the reference values. Based on this methodology, the following accuracy rates were observed for the sensors:

- Sound Sensor: 98%
- IR Sensor: 97%
- Temperature Sensor: 99%
- Humidity Sensor: 98%
- Vibration Sensor: 97%

These high accuracy rates highlight the reliability and precision of the sensors, ensuring that they deliver consistent and accurate measurements. Such performance is essential for maintaining the effectiveness of home automation systems, where precise data is critical for making informed decisions and adjustments in real time.

3. System Reliability

The system's reliability was tested by running continuous operations over 30 days. The MQTT-based communication and PostgreSQL data storage were monitored for uptime and integrity. The results show an uptime of 99.9%, with no significant data loss or corruption.

- MQTT Broker Uptime: 99.9%
- PostgreSQL Database Uptime: 99.9%
- Data Integrity: 100%

These results confirm that the system is highly reliable and can maintain consistent operations over extended periods.

4. Real-World Performance and Potential Limitations

While the system has demonstrated strong performance in a controlled laboratory environment, its effectiveness in real-world conditions may be influenced by factors such as environmental variability, network reliability, power constraints, and long-term stability. This section explores how these elements could impact the system's functionality and suggests potential strategies for maintaining optimal performance.

1. Sensor Accuracy in Different Environments:

Sensor accuracy can vary significantly depending on the deployment environment. For instance, temperature and humidity sensors may experience fluctuations in accuracy under rapidly changing conditions, as seen in spaces with frequent indoor-outdoor transitions. Similarly, environmental factors such as dust, humidity, and electromagnetic interference could impact the performance of sound and vibration sensors, particularly in noisy or industrial settings. To address these challenges, regular sensor calibration and environmental adjustments may be required. Incorporating self-calibration algorithms into the system could help sensors adapt dynamically to shifting environmental conditions, enhancing accuracy and reliability across diverse settings.

2. Handling Network Outages:

Ensuring consistent performance during network outages is a critical consideration for real-world deployments. Since MQTT relies on continuous communication between clients (sensors) and the broker, any disruption in network connectivity could interrupt data transmission. To mitigate this, implementing MQTT's message persistence and Quality of Service (QoS) features can enhance data reliability. Using QoS levels 1 or 2 allows messages to be stored temporarily on the client and broker until successful delivery, reducing the risk of data loss. Furthermore, local data caching on devices like the ESP8266 microcontroller can store sensor readings during outages. This can be transmitted when the network connection is re-established, ensuring continuous data availability and system resilience.

3. Long-Term Stability:

Maintaining stability over extended deployments is essential for any automation system. Factors like hardware wear, sensor degradation, and software issues can impact long-term reliability. To address these potential challenges, periodic maintenance and firmware updates are recommended. Regularly cleaning and recalibrating sensors can prolong their lifespan and preserve accuracy. On the software side, monitoring tools that track real-time system performance can help detect and resolve potential

issues before they cause disruptions. Additionally, implementing automatic fault detection and recovery mechanisms allows the system to operate continuously with minimal manual intervention, making it more resilient over the long term.

4. Impact of Latency on Real-Time Applications:

In larger spaces, such as extensive smart homes or industrial sites, latency may increase due to longer distances between devices and the broker or due to network congestion. Although the system demonstrated low response times in a controlled setting, it may encounter delays in real-world applications where high latency is unavoidable. For latency-sensitive applications, such as security monitoring, edge computing solutions may be required. Processing critical data locally at the edge, before transferring it to the cloud, can reduce dependency on long-distance communication and ensure timely responses. This approach enhances the system's responsiveness, especially in scenarios demanding immediate action.

5. System Adaptability and Scalability in Dynamic Environments:

As home and industrial environments evolve, the need for scalability and adaptability becomes more apparent. New devices and sensors may be added, or existing components may need to be upgraded. While the current system architecture is designed to accommodate additional sensors, real-world deployments may face challenges such as increased network traffic, bandwidth limitations, and power consumption. To ensure scalability, the system can benefit from optimizing resource use, such as bandwidth and processing power, and potentially implementing energy-efficient communication protocols. This adaptability enables the system to support expanding automation needs without compromising performance.

6. Energy Efficiency:

Although MQTT's low bandwidth usage inherently reduces energy consumption, the overall power usage can increase in large-scale, real-world deployments where multiple devices operate continuously. Implementing power-saving techniques, such as sleep modes for sensors not actively in use, can significantly reduce energy consumption. Using energy-efficient hardware components further minimizes the system's energy footprint. Additionally, periodic energy audits can help identify and address power inefficiencies, ensuring the system remains cost-effective and sustainable, particularly in larger installations.

By addressing these potential real-world challenges through adaptive design and robust protocols, the MQTT-based home automation system can maintain high performance, accuracy, and reliability in various real-world settings, ultimately providing a more resilient solution for smart homes and industrial environments.

Discussion

The results indicate that the MQTT-based home automation system performs well across various performance metrics, including response time, data accuracy, system reliability, and scalability, highlighting its suitability for real-time monitoring and control in smart homes. The following points summarize the key findings and implications:

1. Efficiency of MQTT Protocol

The MQTT protocol has proven highly efficient for this home automation application, particularly in facilitating real-time device communication. Its lightweight nature and low bandwidth requirements ensure swift and reliable data transmission between the ESP8266 microcontroller and the MQTT broker. This efficiency is evidenced by the low response times observed during tests, where sensor-triggered actions were performed promptly, ensuring effective automation.

2. Scalability

Scalability is a critical aspect of the MQTT-based home automation system, enabling it to support the integration of additional sensors and handle increasing volumes of data in real-world applications. The system's architecture, leveraging MQTT for communication and PostgreSQL for data storage, has been designed to maintain performance with minimal degradation as the network grows. Future scalability considerations will become increasingly important as advanced technologies, such as Artificial Intelligence (AI) for threat detection and blockchain for secure data handling, are integrated into the system, enabling it to support more complex and data-intensive operations [13].

With the addition of AI for real-time threat detection, the system must efficiently process and analyze larger volumes of data. Techniques such as edge computing can help by allowing preliminary data processing locally on the microcontroller or gateway devices, reducing the data load transmitted to the central server or cloud. This approach decreases latency and enhances the system's responsiveness, which is especially important for applications requiring immediate action, such as security monitoring. By processing data close to the source, edge computing minimizes bandwidth usage, allowing the system to scale while maintaining network efficiency.

Similarly, integrating blockchain technology for enhanced security and data integrity introduces additional data verification

and storage layers. Blockchain's inherently decentralized structure can support scalability by distributing transaction processing across multiple nodes, reducing the strain on any single component. However, blockchain integration also comes with challenges, particularly in terms of latency and transaction costs, as each transaction requires validation across the network. To address these issues, lightweight blockchain solutions or off-chain processing techniques can be employed. For example, non-critical data can be processed off-chain and later appended to the blockchain, conserving resources while maintaining a verifiable data record.

As the system scales to accommodate more devices, load balancing across multiple MQTT brokers may be necessary to ensure a stable, uninterrupted data flow. Distributing data traffic among multiple brokers helps prevent bottlenecks and improve fault tolerance, as the system can reroute traffic to functional brokers if one fails. This load balancing approach enhances system resilience and ensures that high volumes of data can be managed without impacting overall performance.

Additionally, the system's scalability depends on optimizing resource allocation, including network bandwidth, processing power, and energy consumption. Adaptive resource management techniques can dynamically adjust these resources based on real-time demand, ensuring efficient use even as the number of connected devices grows. For instance, during periods of low activity, the system can reduce bandwidth allocation and put certain sensors into low-power mode to conserve energy.

In summary, the scalability of the MQTT-based home automation system is achieved through strategic integration of edge computing, lightweight blockchain protocols, load balancing, and adaptive resource management. These techniques enable the system to scale efficiently, handling increased data volume and complexity while ensuring reliability and performance across large and dynamic environments.

3. Data Accuracy and Reliability

The system's high sensor accuracy, efficient MQTT data handling, and reliable PostgreSQL storage collectively ensure trustworthy performance for critical home automation tasks. The sensors provide accurate real-time data on environmental conditions, such as temperature, humidity, motion, sound, and vibration, enabling the system to monitor changes effectively. This precision is crucial for maintaining optimal temperature levels through air conditioning adjustments, triggering lights based on motion detection, and monitoring unusual vibrations to enhance security.

The MQTT broker facilitates swift and reliable communication between sensors and the central processing unit, ensuring that data is transmitted with minimal delay. This efficient data handling is essential for real-time applications, where quick responses to detected changes, such as turning on lights or alerting occupants, are critical.

The PostgreSQL database plays a key role in maintaining the reliability and integrity of stored data. It ensures that historical sensor data is preserved accurately, enabling the system to analyze patterns and make informed decisions. For example, the system can adjust lighting schedules based on past usage or optimize temperature settings based on recorded environmental trends.

Together, these components enable seamless and precise environmental adjustments, improving the comfort, security, and convenience of daily life in a smart home. This integration ensures the system's robustness and reliability, making it well-suited for real-world home automation applications.

4. Practical Implications

The practical implications of this system are significant. By providing reliable, real-time monitoring and control, the system enhances home security, optimizes energy usage, and improves overall comfort. The ability to integrate various sensors and automate responses to environmental changes demonstrates the potential for widespread adoption in smart homes.

5. Economic Feasibility and Cost Analysis

The economic feasibility of the MQTT-based home automation system is a critical factor influencing its potential for widespread adoption. This section presents a comprehensive cost analysis of the system, covering hardware, software, installation, and maintenance, along with a summary of its affordability and scalability.

Hardware Costs: The system is built around the ESP8266 microcontroller, supported by various sensors, including IR, temperature, DHT11, sound, and vibration sensors. These components are affordable and readily available for IoT applications. For a typical home setup with multiple sensors and microcontrollers, the overall hardware cost remains low, making the system economically accessible.

Software Costs: The MQTT protocol and PostgreSQL database are open-source solutions, eliminating direct software licensing costs. However, additional costs may arise if advanced features, such as AI-based threat detection or blockchain integration, are implemented. These expenses, primarily associated with software development and integration, are one-time investments that enhance the system's capabilities.

Installation Costs: Installation costs depend on the complexity of the setup and the number of devices involved. A simple configuration, involving sensor placement and connecting them to the ESP8266 microcontroller, incurs minimal labor costs. For larger, more complex installations requiring advanced configurations, professional services may be needed, resulting in slightly higher costs.

Maintenance Costs: Maintenance activities include periodic sensor calibration, firmware updates for the ESP8266 microcontroller, and network reliability checks. Due to the durability of the system components, maintenance costs are low and typically involve occasional sensor replacements and routine system checks. These activities ensure the system's long-term stability and performance.

Cost Breakdown: The table below provides a rough cost estimate for the hardware, software, installation, and maintenance components of the system:

Table 7 Cost Breakdown of the MQTT-Based Home Automation System

Category	Component	Estimated Cost (USD)	Remarks
Hardware	ESP8266 Microcontroller	5	Central processing unit with built-in Wi-Fi.
	IR Sensor	3	Motion detection.
	Temperature Sensor	2	Measures ambient temperature.
	DHT11 Sensor	3	Provides humidity and temperature readings.
	Sound Sensor	2	Detects sound levels.
	Vibration Sensor	4	Monitors vibrations for security purposes.
	Power Supply Units	10	Powers the sensors and microcontroller.
Software	MQTT Broker (e.g., Mosquito)	Free	Open-source software for message communication.
	PostgreSQL Database	Free	Open-source database for data storage.
Installation	System Assembly and Configuration	50	Includes labor for wiring and sensor placement.
Maintenance	Sensor Calibration and Testing	20/year	Annual calibration for reliable performance.
	Software Updates	10/year	Regular updates for broker and database software.

Economic Feasibility: The MQTT-based home automation system demonstrates significant economic feasibility due to its low initial hardware costs and reliance on open-source software. Its scalability and minimal maintenance requirements make it an attractive and affordable solution for homeowners seeking to enhance security, optimize energy usage, and automate home functions. By providing high value at a relatively low cost, the system is poised for widespread adoption in the smart home market.

6. Conclusion

This study has demonstrated that the MQTT-based home automation system is efficient and reliable, with high accuracy across various performance metrics. The architecture, which integrates the ESP8266 microcontroller, a suite of sensors, the MQTT protocol, and PostgreSQL for data storage, has proven to be a robust solution for modern smart homes. The system's ability to maintain low response times, high data accuracy, and exceptional reliability underscores its potential for real-world deployment. By enhancing security, optimizing energy usage, and improving overall comfort, this system presents a compelling option for homeowners seeking advanced home automation solutions. The promising results pave the way for further development and integration, potentially expanding its applications and enhancing its capabilities in future iterations.

Building on the foundation established in this work, the next phases of the research will focus on enhancing the security of the MQTT protocol against potential spoofing attacks. The planned future work includes:

1. **Simulating Attacks on MQTT:** will focus on assessing the protocol's vulnerabilities in IoT environments, with particular attention to spoofing attacks. By simulating various types of attacks, the study aims to identify weaknesses and evaluate the robustness of the system under potential threats. These simulations will help highlight areas where additional security measures, such as advanced encryption or authentication protocols, may be required.
2. **Integration of AI for Threat Detection:** Artificial Intelligence (AI) will be integrated to develop advanced threat detection mechanisms.

Machine learning algorithms, particularly supervised learning models such as Random Forests, Support Vector Machines (SVM), and Neural Networks, will be employed for threat detection. These models are well-suited for identifying patterns in large datasets and detecting anomalies. This is crucial for recognizing various types of cyber-attacks, including spoofing, Denial of Service (DoS), and Man-in-the-Middle (MitM) attacks.

Training the AI Algorithms: The algorithms will be trained using labeled datasets containing both normal and malicious MQTT traffic patterns. Open-source IoT datasets, such as the "IoT-23" dataset, which includes various network traffic logs with different attack scenarios, will be used as the primary source for training. In addition, we plan to generate synthetic data by simulating attacks on the MQTT system in a controlled environment to ensure the training data is comprehensive and covers a wide range of attack types. The datasets will include normal communication flows between IoT devices and brokers, as well as logs from spoofing attempts, DoS attacks, and data manipulation incidents. This data will help the models learn to distinguish between legitimate and malicious activities.

Handling Real-Time Threat Detection: To enable real-time threat detection, a combination of feature extraction techniques and sliding window analysis will be used. By continuously analyzing incoming MQTT traffic in short time windows, the AI algorithms can detect deviations from normal behavior patterns. Features such as message frequency, payload size, source IP addresses, and unusual topic subscriptions will be monitored for signs of an attack. When anomalies are detected, the system will trigger an alert and take predefined actions, such as blocking the offending client or redirecting traffic through a more secure route.

In addition to supervised learning methods, unsupervised learning algorithms like K-Means clustering or Autoencoders will detect previously unseen or zero-day attacks. These algorithms can identify outliers in the data without requiring labeled datasets, making them useful for detecting novel attack vectors. Combining supervised and unsupervised approaches will enhance the system's ability to detect various threats.

By using these AI models, the MQTT-based home automation system will be able to autonomously monitor and analyze network traffic, identify potential threats in real-time, and respond swiftly to mitigate security risks. This integration of AI will create a dynamic and adaptive security mechanism that evolves as new attack vectors emerge.

3. **Blockchain for Enhanced Security:** While blockchain technology offers significant advantages in ensuring data integrity and security, its integration into IoT systems like MQTT-based home automation can pose challenges, particularly in terms of scalability, latency, and transaction costs. Traditional blockchain systems, such as those used in cryptocurrency networks, often suffer from scalability limitations due to their reliance on consensus mechanisms like Proof of Work (PoW), which can result in slower transaction processing times and increased computational overhead. Latency can also be an issue in blockchain systems, as the time it takes to validate transactions and add them to the distributed ledger can introduce delays, which are not ideal in real-time IoT environments. Transaction costs, another concern, arise from the fees associated with processing each blockchain transaction, which can become prohibitive as the system scales.

   To mitigate these challenges, we plan to integrate lightweight blockchain technologies that are specifically designed for IoT applications. These include blockchains that utilize more efficient consensus algorithms, such as Proof of Stake (PoS) or Practical Byzantine Fault Tolerance (PBFT), which significantly reduce the computational burden and improve transaction throughput. Furthermore, the use of off-chain scaling solutions, such as sidechains or state channels, can help alleviate the load on the main blockchain by allowing multiple transactions to occur off-chain before being finalized and recorded on the main ledger. This approach reduces both latency and transaction costs, ensuring that the system can handle a higher volume of transactions in a timely and cost-effective manner.

   Additionally, optimized transaction handling methods, such as batching transactions or only recording critical data on the blockchain, will be employed to further improve efficiency. By carefully selecting which data needs to be immutably stored on the blockchain and which can be handled off-chain or through traditional databases, we can strike a balance between security, performance, and cost. This approach will allow the system to maintain the benefits of blockchain's decentralized and secure nature while minimizing the associated limitations.

   By implementing these solutions, the integration of blockchain in the MQTT-based home automation system will remain scalable, efficient, and cost-effective, ensuring that the system can securely process a high volume of transactions without compromising performance.

4. **Developing a Comprehensive Security Framework:** A comprehensive security framework combining AI and Blockchain technologies will be developed to secure the MQTT protocol from spoofing attacks. This framework will offer real-time threat detection and mitigation, enhancing the overall security of IoT systems.

By addressing these areas in future work, we aim to create a more secure and resilient IoT ecosystem, ensuring the reliability and safety of connected devices and networks. This research will contribute to advancing IoT security standards and promoting the wider adoption of IoT technologies across various sectors.

References

[1] Institute of Electrical and Electronics Engineers, *2019 54th International Universities Power Engineering Conference (UPEC) : proceedings : 3-6 September 2019, Bucharest, Romania.*

[2] M. F. Usmani, "MQTT Protocol for the IoT." International Journal of Internet of Things, vol. 12, no. 3, pp. 45-50, September 2020. DOI: 10.1000/xyz123.

[3] IEEE Communications Society and Institute of Electrical and Electronics Engineers, *2018 10th International Conference on Communication Systems & Networks (COMSNETS) : 3-7 Jan. 2018.*

[4] Saveetha Engineering College and Institute of Electrical and Electronics Engineers, *IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI) - 2017 : 21st & 22nd September 2017*.

[5] B. N. Alhasnawi, B. H. Jasim, Z. A. S. A. Rahman, and P. Siano, "A novel robust smart energy management and demand reduction for smart homes based on internet of energy," *Sensors*, vol. 21, no. 14, Jul. 2021, doi: 10.3390/s21144756.

[6] M. Esposito, A. Belli, L. Palma, and P. Pierleoni, "Design and Implementation of a Framework for Smart Home Automation Based on Cellular IoT, MQTT, and Serverless Functions," *Sensors*, vol. 23, no. 9, May 2023, doi: 10.3390/s23094459.

[7] V. Seoane, C. Garcia-Rubio, F. Almenares, and C. Campo, "Performance evaluation of CoAP and MQTT with security support for IoT environments," *Computer Networks*, vol. 197, Oct. 2021, doi: 10.1016/j.comnet.2021.108338.

[8] T. Magara and Y. Zhou, "Internet of Things (IoT) of Smart Homes: Privacy and Security," *Journal of Electrical and Computer Engineering*, vol. 2024, 2024, doi: 10.1155/2024/7716956.

[9] C. B. Gemirter, Ş. Çenturca, and Ş. Baydere, "A Comparative Evaluation of AMQP, MQTT and HTTP Protocols Using Real-Time Public Smart City Data," in *Proceedings - 6th International Conference on Computer Science and Engineering, UBMK 2021*, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 542–547. doi: 10.1109/UBMK52708.2021.9559032.

[10] M. Singh, M. A. Rajan, V. L. Shivraj, and P. Balamuralidhar, "Secure MQTT for Internet of Things (IoT)," in *Proceedings - 2015 5th International Conference on Communication Systems and Network Technologies, CSNT 2015*, Institute of Electrical and Electronics Engineers Inc., Sep. 2015, pp. 746–751. doi: 10.1109/CSNT.2015.16.

[11] M. A. Khan *et al.*, "A deep learning-based intrusion detection system for mqtt enabled iot," *Sensors*, vol. 21, no. 21. MDPI, Nov. 01, 2021. doi: 10.3390/s21217016.

[12] V. Kumar, N. Malik, J. Singla, N. Z. Jhanjhi, F. Amsaad, and A. Razaque, "Light Weight Authentication Scheme for Smart Home IoT Devices," *Cryptography*, vol. 6, no. 3, Sep. 2022, doi: 10.3390/cryptography6030037.

[13] S. Shapsough, M. Takrouri, R. Dhaouadi, and I. Zualkernan, "An MQTT-Based Scalable Architecture for Remote Monitoring and Control of Large-Scale Solar Photovoltaic Systems," in *Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, LNICST*, Springer Verlag, 2019, pp. 57–67. doi: 10.1007/978-3-030-05928-6_6.

[14] IEEE Communications Society. Internet of Things Emerging Technologies Initiatives, IEEE Computational Intelligence Society, Institute of Electrical and Electronics Engineers, and S. Internet of Things Week (2017 : Geneva, *GIoTS2017 : Global Internet of Things Summit : 2017 proceedings papers : CICG, Geneva, June 6-9, 2017.*

[15] F. Morgado-Dias, F. Quintal, Madeira Interactive Technologies Institute, Universidade da Madeira, Institute of Knowledge and Development, and Institute of Electrical and Electronics Engineers, *Energy and Sustainability in Small Developing Economies 2018 : July 9-12, 2018, Madeira - Portugal*.

[16] I. Froiz-Míguez, T. M. Fernández-Caramés, P. Fraga-Lamas, and L. Castedo, "Design, implementation and practical evaluation of an iot home automation system for fog computing applications based on MQTT and ZigBee-WiFi sensor nodes," *Sensors (Switzerland)*, vol. 18, no. 8, Aug. 2018, doi: 10.3390/s18082660.

Article Information Form:

Conflict of Interest Notice

The authors declare that there is no conflict of interest regarding the publication of this paper.

Ethical Approval and Informed Consent
It is declared that during the preparation process of this study, scientific and ethical principles were followed, and all the studies benefited from are stated in the bibliography.

Availability of data and material
Not applicable.

Plagiarism Statement
This article has been scanned by iThenticate ™.

SAKARYA UNIVERSITY JOURNAL OF COMPUTER AND INFORMATION SCIENCES
http://saucis.sakarya.edu.tr/
e-ISSN: 2636-8129
Publisher: Sakarya University
Vol. 7, No. 3, 2024, 510-520
DOI: 10.35377/saucis...1483985
SAUCIS
SAKARYA UNIVERSITY

RESEARCH ARTICLE

Optimal Allocation and Sizing of Multiple DGs with Reactive Power Capabilities in a Three-Phase Unbalanced Distribution System

Zahira Aboumaria[1, *], Selçuk Emiroğlu[2]

[1] Sakarya University, Institute of Natural Sciences, Dept of Electrical-Electronics Eng., Sakarya, Türkiye
[2] Sakarya University, Dept of Electrical-Electronics Engineering, Sakarya, Türkiye

Corresponding author:
Zahira Aboumaria, Sakarya University, Institute of Natural Sciences, Dept of Elec. Electronics Eng., Sakarya, Türkiye
zahira.aboumaria@ogr.sakarya.edu.tr

Article History:
Received: 14.05.2024
Accepted: 09.08.2024
Published Online: 31.12.2024

ABSTRACT
Nowadays, Distributed Generators (DGs) are widely adopted in distribution networks to deliver fast, reliable, and clean power to the consumer maximize environmental preservation, and mitigate the impact of energy production on the environment. However, recurring issues like poor voltage profiling/stability and power loss arising from improper allocation and unsuitable sizing of the DGs have made it necessary for methods and approaches to be sought in order to mitigate these issues. This study proposes a method that can be used in optimizing the allocation and sizes of the DGs. The study employs the IEEE 37 node test system in OpenDSS to carry out power flow. The DG size, node, and power factor are the coordinated control variables presented in this study to minimize the power loss. Genetic Algorithm, Pattern Search, Particle Swarm Optimization, and Grey Wolf Optimizer algorithms have been exploited in the IEEE 37 node test feeder to find the optimal location, sizes, and power factors of the DGs. Notable variations resulting from four different cases considering power loss as an objective function are also presented. Results indicate that optimally sized and placed DGs operated with optimal power factors have reduced power losses by enhancing the voltage profile. In addition, the effect of the reactive power capability of DGs on the distribution system has been shown.

Keywords: Unbalanced distributed network, IEEE 37 node test feeder, Distributed generation, Genetic algorithm, Power loss, Optimization

1. Introduction

Distributed Generation (DG) systems have a crucial role in the minimization of power losses, which occur during the transmission of electricity over long distances from power plants to end-users. Power losses are minimized when electricity travels vast distances from power plants to users using distributed generation (DG) systems. They accomplish this by generating power close to the site of consumption, hence lowering transmission losses and relieving system strain. Because DG systems are often more efficient than central power plants, they improve the overall system efficiency by lowering energy consumption and costs [1]. Distributed applications provide significant advantages, generate electricity for local injections, and interact with low-voltage transformers. Incorporating DG can reduce transmission line losses, increase grid resiliency, minimize additional generating costs, and reduce the need to invest in up-to-date utility generation capacity [2].

Electric utility systems seeking the development of energy by distributed PV allocation can reap a variety of advantages as well as offer backup in the worst-case scenario of disruption with correct calibration [2, 3]. The rising complexity of power distribution networks has fueled the demand for optimizing power systems through efficient and dependable solutions. Load flow studies must deal with a wide range of system configurations properly and rapidly, making them an essential analysis of power systems. The distribution systems are frequently unbalanced due to single-phase, two-phase, or three-phase loads while employing a radial system to generate power [4, 5].

The positioning and sizes of DGs have a significant impact on [6]: Voltage Regulation meaning at the point of use, DGs stabilize voltage, decreasing voltage variations' losses. Control of Reactive Power: DGs offer reactive power, which reduces how much electricity is needed for compensation equipment and power losses. Locating DGs in high-demand areas reduces transmission losses [7, 8]. Sizing DGs correctly enables effective functioning [9, 10, 11]. Various strategies, including heuristic algorithms, are used to solve the optimization challenge for DG systems [10]. MATLAB [12] is capable of running an optimization algorithm, which includes a Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Grey Wolf Optimizer (GWO), and Pattern Search (PS), to optimize the placement, sizing, and power factor of DG units in the system.

Cite as: Z. Aboumaria and S. Emiroğlu, "Optimal allocation and sizing of multiple DGs with reactive power capabilities in a three-phase unbalanced distribution System", *Sakarya University Journal of Computer and Information Sciences*, vol.7, no. 3, pp. 510-520, 2024. Doi: 10.35377/saucis…1483985

PSO Algorithms are a clever way to explore and improve optimization [13] within a search area. They are inspired by the social behavior found in regulated colonies. Using a group of people known as a "swarm," this program explores potentially interesting regions in the search space. We refer to these people as "particles" or "agents.". GWO algorithm optimizes solutions based on wolf pack behavior [14]. GAs are evolutionary algorithms that efficiently optimize DG systems, especially for difficult issues [15-17]. These algorithms have reduced losses by ensuring the voltage is within the defined limits. The results obtained from the algorithms are compared.

Many studies have applied optimization techniques like GA, PSO, PS, and GWO to solve power system problems. Typically, these studies focus on transmission systems or balanced distribution networks, and they often assume that distributed generators (DGs) operate at a unity power factor. However, real-world distribution systems exhibit imbalances due to uneven loading and line characteristics. In this research, we concentrated on the optimal DGs placement in unbalanced distribution systems, considering both unity power factor and optimal power factor scenarios. Our simulations revealed that optimizing the power factor of DGs can significantly lower system losses by leveraging their reactive power capabilities.

The paper is organized as follows Section 2, explains the objective function while satisfying all constraints, Section 3, describes the test system, and Section 4, there's explanation of the algorithms used in the study. Section 5 gives the simulation results, and the conclusion is given.

2. Problem Formulation

For the IEEE 37 node test system, the ideal amount and size of DG units to be deployed to cut down on system power loss and get the proper voltage profile are four. System power loss must therefore be expressed as a function of system bus voltages and DG capacity. The current load is steady, and the section load is evenly distributed [17].

Improving power system efficiency by reducing operating costs and eliminating energy losses is one benefit of optimizing power flow within power systems, among other benefits. Additionally, enhancing voltage stability strengthens the power system's dependability and stability [17-22]. Therefore, to employ an optimization algorithm, the optimization issue, including the objective function and constraints, must first be established [22].

2.1. Objective Function

Since power loss has directly affected system efficiency and economy, minimization of the power loss has been selected as an objective function. Minimizing the power losses enhances the power distribution system's performance and dependability, which lowers operating costs and improves energy efficiency [23].
The problem's goal function has been established as lowering the overall actual power loss. The objective function is:

$$min\ P_{loss} = \ min\left[\sum_{k=1}^{T} g_k\{V_i^2 + V_j^2 - 2V_iV_j\cos(\delta_i - \delta_j)\}\right] \tag{1}$$

where, g_k is the conductance at line k between i and j nodes, V_i, V_j and δ_i, δ_j are the voltage magnitudes of the node and angles of nodes i and j. T is the number of the line.

2.2. Constraint

The following are the constraints of the problem in the optimization process as given below.

2.2.1. Voltage Limitation

The following constraint is described to ensure that the voltage of any node remains within defined limits [22]:

$$V_{min,i} \ \leq V_i \ \leq V_{max,i}; \qquad i = 1, \dots, N_B \tag{2}$$

2.2.2. Power Flow Equation

The balance of active and reactive power must be defined as an equality constraint of the optimization problem [23]:

$$P_{ss} + \sum_{i=1}^{N_{DG}} P_{DG,i} = \sum_{j=1}^{N_B} P_{Dj} + \sum_{K=1}^{N_L} P_{L,K} \tag{3}$$

$$Q_{ss} + \sum_{i=1}^{N_{DG}} Q_{DG,i} = \sum_{j=1}^{N_B} Q_{Dj} + \sum_{K=1}^{N_L} Q_{L,K} \tag{4}$$

where, P_{ss}, $P_{DG,i}$, P_{Dj} and $P_{L,K}$ are the active power drawn from the substation, active power of the DGs, active power of load and active power loss, respectively. Also, Q_{ss}, $Q_{DG,i}$, $Q_{L,K}$ and Q_{Dj} are the reactive power drawn from the substation, reactive power of the DGs, reactive power loss and reactive power of the load, respectively.

2.2.3. The Total Active Power of DGs

The total active power output of multiple DGs installed in the test system should be less than or equal to the total real power of the loads.

$$\sum_{i=1}^{N_{DG}} P_{DG,i} \leq \sum_{j=1}^{N_B} P_{Dj} \tag{5}$$

2.2.4. DG Capacity Limit

DG capacity limit is considered within the maximum and minimum power as: [24]

$$P_{DG\,min,i} \leq P_{DG,i} \leq P_{DG\,max,i}\ ;\ i = 1, \dots, N_{DG} \tag{6}$$

2.2.5. DG Power Factor Limit

The DG power factor is considered an inequality constraint for DGs operating under different power factors except 1.

$$Pf_{DG\,min,i} \leq Pf_{DG,i} \leq Pf_{DG\,max,i}\ ;\ i = 1, \dots, N_{DG} \tag{7}$$

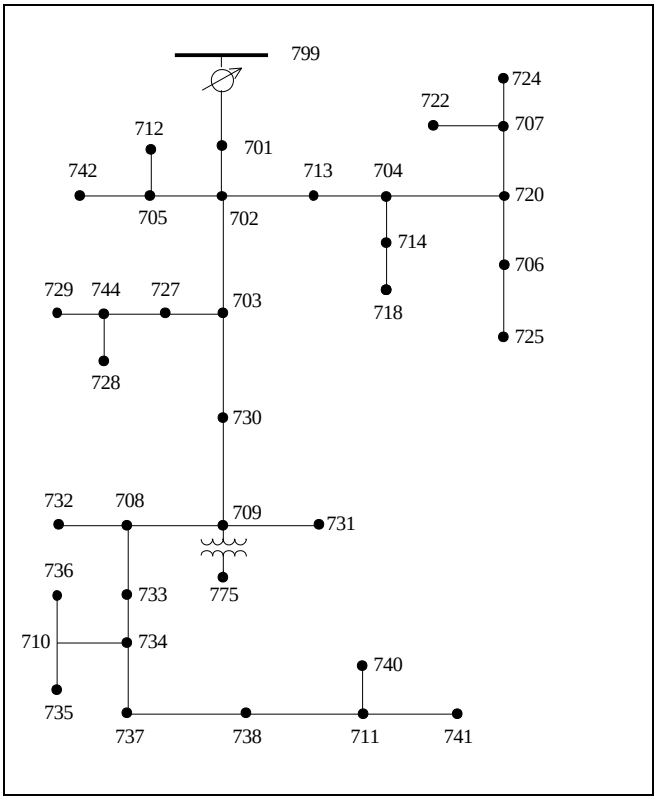


Figure 1. Single line diagram of IEEE 37 Test Feeder *[30]*

3. Unbalanced Radial Distribution System

The IEEE 37-node test system is a widely used distribution network for analyzing unbalanced radial distribution networks reflecting real-world complexities such as uneven loading and varied line characteristics [25, 26]. Our study used the IEEE 37 node distribution test system to model an unbalanced radial distribution network. This system allows us to rigorously assess how DGs can be optimally placed within an unbalanced distribution system [27, 28]. By examining both unit power factor and optimal power factor operated DGs, the simulations aimed to explore the potential for reducing system losses and enhancing overall performance.

This network is an unbalanced system with a medium voltage of 4.8kV, load size of 2547 KW and 1201 MVAr and consists of 37 nodes [29]. All data is given from [30, 31]. MATLAB [11] is employed to solve the optimization using GA, GWO, PS, and PSO to optimize the location, sizing, and power factor of DGs in the system. The IEEE 37 node unbalanced distribution system has adopted the concept. A comparison is made between the simulation results and those obtained using other methods. A single-line diagram of the IEEE 37-node test feeder has been displayed in Figure 1 [30]. Power flow simulations have been done using OpenDSS [32].

4. Optimization Algorithms

In our simulations, we employed several advanced optimization algorithms to enhance the distribution system's performance. GA, GWO, PS, and PSO have been used due to their proficiency in solving complicated problems to determine the place and size of DGs. [33, 34].

4.1. Genetic Algorithm

Evolutionary processes and natural selection mechanisms are the basis for Genetic Algorithms (GAs). They are used to look for almost-optimal answers to problems with search and optimization. The first step in the method is to create a population of candidate solutions represented as chromosomes at random. Then, a fitness function is used to evaluate these chromosomes to determine how well they address the given task. Based on these fitness evaluations, chromosomes are selected for reproduction, with the likelihood of selection proportional to their fitness scores. The selected chromosomes then undergo crossover, where segments of their genetic material are exchanged to produce new offspring. The mutation is applied to some offspring to introduce random variations and maintain genetic diversity. The newly created offspring and some of the best chromosomes from the current population form the next generation. This new population is then evaluated, and the selection cycle, crossover, and mutation go on until a termination criterion, such as the number of iterations or convergence tolerance, is met [35]. The process iterates to evolve and refine solutions to converge on the optimal or near-optimal answer [36]. The flowchart of the GA is shown in Fig. 2.

4.2. Particle Swarm Optimization

A computational technique called Particle Swarm Optimization (PSO) has been driven by the social interactions of fish and birds. Allowing a swarm of potential solutions, particles, to travel over the solution space can be utilized to find the best answers to issues [13].
Each particle of the swarm symbolizes a potential solution to the optimization problem. Firstly, the position and velocity of each particle are randomly initialized within the defined search space. The velocity of each particle has been used to update its position as in Eq. 8.

$$x_i^{new} = x_i^{old} + v_i \tag{8}$$

where v_i is the velocity of the i.th particle. Every particle's velocity is updated by its previous velocity, the distance to its personal best, and the global best.

$$v_i^{new} = w \cdot v_i^{old} + c_1 \cdot r_1 \cdot \left(Pbest_i - x_i^{old}\right) + c_2 \cdot r_2 \cdot \left(Gbest - x_i^{old}\right) \tag{9}$$

w, c_1 and c_2 are the inertia weight, cognitive and social coefficients, respectively. Also, random numbers r_1 and r_2 are in [0,1].

$$Pbest_i^{new} = \begin{cases} x_i^{new} & if\ f\left(X_i^{k+1}\right) < f\left(Pbest_i^{old}\right) \\ Pbest_i^{old} & otherwise \end{cases} \tag{10}$$

The global best is updated if any particle's personal best has a better fitness value [37].

$$Gbest^{new} = \begin{cases} Pbest_i^{new} & if\ f(Pbest_i^{new}) < f(Gbest^{old}) \\ Gbest^{old} & otherwise \end{cases} \tag{11}$$

The algorithm goes through the flowchart until a termination criterion is met, as shown in the flowchart of PSO in Fig. 3.

4.3. Pattern Search

Pattern Search (PS) is an optimization algorithm that iteratively refines solutions by exploring a structured search pattern. The process starts with an initial solution x_0 and evaluates the objective function $f(x)$ at this point. The algorithm updates the solution in each iteration using the following equations [38].

$$\begin{aligned} x^u &= x + h_i \hat{e}_i \\ x^d &= x - h_i \hat{e}_i \end{aligned} \tag{12}$$

where $\hat{e}_i$ is a unit vector in position i and step size is initially $h=x_{max}-x_{min}$.

The search then updates x to be the best of the three alternatives (x, x_u, and x_d) finding the

$$x = \arg\min\left(f(x), f(x^u), f(x^d)\right) \tag{13}$$

If no improvement is found, the step size is decreased, and the search pattern is adjusted. The flowchart starts with initializing the solution and step size, proceeds to evaluate the objective function, updates the solution, checks for improvement, adjusts the search pattern or step size, and iterates until a termination criterion is met as shown in Fig. 4. Fig. 4 shows the flowchart of the PS. This iterative refinement approaches the near-optimal or optimal solution by systematically exploring the search space [39].

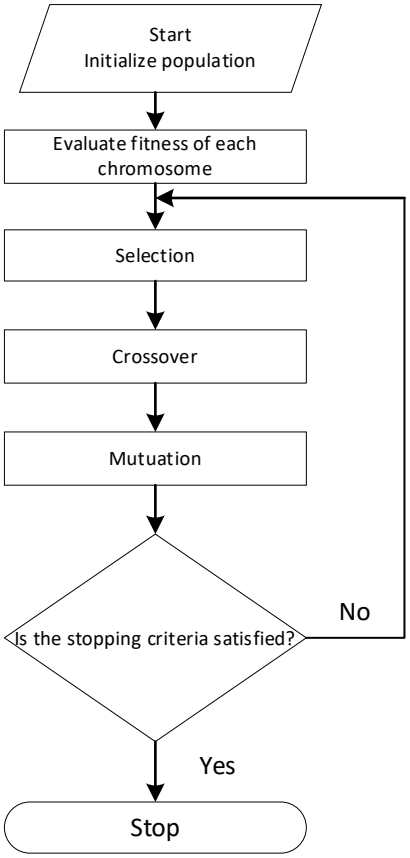


Figure 2. Flowchart of GA

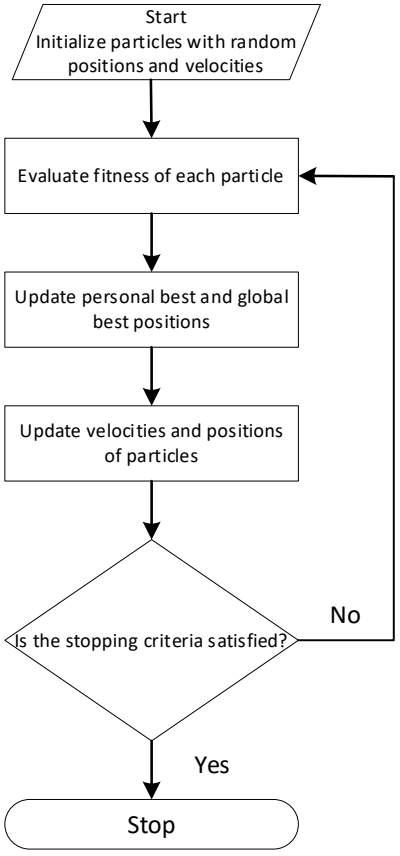


Figure 3. Flowchart of PSO

4.4. Grey Wolf Optimization

The Grey Wolf Optimization (GWO) algorithm is a nature-inspired optimization technique that simulates the hierarchical hunting mechanism of grey wolves [40]. The algorithm involves a population of candidate solutions, each represented as a grey wolf categorized into four groups: delta, beta, and alpha wolves, according to their fitness value. The position update of each wolf is guided by the positions of the delta, beta, and alpha wolves [40]. The GWO algorithm's flowchart is depicted in Fig. 5. The mathematical update rules are given by [40]:

$$\vec{D} = \left|\vec{C}.\overrightarrow{X_p}(t) - \vec{X}(t)\right| \tag{14}$$

$$\vec{X}(t+1) = \overrightarrow{X_p}(t) - \vec{A}.\vec{D}$$

where

$$\vec{A} = 2\vec{a}.\overrightarrow{r_1} - \vec{a}$$

$$\vec{C} = 2.\overrightarrow{r_2}$$

where t is the iteration number, $\vec{X}$ and $\overrightarrow{X_p}$ position and hunting position vector of wolves. A and C are coefficient vectors. $\vec{a}$ varies from 2 to 0, the random vectors $\overrightarrow{r_1}$ and $\overrightarrow{r_2}$ are in [0, 1] [41].

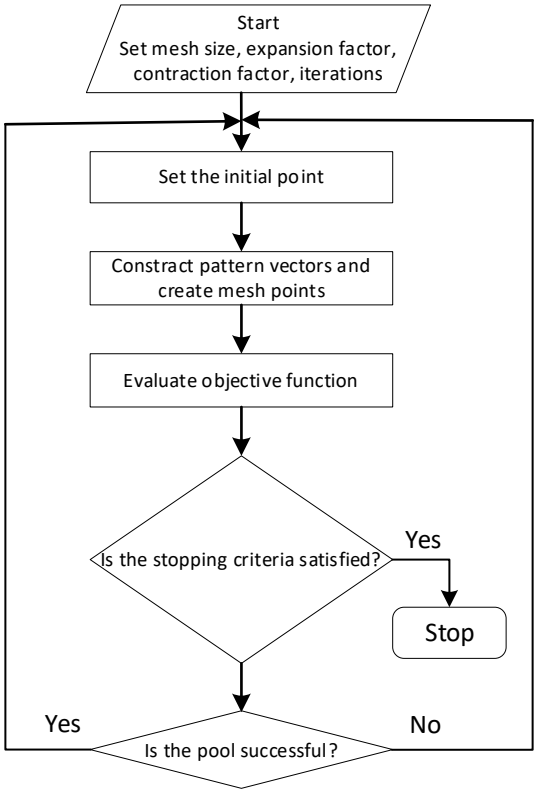


Figure 4. Flowchart of PS

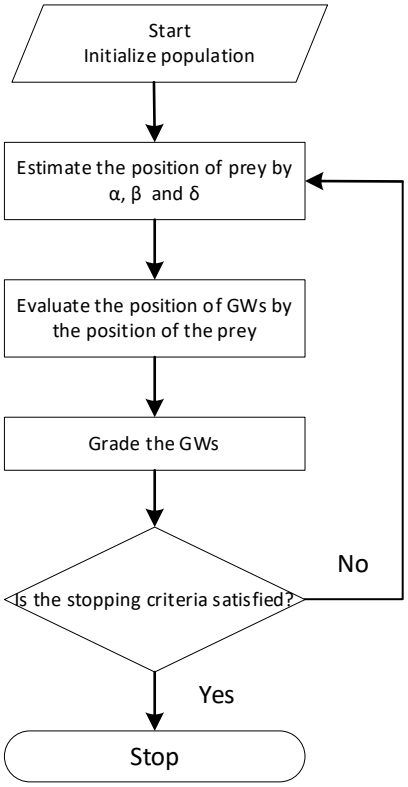


Figure 5. Flowchart of GWO

The position of optimal wolves can be calculated as follows [42].

$$\overrightarrow{D_\alpha} = \left|\overrightarrow{C_1}.\overrightarrow{X_\alpha} - \vec{X}\right|, \overrightarrow{D_\beta} = \left|\overrightarrow{C_2}.\overrightarrow{X_\beta} - \vec{X}\right|, \overrightarrow{D_\delta} = \left|\overrightarrow{C_3}.\overrightarrow{X_\delta} - \vec{X}\right| \quad (15)$$

$$\overrightarrow{X_1} = \overrightarrow{X_\alpha} - \overrightarrow{A_1}.\overrightarrow{D_\alpha}, \overrightarrow{X_2} = \overrightarrow{X_\beta} - \overrightarrow{A_1}.\overrightarrow{D_\beta}, \overrightarrow{X_3} = \overrightarrow{X_\delta} - \overrightarrow{A_1}.\overrightarrow{D_\delta}$$

$$\vec{X}(t+1) = \frac{\overrightarrow{X_1} + \overrightarrow{X_2} + \overrightarrow{X_3}}{3}$$

5. Simulation Results

The optimization algorithms have been evaluated and simulated on a computational system running Windows 11 with a processor of Intel Core i7 3770, 16 GB of RAM, and NVIDIA GeForce GT 640 Graphics. Also, the parameters of the algorithm used in the simulations are given in Table 1.

Table 1. The Control Parameters Used in the Simulation of Optimization Algorithms

Algorithm	**Parameter**	**Value**
GA	Population size, Elite count, Crossover, Scaling factor,	50, 0.05, 0.8, 0.4
PSO	Swarm size, Inertia Coefficient (w), Cognitive Coefficient (c_1), social Coefficient (c_2), Inertia Damping Weight	50, 1, 2, 2, 0.99
PS	Mesh Contraction, Mesh expansion	0.5, 2
GWO	Search agent	50

For the first case, all DGs are operated to unit power factor, four DGs produce only real power to the system, there is no reactive power generated by DGs, the four methods are applied such as GA, PSO, PS and GWO. The performance of GA for obtaining the optimal DG placement and the size are better than other algorithms in Table 2 in terms of power loss. In this case, two control variables are used for each DG for the optimization algorithm, so in total, we have 8 control variables for four DGs, which are the bus number and size of DGs. The maximum iteration was selected as 100 before the population number was 50. The maximum active power limit for DGs is 2547 kW. The total DG size is equal to or smaller than the total real load value because this is the optimization constraint it means to be satisfied for all algorithms. Before DG installation, In the base scenario, the lowest and highest voltages are 0.9664 pu and 1.0607 pu, respectively. For ideally sized and arranged DGs, the least and most voltages for the GA outcomes are 0.9915 pu and 1.0477 pu, respectively. Table 2 displays the node

voltages that have remained within the defined limits for all algorithms while the voltages are not within the limit before optimization. Also, the power losses are decreased after the optimization for all algorithms, as shown in Table 2.

The four DGs are installed as follows, for GA at nodes 737,703, 701, and 707, for PSO at nodes 736, 744, 727, and 718, for PS 734, 701,713 and 722, and finally at 737, 730, 701, 722 for GWO method as given in Table 2. In this case, the GA gives better results because the minimum loss value is 19.302 kW with a maximum loss reduction value of 73.85%. Table 3 shows the second case's most appropriate location, size, and power factor. In this case, DGs are operating with a lagging power factor to use their reactive power capabilities. Then, three control variables are used for each DG for the optimization algorithm, so in total, we have 12 control variables for four DGs, which are the bus number, size, and power factor of DGs.

Table 2. Various Approaches to Optimal DG Size and Position for an IEEE 37-node System Operating at a Unity Power Factor

Case	**Method**	**Installed DG**					**Total DG Power**	**Ploss (kW)**	**Loss Reduction (%)**	**Vmin**	**Vmax**
Base Case	-	Without DG	Total Real Load(kW)= 2457				-	73.81	-	0.9664	1.0607
			DG 1	**DG 2**	**DG 3**	**DG 4**					
4 DGs	**GA**	**Node**	737	703	701	707	2426.6	19.302	73.85%	0.9915	1.0477
		Size (kW)	637.94	853.35	584.06	351.24					
	PSO	**Node**	736	744	727	718	2354.4	19.693	73.32%	0.9916	1.0475
		Size (kW)	747.3	303.75	1012.7	290.64					
	PS	**Node**	734	701	713	722	2457	19.94	72.98%	0.9915	1.0478
		Size (kW)	863.78	659.63	663.71	269.88					
	GWO	**Node**	737	730	701	722	2455.8	19.354	73.78%	0.9903	1.0478
		Size (kW)	610.71	508.18	1052	284.91					

Table 3. Optimal DG Size and Location with Different Methods for IEEE 37 Node System at Optimal Power Factor

Case	Method	Installed DG					Total DG Power kW	Ploss (kW)	Loss Reduction (%)	Vmin	Vmax
Base Case	-	Without DG	Total Real Load(kW)= 2457				-	73.81	-	0.9664	1.0607
			DG 1	DG 2	DG 3	DG 4					
4 DGs	GA	Node	737	703	701	722	2447.3	7.0936	90.39%	0.9986	1.0363
		Size (kW)	656.27	743.66	791.74	255.6					
		Power factor	0.90521	0.906	0.85332	0.77254					
	PSO	Node	737	709	701	720	2455.1	7.148	90.32%	0.9968	1.0353
		Size (kW)	518.16	559.14	896.94	480.83					
		Power factor	0.9013	0.90984	0.88098	0.88838					
	PS	Node	738	733	701	704	2456.4	8.3903	88.63%	0.9968	1.0353
		Size (kW)	341.44	631.19	772.38	711.38					
		Power factor	0.6875	1	0.75	0.9375					
	GWO	Node	738	733	702	722	2383.52	7.3664	90.02%	0.9992	1.036
		Size (kW)	435.93	432.84	1267	247.75					
		Power factor	0.92891	0.87854	0.91204	0.8546					

The power losses also decreased with the optimization of all algorithms, as given in Table 3. The four DGs are installed as follows, for GA at nodes 737,703, 701, and 722, for PSO at nodes 737, 709, 701, and 720, for PS 738, 733,701 and 704, and finally at 738, 733, 702, 722 for GWO method as given in Table 3. In this case, the GA gives better results with a minimum loss value of 7.0936 kW and a maximum loss reduction value of 90.39%.

When comparing the results obtained with GA for two scenarios representing DG operating at unity power factor and different power factors, the power loss reduction in the first scenario was 73.85%, while in the second scenario, it was 90.39%. This indicates that DGs with reactive power capability significantly reduced power losses more in the second scenario. Thus, utilizing DGs with reactive power capabilities led to a decrease in total power losses.

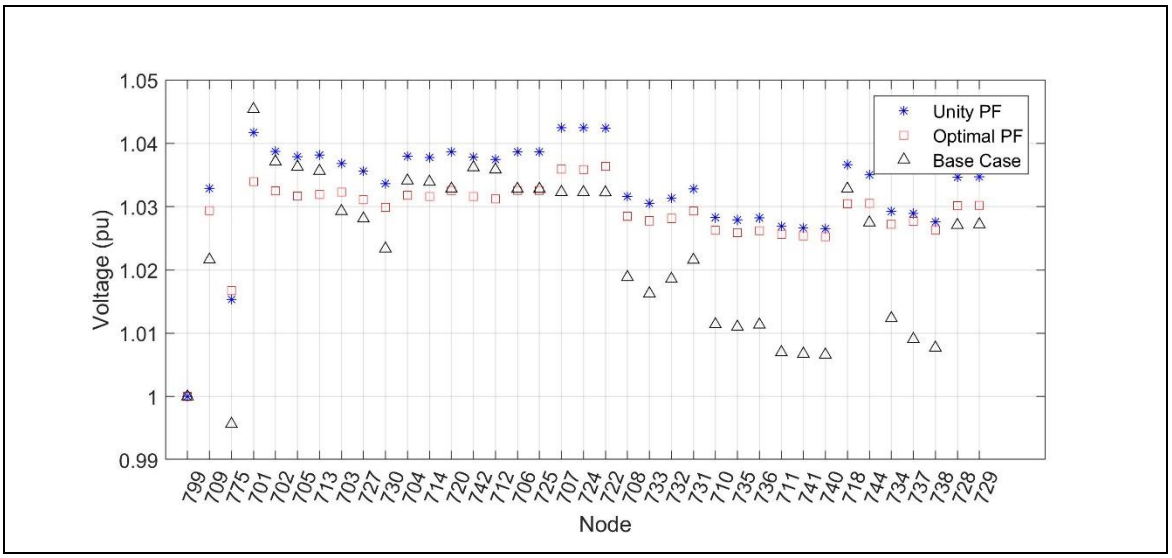


Figure 6. Voltage Profile of Phase for the Base Case, the Cases DGs Operated at Unity and Optimal Power Factor

The voltage profile was boosted and real power loss was significantly reduced following the insertion of DG units in the system as shown in Figures 6, 7 and 8. The constraints are satisfied as given; the GA gives the finding as improving power loss diminution outcomes in this scenario, with a 90.39% power loss reduction and the lowest voltage value of 0.99 pu and the highest voltage value of 1.03pu, respectively.

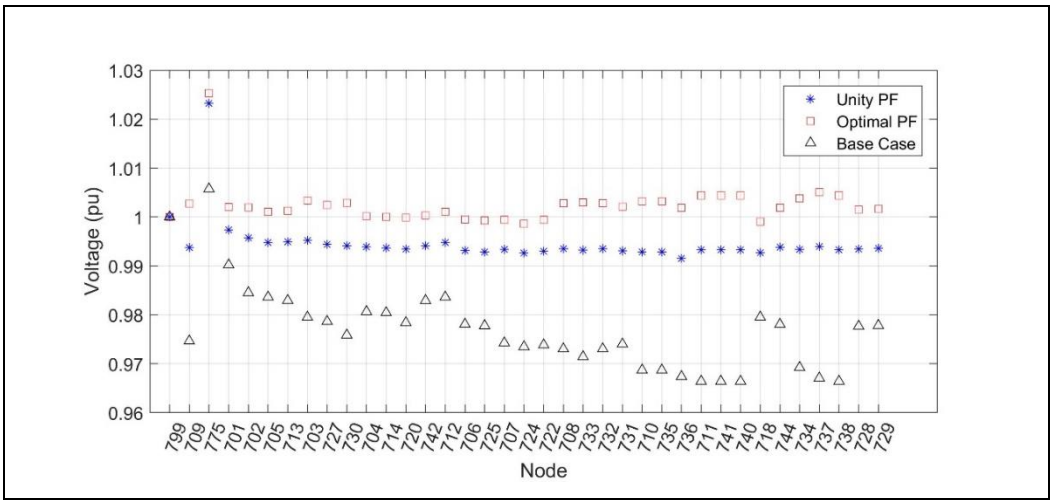


Figure 7. Voltage Profile of Phase b for the Base Case, the Cases DGs Operated at Unity and Optimal Power Factor

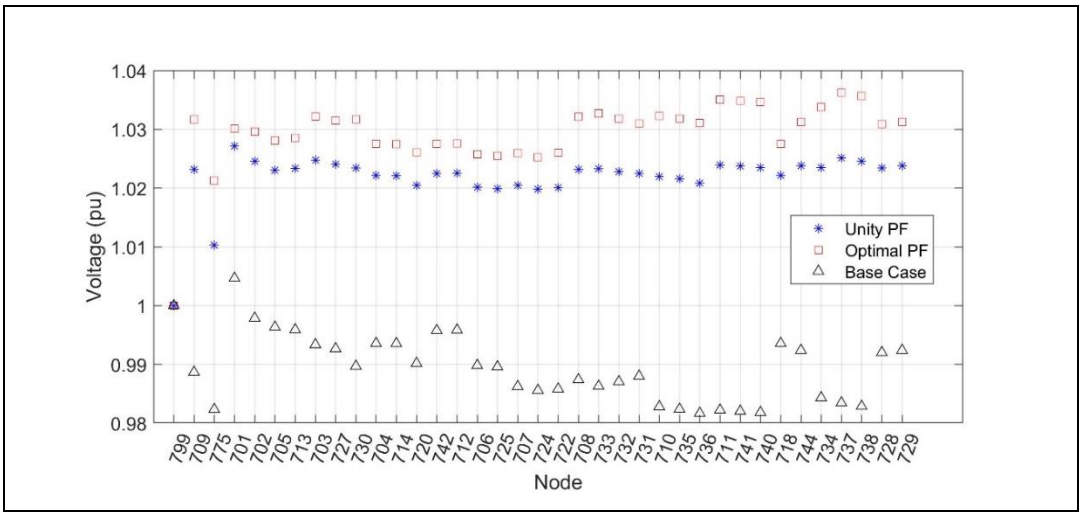


Figure 8. Voltage Profile of Phase c for the Base Case, the Cases DGs Operated at Unity and Optimal Power Factor

6. Conclusion

This paper shows that DG unit integration in the distribution test system aims to minimize power loss by improving the voltage profile. In this study, several heuristic optimization algorithms have been utilized to resolve the DGs unit allocation and sizing problem. Compared to other optimization algorithms used in the study, GA yields better results for optimal locations and sizes in reducing power losses. The approaches drop the active power losses by ensuring the voltages are within the defined limits. By improving the voltage profile of the whole network, power losses have been minimized through the use of optimally located and sized distributed generations, which in turn supports reducing network load. The utilization of DGs' reactive power capabilities demonstrates that when they perform at the ideal power factor, the power losses more than DGs operated under the unity power factor due to the lack of reactive power capability.

References

[1] Preeti Rani, Ved Parkash, Naveen Kumar Sharma, Technological aspects, utilization and impact on power system for distributed generation: A comprehensive survey, Renewable and Sustainable Energy Reviews,V olume 192,2024,114257

[2] Bollen, Math HJ, and Fainan Hassan. Integration of distributed generation in the power system. John wiley & sons, 2011.

[3] Borges, C. L., & Falcão, D. M. (2006, July). Optimal distributed generation allocation for reliability, losses, and voltage improvement. Int. Jour. of Elect.l Power & Ene. Sys., 28(6), 413–420. https://doi.org/10.1016/j.ijepes.2006.02.003.

[4] Devabalaji, K., & Ravi, K. (2016, September). Optimal size and siting of multiple DG and DSTATCOM in radial distribution system using Bacterial Foraging Optimization Algorithm. Ain Shams Engineering Journal, 7(3), 959–971. https://doi.org/10.1016/j.asej.2015.07.002.

[5] Y. Alinejad-Beromi, M. Sedighizadeh, M. R. Bayat and M. E. Khodayar, "Using genetic algorithm for distributed generation allocation to reduce losses and improve voltage profile," 2007 42nd International Universities Power Engineering Conference, Brighton, UK, 2007, pp. 954-959, doi: 10.1109/UPEC.2007.4469077.

[6] Gandomkar, M., Vakilian, M., & Ehsan, M. A combination of genetic algorithm and simulated annealing for optimal DG allocation in distribution networks. Canadian Conference on Electrical and Computer Engineering, 2005. https://doi.org/10.1109/CCECE.2005.1557013.

[7] Ntombela, M., Musasa, K., & Leoaneka, M. C. (2022, October 10). Power Loss Minimization and Voltage Profile Improvement by System Reconfiguration, DG Sizing, and Placement. Computation, 10(10), 180. https://doi.org/10.3390/computation10100180.

[8] Selim, A., Kamel, S., & Jurado, F. (2020, January). Efficient optimization technique for multiple DG allocation in distribution networks. Applied Soft Computing, 86, 105938. https://doi.org/10.1016/j.asoc.2019.105938.

[9] Alhamali, M. E. Farrag, G. Bevan and D. M. Hepburn, "Determination of optimal site and capacity of DG systems in distribution network based on genetic algorithm," 2017 52nd UPEC, Heraklion, Greece, 2017, pp. 1-6, doi: 10.1109/UPEC.2017.8231996.

[10] S. Biswas, S. K. Goswami and D. Bhattacharya, "Optimal Placement of Distributed Generation in an Unbalanced Radial Distribution System Considering Load Variation," 2019 IEEE Region 10 Symposium (TENSYMP), Kolkata, India, 2019, pp. 173-178, doi: 10.1109/TENSYMP46218.2019.8971226.

[11] P. Alemi and G. B. Gharehpetian, "DG allocation using an analytical method to minimize losses and to improve voltage security," 2008 IEEE 2nd International Power and Energy Conference, Johor Bahru, Malaysia, 2008, pp. 1575-1580, doi: 10.1109/PECON.2008.4762731.

[12] MATLAB, Natick, Massachusetts: The MathWorks Inc.; 2022.

[13] RC, Kennedy J. Eberhart. "Particle swarm optimization." Proc IEEE Int Conf Neural Networks. Vol. 4. 1995.

[14] Lakum, A., & Mahajan, V. (2021, August). A novel approach for optimal placement and sizing of active power filters in radial distribution system with nonlinear distributed generation using adaptive grey wolf optimizer. Engineering Science and Technology, an International Journal, 24(4), 911–924. https://doi.org/10.1016/j.jestch.2021.01.011.

[15] KUMAR, T., & GANESH, D. (2015, January 15). Optimal allocation of DG unit for the radial distribution system using genetic algorithm. IJIREEICE, 87–90. https://doi.org/10.17148/ijireeice.2015.3118.

[16] Sheren. (n.d.). Optimal Allocation of DG Units for Radial Distribution Systems using Genetic Algorithm. Int. Journal of Engin. and Advanced Tech., Volume-1(Issue-6, August 2012).

[17] Madhusudhan, M., Kumar, N. & Pradeepa, H. Optimal location and capacity of DG systems in distribution network using genetic algorithm. Int. j. inf. tecnol. 13, 155–162 (2021). https://doi.org/10.1007/s41870-020-00545-2.

[18] Gümüş, T. E., Emiroglu, S., & Yalcin, M. A. (2023, January). Optimal DG allocation and sizing in distribution systems with Thevenin based impedance stability index. International Journal of Electrical Power & Energy Systems, 144, 108555. https://doi.org/10.1016/j.ijepes.2022.108555

[19] Das, G., Hazarika, D. Multi-objective Particle Swarm Optimization-Based Placement and Sizing of Distributed Generators Integrated to Unbalanced Low-Voltage Microgrids by Four-Leg Inverters. J. Inst. Eng. India Ser. B 104, 731–747 (2023).

[20] Seyed Abbas Taher, Seyed Ahmadreza Afsari Optimal location and sizing of DSTATCOM in distribution systems by immune algorithm Electr. Power Energy Syst., 60 (2014), pp. 34-44.
[21] Yuvaraj, T., & Ravi, K. (2018, December). Multi-objective simultaneous DG and DSTATCOM allocation in radial distribution networks using cuckoo searching algorithm. Alexandria Engineering Journal, 57(4), 2729–2742. https://doi.org/10.1016/j.aej.2018.01.001.
[22] S. Biswas, S. K. Goswami and D. Bhattacharya, "Optimal Placement of Distributed Generation in an Unbalanced Radial Distribution System Considering Load Variation," 2019 IEEE Region 10 Symposium (TENSYMP), Kolkata, India, 2019, pp. 173-178, doi: 10.1109/TENSYMP46218.2019.8971226.
[23] Maria Teresa Costa-Campi, Daniel Daví-Arderius, Elisa Trujillo-Baute,The economic impact of electricity losses,Energy Economics,Volume 75,2018,Pages 309-322,
[24] Truong, K. H., Nallagownden, P., Elamvazuthi, I., & Vo, D. N. (2020, March). A Quasi-Oppositional-Chaotic Symbiotic Organisms Search algorithm for optimal allocation of DG in radial distribution networks. Applied Soft Computing, 88, 106067. https://doi.org/10.1016/j.asoc.2020.106067.
[25] Muthukumar Kandasamy, Renugadevi Thangavel, Thamaraiselvi Arumugam, Sureshkumar Kumaravel, Sakthivel Aruchamy, Wook-Won Kim, Zong Woo Geem,Strategic incorporation of DSTATCOM and distributed generations in balanced and unbalanced radial power distribution networks considering time varying loads,Energy Reports,Volume 9,2023,Pages 4345-4359
[26] M. Naveen Babu, P.K. Dhal,Impact of load flow and network reconfiguration for unbalanced distribution systems,Measurement: Sensors,Volume 32,2024,101078
[27] Sudipta Ghosh, S.P. Ghoshal, Saradindu Ghosh,Optimal sizing and placement of distributed generation in a network system, International Journal of Electrical Power & Energy Systems,Volume 32, Issue 8,2010,Pages 849-856,
[28] I. Dumancic, C. Tranchita and J. Kluge, "Dynamic Simulation of Distribution Power Systems with Distributed Renewable Generation," 2021 IEEE PES/IAS PowerAfrica, Nairobi, Kenya, 2021, pp. 1-5, doi: 10.1109/PowerAfrica52236.2021.9543396.
[29] K. P. Schneider, B. A. Mather, B. C. Pal, C. W. Ten, G. J. Shirek, H. Zhu, J. C. Fuller, J. L. R. Pereira, L. F. Ochoa, L. R. de Araujo, R. C. Dugan, S. Matthias, S. Paudyal, T. E. McDermott, and W Kersting, “Analytic Considerations and Design Basis for the IEEE Distribution Test Feeders,” IEEE Transactions on Power Systems, vol. PP, no. 99, pp. 1-1, 2017
[30] W. H. Kersting, “Radial distribution test feeders,” IEEE Transactions on Power Systems, vol. 6, no. 3, pp. 975–985, Aug. 1991. doi: 10.1109/59.119237
[31] Distribution System Analysis Subcommittee, IEEE 37 node test feeder, IEEE Power Engineering Society, The Institute of Electrical and Electronics Engineers, Inc. https://cmte.ieee.org/pes-testfeeders/wp-content/uploads/sites/167/2017/08/feeder37.zip
[32] R. C. Dugan, T. E. McDermott, “An open source platform for collaborating on smart grid research,” in Proc. IEEE Power Energy Soc. Gen. Meet., Detroit, 2011, pp. 1–7. doi:10.1109/pes.2011.6039829
[33] G. Nageswara Reddy, G. Pavan Kumar, 2013, Best Location of Distributed Generation on Distribution Networks Using GA, International Journal of Engineering Research & Technology, 2, Issue 11.
[34] Al-Ammar, Essam. “Optimal Allocation and Sizing of Distributed Generation in Distribution Networks Using Genetic Algorithms.” 11th International Conference on Electrical Power Quality and Utilization (2011).
[35] Holland, J.H., 1975. Adaptation in Natural and Artificial Systems, second ed. University of Michigan Press, Ann Arbor, MI, 1992.
[36] P. Gopu, S. Naaz and K. Aiman, "Optimal Placement of Distributed Generation using Genetic Algorithm," 2021 International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT), Bhilai, India, 2021, pp. 1-6, doi: 10.1109/ICAECT49130.2021.9392496.
[37] Majid Jaberipour, Esmaile Khorram, Behrooz Karimi,Particle swarm algorithm for solving systems of nonlinear equations,Computers & Mathematics with Applications,Volume 62, Issue 2,2011,Pages 566-576
[38] Gardeux, V., H. Omran, M. G., Chelouah, R., Siarry, P., & Glover, F. (2017). Adaptive pattern search for large-scale optimization. Applied Intelligence, 47, 319-330.
[39] Cihan Ersali, Baran Hekimoglu, Musa Yilmaz, Alfredo A. Martinez-Morales, Tahir Cetin Akinci,Disturbance rejecting PID-FF controller design of a non-ideal buck converter using an innovative snake optimizer with pattern search algorithm,Heliyon,Volume 10, Issue 14,2024
[40] Mirjalili, Seyedali, Seyed Mohammad Mirjalili, and Andrew Lewis. "Grey wolf optimizer." Advances in engineering software 69 (2014): 46-61.
[41] Fattahi, Hadi, Hossein Ghaedi, and Danial Jahed Armaghani. "Optimizing fracture toughness estimation for rock structures: A soft computing approach with GWO and IWO algorithms." Measurement (2024): 115306.
[42] Moayedi, Hossein, Hoang Nguyen, and Loke Kok Foong. "Nonlinear evolutionary swarm intelligence of grasshopper optimization algorithm and gray wolf optimization for weight adjustment of neural network." Engineering with Computers 37.2 (2021): 1265-1275.

Article Information Form

Acknowledgments
We express our gratitude to the referees for their insightful comments that enhanced the paper's presentation.

Authors' Contributions
The two authors worked together to complete this project. The final manuscript was read and approved by all the authors.

Conflict of Interest
There is no conflict of interest declared by the writers.

Funding
Funding has not been revealed by the writers.

Plagiarism Statement
This article has been scanned by iThenticate ™.

SAKARYA UNIVERSITY JOURNAL OF COMPUTER AND INFORMATION SCIENCES
http://saucis.sakarya.edu.tr/
e-ISSN: 2636-8129
Publisher: Sakarya University
Vol. 7, No. 3, 2024, 521-530
DOI: 10.35377/saucis...1441194

REVIEW

Examining Artificial Intelligence and Fundamental Human Rights Through a Review and Student Perspectives from North Macedonian Universities

Enes Bajrami[1,*], Festim Halili[2], Florim Idrizi[2]

[1] Department of Computer Science and Engineering, Ss. Cyril and Methodius University, North Macedonia
[2] Department of Computer Science, University of Tetova, North Macedonia

Corresponding author:
Enes Bajrami, Department of Computer Science and Engineering, Ss. Cyril and Methodius University, North Macedonia
enes.bajrami@students.finki.ukim.mk

Article History:
Received: 22.02.2024
Accepted: 04.06.2024
Published Online: 31.12.2024

ABSTRACT
This comprehensive paper seeks to explore the intricate intersection between artificial intelligence (AI) and fundamental human rights, shedding light on pivotal areas including Privacy & Surveillance, Bias in Decision Systems, and Autonomous Systems. Through an exhaustive analysis of scholarly literature and contemporary advancements, this paper aims to unveil the complex interplay between AI technologies and the safeguarding of human rights. Moreover, it integrates viewpoints derived from students representing diverse academic backgrounds across numerous universities in North Macedonia, elicited through a meticulously crafted questionnaire. In essence, this paper endeavors to provide a holistic understanding of the multifaceted relationship between AI and human rights, drawing upon academic research, real-world examples, and the perspectives of the next generation of thinkers and leaders. By delving into these critical areas and synthesizing insights from various sources, it seeks to contribute to ongoing discourse and facilitate informed discussions on the ethical implications and societal ramifications of AI advancements.

Keywords: Artificial Intelligence, Ethics, Privacy, Bias, Fundamental

1. Introduction

By looking at the most recent achievements and increasing their implementation, AI and Autonomous Systems have gained more influence over our lives. Ethical questions about these systems have become more obvious and real as their influence has grown [1]. System development is now more than just a technological or engineering problem, as evidenced by biased algorithms in social media, autonomous car decision-making systems, and even the social effects of automatization in entire transportation ecosystems like autonomous maritime [2]. Ethics and our values must be incorporated into AI and Autonomous Systems as soon as possible, as they are already present in the world around us [3]. Concerning ethics as a piece of framework configuration has likewise acquired consideration from legislative and normalization levels. The academic discourse on the connection of AI and ethics has been continuous for decades, but the advancement of frameworks and ethical inquire about have been marginally crossed [4]. The ethical investigate has been primarily centered on the potential of AI on hypothetical level [5]. So, the address remains open on application level: How should ethics be executed in practice into these systems? [1] [6]. As an arrangement for understanding the field of ethics of AI, philosophical conceptualization ought to be utilized [1] [7]. This paper aims to conduct a comprehensive review of the intersection between AI and Fundamental Human Rights, drawing insights from various research papers and books. Subsequently, we administered a questionnaire across three universities - University of Tetova, Ss. Cyril and Methodius University, and South East European University - to gather students' perspectives on the relationship between AI and human rights.

2. Previous Research

Numerous research papers and books have delved into the complex relationship between AI and fundamental human rights. Scholars have explored topics such as algorithmic bias, privacy concerns, surveillance, and the impact of automation on employment and socioeconomic equality. These studies highlight the need for robust frameworks and regulations to ensure AI systems respect and uphold human rights principles. In the article [8] the authors examine firstly whether AI inalienably clashes with human rights and human independence. Another, they dive into how AI could be connected to the beneficence criterion of AI morals and how AI could be applied in human rights-related regions. At last, they expand on personal viewpoints of what it implies to comply with human rights, tending to AI-specific issue zones. This article [9] was composed sometime recently the distribution by the EU Commission of its proposition for an AI direction. In a to begin with a temporary

Cite as: E. Bajrami, F. Halili and F. Idrizi, "Examining artificial intelligence and fundamental human rights through a review and student perspectives from North Macedonian universities", *Sakarya University Journal of Computer and Information Sciences*, vol.7, no. 3, pp. 521-530, 2024. Doi: 10.35377/saucis…1441194

investigation of the proposed direction, the creators watch that the proposed direction consolidates a few of the fundamental standards laid down within the article: it prioritizes principal rights. It consolidates a few human rights standards, such as responsibility, and the consideration of administration through supervisory specialists to execute and uphold the control. By the by, they still feel that numerous of the recommendations in the article, which would offer assistance to operationalize the direction, are not tended to. One case is the decreased scope of the control to a list of "high-risk applications," clearing out without a legitimate system all other AI applications. The creators accept that the standards that rouse the control ought to be too connected in "lower-risk applications." Characterizing as it were the compliance prepare for AI designers, but taking off open the particular specialized prerequisites that these high-risk applications might meet clears out untouched the existing hole between lawful dialect and designing hone. In the article [10] the significance of this study in the current context is emphasized, highlighting the considerable efforts still required to integrate AI in safeguarding human rights and ensuring human dignity. Specifically, the research develops a methodology to embed human rights principles within technological systems, aiming to protect and secure these rights in the digital realm. This study systematically reviews existing literature to address the primary issue of fortifying human rights, asserting that AI should be employed to uphold rather than violate these rights. The study concludes with recommendations for future research and development in this critical domain. This article [11] advances the field of human-centered AI by providing practical recommendations for designing AI systems that enhance user experiences, promote user empowerment, and adhere to ethical standards. It emphasizes the harmonious coexistence of humans and AI, aiming to enhance well-being and autonomy while envisioning a future where AI technologies benefit humanity. Overall, this research underscores the importance of human-centered AI in creating a positive impact. By focusing on users' needs and values, AI systems can be designed to empower individuals and enrich their experiences. Ethical considerations are essential to ensure fairness and transparency. Through effective collaboration between humans and AI, the potential of AI can be harnessed to create a future that aligns with human aspirations and promotes societal well-being. In the article [12] authors provide a thorough overview of the field of AI ethics, encompassing a summary and analysis of ethical issues, guidelines, and principles related to AI. They discuss various approaches to address these ethical concerns and outline methods for evaluating the ethics of AI technologies. Additionally, the article explores research challenges and future perspectives. This comprehensive review aims to offer researchers a broad understanding of AI ethics, thereby aiding their further investigation and research. This article [13] presents an overview and analysis of the ethical issues associated with artificial intelligence, strategies for addressing them, and techniques for assessing AI ethics. The study highlights the growing ethical and social implications resulting from the widespread use of technology in various sectors. It also points out the inadequacy of current technical solutions and the need for an appropriate framework to manage these concerns. Ultimately, the paper emphasizes the need for comprehensive research to develop effective technological solutions to these ethical challenges. In this article [14] the authors provide a comprehensive analysis of algorithmic bias, covering its origins, ethical and social implications, and potential remedies. They introduce an innovative methodology for identifying and measuring algorithmic bias that combines statistical analysis with input from users and domain experts. The paper explores various algorithmic biases, such as selection bias, confirmation bias, and measurement bias. It investigates the underlying causes, including data integrity issues, algorithmic design decisions, and institutional prejudices. The study focuses on the negative impacts of algorithmic bias, such as perpetuating social inequality and hindering societal progress. By identifying the sources and consequences of algorithmic bias and suggesting effective interventions, this research aims to contribute to developing fair and equitable AI systems that can promote societal advancement and benefit individuals across diverse demographics.

3. Research Methodology

In this review paper, the methodology involved a two-step process aimed at examining the intersection between artificial intelligence (AI) and fundamental human rights.

3.1. Literature Review

Initially, a comprehensive literature review was conducted to explore various aspects of AI's impact on human rights, such as algorithmic bias, privacy concerns, and autonomy. Academic databases were searched for relevant articles, books, and reports that discussed these topics. This literature review provided a solid foundation for understanding the key issues and debates surrounding AI and human rights, serving as the basis for further inquiry.

3.2. Questionnaire Survey

Following the literature review, a questionnaire was developed and administered to students from three universities in North Macedonia. The purpose of the questionnaire was to gather empirical data on students' perspectives and experiences regarding AI and its implications for human rights. To distribute the questionnaire, contact was established with the relevant heads of study programs at each university to obtain permission for the survey. Once permission was granted, the questionnaire was administered online. Participants were assured of the confidentiality and anonymity of their responses, and informed consent was obtained before they participated.

3.3. Data Collection and Analysis

The link to the online questionnaire was made available to students for a specified period, after which the link was closed to prevent further responses. The collected data were then compiled and analyzed. Qualitative analysis techniques were

employed to identify common themes and patterns across the responses. Additionally, the data were visualized using pie charts to provide a clear and concise representation of the findings.

3.4. Ethical Consideration

Throughout the research process, ethical considerations were taken into account. Participants were informed about the purpose of the study, and their consent was obtained before they took part in the survey. Measures were also taken to ensure the confidentiality and anonymity of their responses.

By integrating insights from existing research with empirical data from students in North Macedonia, this study aims to provide a nuanced understanding of the complex relationship between AI and fundamental human rights. The methodology employed in this review paper ensures the integrity and validity of the findings, contributing to the ongoing discourse on AI and human rights.

4. Fundamental Disputations

In this section, we discuss the ethical issues that arise when humans use AI and robotics systems that are more or less autonomous. This means that we look at problems that arise when certain uses of the technologies are done, but not when others are done. It should be remembered, nonetheless, that advances will continuously make a few purposes simpler, and in this way more successive, and upset different purposes. Because the design of technical objects has an ethical impact on how they are used [15] [16], we also need "responsible design" in this area in addition to "responsible use." The emphasis on use does not presuppose which ethical strategies are most suitable for addressing these issues; they likely could be goodness moral instead of consequentialist or worth based [17]. This section is additionally impartial regarding the inquiry of whether computer-based intelligence frameworks have "intelligence" or other mental properties: It would work just as well if AI were just seen as the current face of automation [18].

4.1. Privacy & Surveillance

In information technology, privacy and surveillance are generally discussed which primarily concerns access to personal information and private data [19]. Privacy has a few perceived viewpoints, e.g., "he right to be let alone", information privacy, privacy as a part of personhood, command over data around oneself, and the right to mystery [20]. Security studies have generally centered around state observation by secret administrations yet presently incorporate reconnaissance by other state specialists, organizations, and even people [21]. The digital world has grown significantly: All information assortment and capacity are currently computerized, our lives are progressively computerized, most advanced information is associated with a solitary Web, and there is something else and more sensor innovation being used that creates information about non-computerized parts of our lives [22]. Simultaneously, controlling who gathers which information, and who approaches, is a lot harder in the computerized world than it was in the simple universe of paper and calls [21]. For instance, face recognition in photographs and recordings permits ID and along these lines profiling and looking for people [23]. This keeps involving different methods for recognizable proof, e.g., "device fingerprinting", which are typical on the Web (now and again uncovered in the "privacy policy") [24] [21]. For the "enormous 5" companies (Amazon, Google, Microsoft, Apple, Facebook), the principal information assortment in some portion of their business seems, by all accounts, to be founded on duplicity, taking advantage of human shortcomings, promoting hesitation, creating compulsion, and control [25], their company's primary data collection function appears to be founded on deception, exploiting human weaknesses, encouraging procrastination, fostering addiction, and manipulation [26]. It has made many endeavors escape from the grip of these partnerships, e.g., in activities of "moderation", in some cases through the open-source development, however apparently present-day residents have lost the level of independence expected to escape while completely going on with their life and work. We have lost information responsibility if "proprietorship" is the right connection here. We have failed to keep a grip on our information [27] [28] [29].

4.2. Bias in Decision Systems

Automated AI decision choice emotionally supportive networks and "predictive analytics" work on information and produce a choice as "output". This output may be relatively insignificant or extremely significant [21]. There are many advantages to the rapid development of AI, but there are also potential dangers and difficulties. One of the key worries is the adverse consequences of predisposition in simulated intelligence on people and society. AI bias has the potential to exacerbate and even perpetuate existing inequality, resulting in marginalized groups being subjected to discrimination and limiting their access to essential services [30]. To guarantee that computer-based intelligence frameworks are fair, impartial, and serve the requirements, everything being equal, it is essential to distinguish and relieve predisposition in AI. Besides, using one-sided computer-based intelligence has various moral ramifications, including the potential for separation, obligation of engineers and policymakers, subverting public confidence in innovation, and restricting human organization and independence [31]. Tending to these moral ramifications will require a purposeful exertion from all partners included, and it is critical to foster moral rules and administrative structures that advance reasonableness, straightforwardness, and responsibility in the turn of events and utilization of man-made intelligence frameworks [32]. The use of biased AI has various moral ramifications that should be thought of. The possibility of discrimination against individuals or groups based on factors like race, gender, age, or disability is one of the main concerns. One more ethical concern is the obligation of engineers, organizations, and states

in guaranteeing that AI frameworks are planned and utilized fairly and straightforwardly [33] [34]. Additionally, the public's trust in technology may be eroded by the use of biased AI systems, resulting in lower adoption or even rejection of new technologies. The potential benefits of AI may not be realized if people do not trust the technology or if it is viewed as a tool for discrimination. This can have serious repercussions for both the economy and society [32] [33]. Last but not least, biased AI's impact on human autonomy and agency must be considered. At the point when man-made intelligence frameworks are one-sided, they can restrict individual opportunities and build up cultural power elements [35].

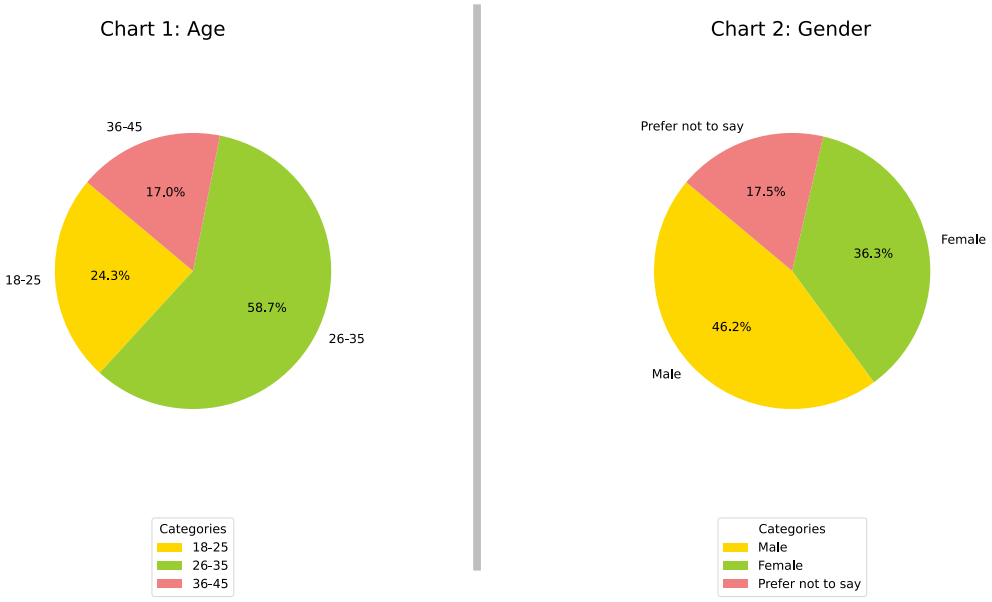
4.3. Autonomous Systems

There are a few ideas of independence in the conversation of independent systems. A more deeply felt idea is engaged in philosophical discussions where independence is the reason for liability and personhood [36]. In this unique circumstance, obligation suggests independence, yet not contrarily, so there can be frameworks that have levels of specialized independence without raising issues of liability. In robotics, the weaker, more technical concept of autonomy is relative and gradual: A system is considered autonomous to some extent about human control [37]. Taking everything into account, question is how much independent robots raise gives our present calculated plans should adjust to, or whether they simply require specialized changes. In many locales, there is a complex arrangement of common and criminal risks to determine such issues. Specialized norms, e.g., for the protected utilization of hardware in clinical conditions, will probably should be changed [38] [39]. Among the numerous independent frameworks ashore, on water, submerged, in air or space, we examine two examples: independent vehicles and independent weapons [21].

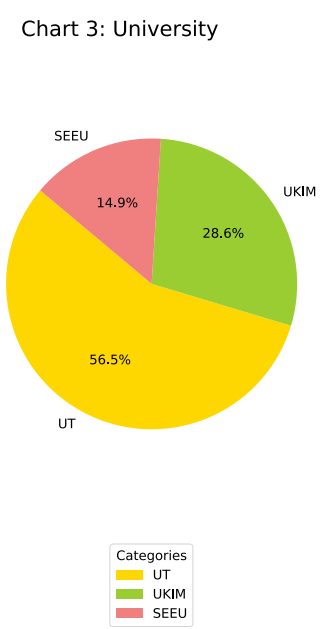
5. Results and discussion of student’s questionnaire

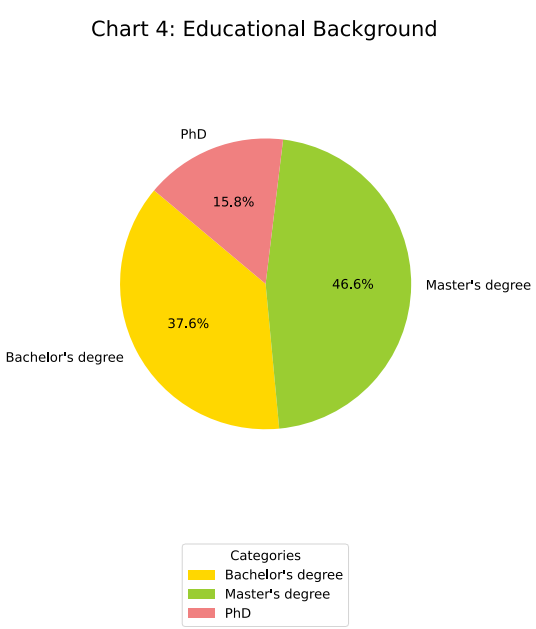
5.1. Students' Questionnaire

In this section, we present the student questionnaire from three distinct universities, forming a crucial part of our research on AI and Human Rights. A total of 234 students from various study cycles have participated in our questionnaire, providing invaluable insights into this intersectional field.

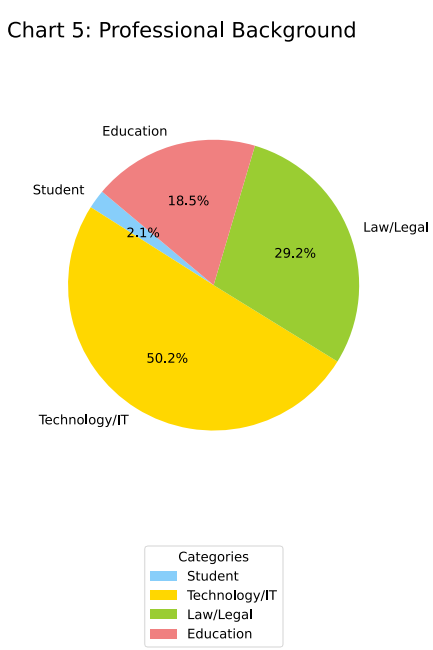


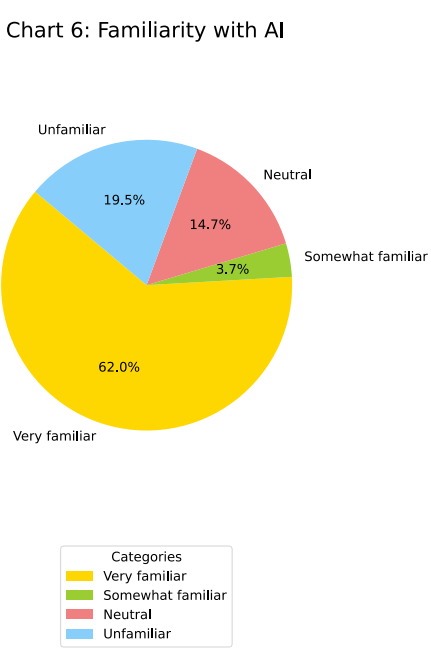
In Chart 1 our initial survey question regarding age, out of 234 responses, 60.3% fell within the 26-35 age bracket, 22.2% were between 18 and 25 years old, and 17.5% were aged 36-45 years old. Moving on to the next question on gender, our findings revealed that 46.2% identified as male, 36.3% as female, and 17.5% preferred not to disclose their gender.





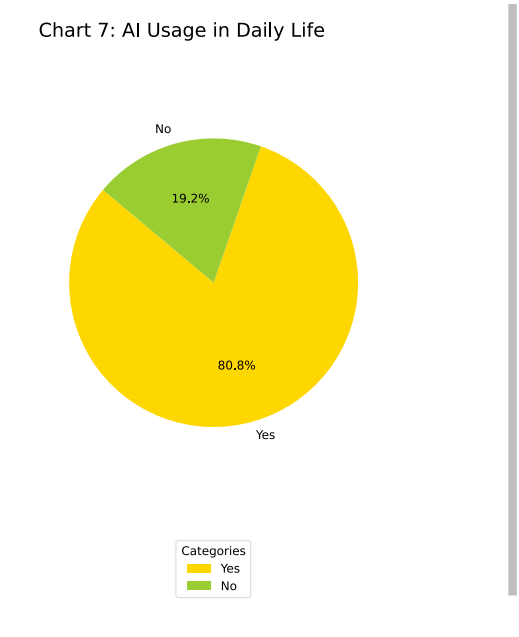
The subsequent question pertains to the universities from which students submitted their questionnaires. University of Tetova[1] emerged with the highest percentage at 56.5%, followed by Ss. Cyril and Methodius University[2] with 28.6%, and finally, the South East European University[3] with 14.9%. Another question inquired about participants' educational backgrounds. The results indicated that the Master's degree cycle held the majority at 46.6%, followed by Bachelor's degree holders at 37.6%, and lastly, individuals pursuing a PhD, comprising 15.8% of the respondents.

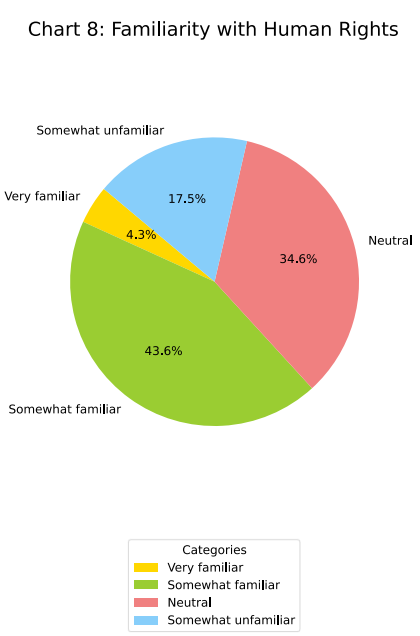




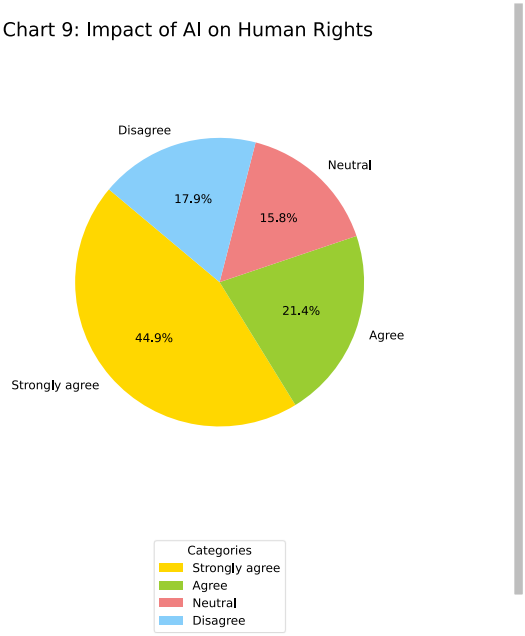
Our questionnaire highlighted the importance of professional background, especially for students balancing work and studies. IT sector had the highest representation at 50.2%, followed by Law professions at 29.2%, and Education at 18.5%. A smaller percentage were students. Regarding AI familiarity, 62.0% were very familiar, 19.5% unfamiliar, 14.7% neutral, and 3.7% somewhat familiar.

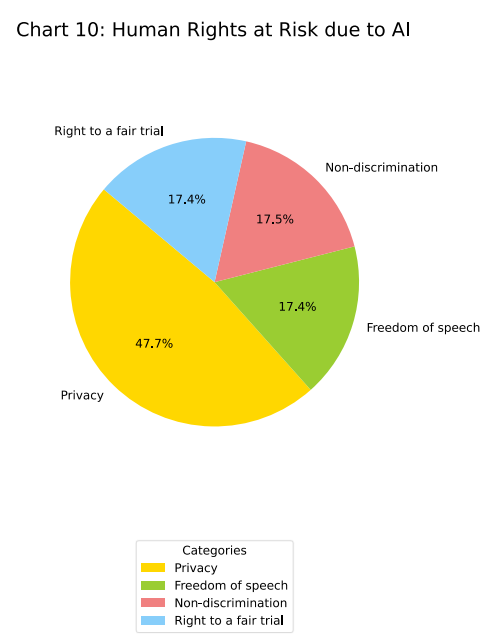
[1] UT
[2] UKIM
[3] SEEU





In our questionnaire, 80.8% reported encountering or using AI applications, while 19.2% hadn't. Regarding fundamental human rights familiarity, responses showed: 43.6% somewhat familiar, 34.6% neutral, 17.5% somewhat unfamiliar, and only 4.3% very familiar.





In our questionnaire, beliefs about AI's impact on human rights were: 44.9% strongly agree, 21.4% agree, 17.9% disagree, and 15.8% neutral. Regarding specific human rights vulnerable to AI risks, students' opinions were: 47.7% Privacy, 17.4% Freedom of speech, 17.4% Right to a fair trial, and 17.5% Non-discrimination.

Continuing our analysis, we looked at participants' concern about AI misuse violating human rights: 45.3% very concerned, 20.9% concerned, 17.9% not very concerned, and 15.8% neutral. Regarding confidence in legal frameworks, notably in North Macedonia with its National Strategy for AI by the Fund for Innovation and Technology Development: 44% very confident, 19.7% not very confident, 19.2% confident, and 16.7% neutral.

In our survey, opinions on groups facing disproportionate impacts from AI's ethical implications were: 46.6% women, 19.7% children, 18.4% men, 14.5% LGBTQ+ individuals, and 0.9% minority communities. Next, strategies for inclusive AI design were: 61.1% strict regulatory guidelines, 20.5% ethical AI education for developers, and 18.4% diverse AI development teams.

Chart 11: Concern about Potential Misuse of AI

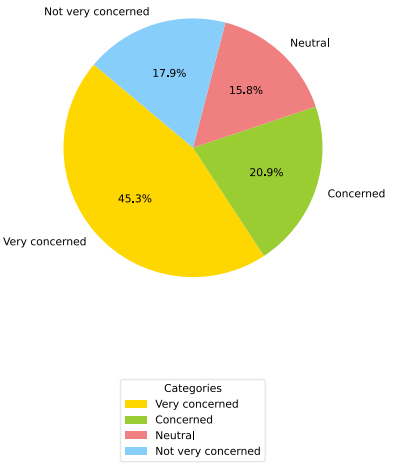


Chart 12: Confidence in Legal Frameworks for AI

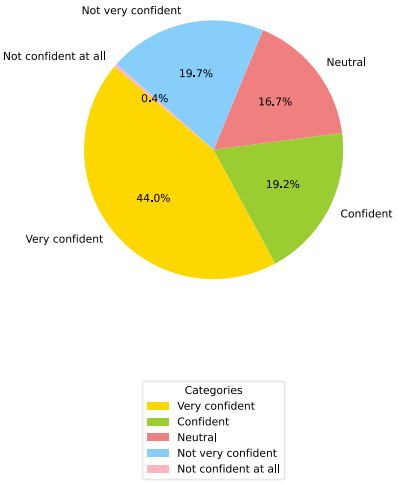


Chart 13: Groups Affected by AI on Human Rights

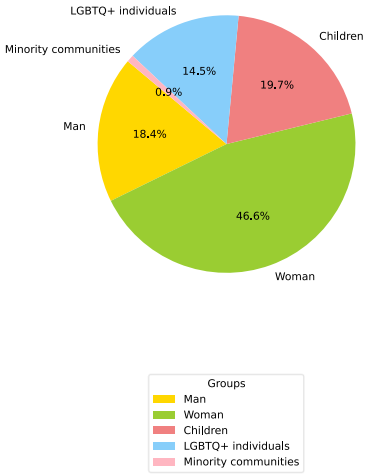


Chart 14: Ensuring Inclusivity in AI

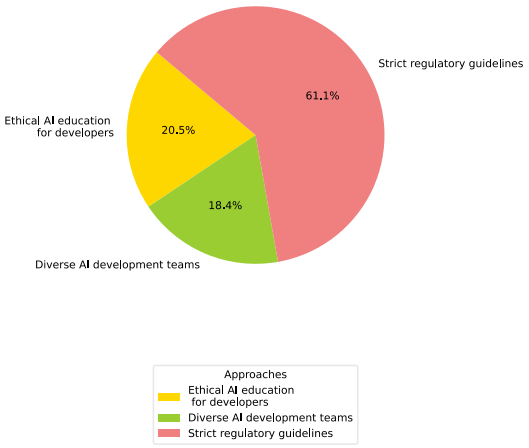


Chart 15: Public Awareness of AI Ethics

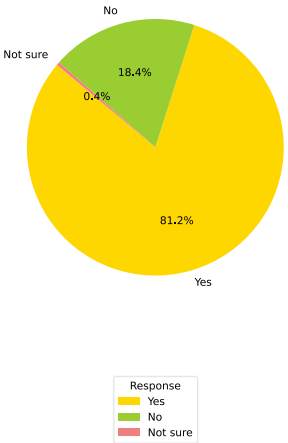
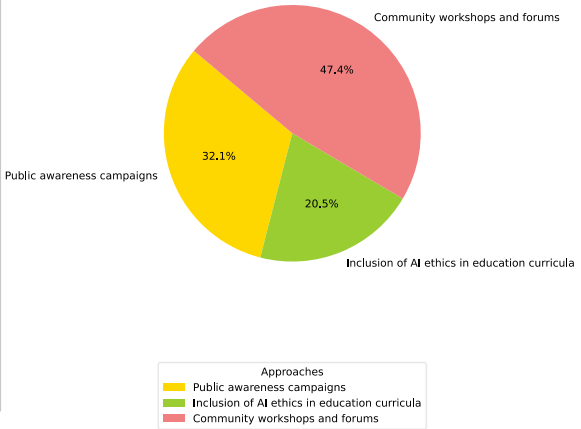


Chart 16: Educating the Public about AI Ethics



Our penultimate question delves into whether participants perceive a need for heightened public awareness and education concerning the ethical implications of AI on human rights. The responses are as follows: 81.2% answered affirmatively, indicating a perceived necessity, while 18.4% expressed dissent, and a minor 0.4% reported uncertainty on the matter. Concluding our questionnaire, our final inquiry focuses on how organizations and governments can enhance public education regarding the ethical use of AI and its potential impact on human rights. Student responses are as follows: 47.4% advocated for community workshops and forums, 32.1% suggested public awareness campaigns, and 20.5% indicated other methods not specified in the response.

5.2. Discussion

In our manuscript, we incorporated a questionnaire to gauge students' perspectives on the intersection of AI and fundamental human rights. Upon analyzing the submitted responses, it becomes apparent that students possess a degree of awareness regarding these issues. They acknowledge the potential impact of AI on the future of human privacy, reflecting a growing recognition of the implications associated with advancing technology. Furthermore, the questionnaire reveals that students believe North Macedonia requires stringent regulatory guidelines to govern the deployment and utilization of AI. This underscores a prevailing sentiment among respondents regarding the necessity for robust legal frameworks to safeguard individual rights and interests in the face of technological advancements. Additionally, the suggestion for community workshops and forums highlights a desire among students for greater public engagement and education on AI-related topics. Such initiatives could serve to enhance awareness, foster dialogue, and empower individuals to navigate the complexities of AI and its implications for human rights. Overall, the insights gleaned from the questionnaire underscore the importance of proactive measures to address the ethical, legal, and societal dimensions of AI. By heeding students' perspectives and advocating for informed decision-making, stakeholders can work towards cultivating a more responsible and rights-conscious approach to AI development and deployment in North Macedonia and beyond.

6. Conclusion

In conclusion, this paper provides an in-depth examination of the intricate relationship between artificial intelligence (AI) and fundamental human rights. It primarily focuses on key areas such as privacy and surveillance, biases inherent in decision-making systems, and the implications of autonomous AI systems. By synthesizing findings from a various of research papers and incorporating insights derived from questionnaire responses collected from multiple universities in North Macedonia, our analysis reflects the current landscape and aligns with projected developments in the field. However, it is apparent that further exploration and investigation are imperative to confront emerging challenges and grasp the subtleties within this domain. Moving forward, it is essential to prioritize sustained research endeavors aimed at deepening our comprehension and adeptly navigating the multifaceted terrain of AI and its ramifications on human rights.

References

[1] V. Vakkuri and P. Abrahamsson, "The Key Concepts of Ethics of Artificial Intelligence," *IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC)*, pp. 1-6, 2018.

[2] J. M. Abowd, "How Will Statistical Agencies Operate When All Data Are Private?," *Journal of Privacy and Confidentiality*, vol. 7, no. 3, pp. 1-15, 2017.

[3] C. Allen, I. Smit and W. Wallach, "Artificial Morality: Top-down, Bottom-up, and Hybrid Approaches," *Ethics and Information Technology* , vol. 7, pp. 149-155, 2005.

[4] C. Allen, W. Wallach and I. Smit, "Why Machine Ethics?," *Intelligent Systems, IEEE*, vol. 21, no. 4, pp. 12-17, 2006.

[5] M. Brundage, "Limitations and risks of machine ethics," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 26, no. 3, pp. 355-372, 2014.

[6] C. Mayer, Developing Autonomous Systems in an Ethical Manner, Norfolk, Virginia: NATO, 2015.

[7] P. Boddington, Towards a Code of Ethics for Artificial Intelligence, 1st ed., Springer, 2017, pp. 0-143.

[8] Alexander KRIEBITZ;Christoph LÜTGE, "Artificial Intelligence and Human Rights: A Business Ethical Assessment," *Business and Human Rights Journa*, vol. 5, no. 1, 2020.

[9] Jesús Salgado-Criado, Celia Fernández-Aller, "A Wide Human-Rights Approach to Artificial Intelligence Regulation in Europe," *IEEE Technology and Society Magazine*, vol. 40, no. 2, pp. 55 - 65, 2021.

[10] S. Kathuria, P. Rawat, R. Singh, A. Gehlot, N. Kathuria and S. Pandey, "Artificial Intelligence: Creation of Human Rights in Technology," *International Conference on Artificial Intelligence and Smart Communication (AISC)*, pp. 328-331, 2023.

[11] U. A. Usmani, A. Happonen and J. Watada, "Human-Centered Artificial Intelligence: Designing for User Empowerment and Ethical Considerations," *International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA), Istanbul, Turkiye,* pp. 1-7, 2023.

[12] C. Huang, Z. Zhang, B. Mao and X. Yao, "An Overview of Artificial Intelligence Ethics," *IEEE Transactions on Artificial Intelligence,* vol. 4, no. 4, pp. 799-819, 2023.

[13] Z. Ayub and M. T. Banday, "Ethics in Artificial Intelligence: An Analysis of Ethical Issues and Possible Solutions," *Third International Conference on Smart Technologies, Communication and Robotics (STCR), Sathyamangalam, India,* pp. 1-6, 2023.

[14] L. R. Jain and V. Menon, "AI Algorithmic Bias: Understanding its Causes, Ethical and Social Implications," *IEEE 35th International Conference on Tools with Artificial Intelligence (ICTAI), Atlanta, GA, USA,* pp. 460-467, 2023.

[15] Wybo Houkes;Pieter E. Vermaas, Philosophy of Engineering and Technology, Springer, 2010.

[16] Peter-Paul Verbeek, "Moralizing Technology: Understanding and Designing the Morality of Things," The University of Chicago Press, 2011.

[17] L. Floridi, J. Cowls, M. Beltrametti, R. Chatila, P. Chazerand, V. Dignum, C. Luetge, R. Madelin, U. Pagallo, F. Rossi, B. Schafer, P. Valcke and E. Vayena, "AI4People—An Ethical Framework for a Good AI Society: Opportunities, Risks, Principles, and Recommendations," *Minds and Machines,* vol. 28, pp. 689-707, 2018.

[18] Vincent C. Müller; Nick Bostrom , "Future Progress in Artificial Intelligence: A Survey of Expert Opinion," *Fundamental Issues of Artificial Intelligence - Springer,* vol. 376, p. 555–572, 2016.

[19] K. Macnish, The Ethics of Surveillance, 1st ed., Routledge, 2018, p. 216.

[20] Colin J. Bennett and Charles Raab, The Governance of Privacy - Policy Instruments in Global Perspective, The MIT Press, 2006.

[21] V. C. Müller, Ethics of Artificial Intelligence and Robotics, Stanford Encyclopedia of Philosophy, 2020.

[22] A. Bertolini and G. Aiello, "Robot companions: A legal and ethical analysis," *The Information Society - An International Journal,* vol. 34, no. 3, pp. 130-140, 2018.

[23] R. Sparrow, "Killer Robots," *Journal of Applied Philosophy,* vol. 24, no. 1, pp. 62-77, 2007.

[24] R. Sparrow, "Robots in aged care: a dystopian future?," *AI & Society,* vol. 31, pp. 445-454, 2015.

[25] B. Schneier, Data and Goliath: The Hidden Battles to Collect Your Data and Control Your World, W. W. Norton & Company, 2016, pp. 1-448.

[26] S. Zuboff, The Age of Surveillance Capitalism: The Fight for a Human Future at the New Frontier of Power, 1st ed., PublicAffairs, 2019.

[27] C. Newport, Digital Minimalism, Portfolio Penguin, 2019.

[28] John R. Searle, "Minds, brains, and programs," Behavioral and Brain Sciences, vol. 3, no. 3, pp. 417 - 424, 1980.

[29] Bernd Carsten Stahl;David Wright, "Ethics and Privacy in AI and Big Data: Implementing Responsible Research and Innovation," IEEE Security Privacy, vol. 16, no. 3, pp. 26-33, 2018.

[30] Safiya Umoja Noble, Algorithms of Oppression - How Search Engines Reinforce Racism, NYU Press, 2018.

[31] Kenneth Holstein, Jennifer Wortman Vaughan, Hal Daumé III, Miro Dudík, Hanna Wallach, "Improving fairness in machine learning systems: What do industry practitioners need?," Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, 2019.

[32] E. Ferrara, "Fairness and Bias in Artificial Intelligence: A Brief Survey of Sources, Impacts, and Mitigation Strategies," vol. 6, no. 1, 2024.

[33] Ziad Obermeyer;Brian Powers;Christine Vogeli;Sendhil Mullainathan, "Dissecting racial bias in an algorithm used to manage the health of populations," Science, vol. 366, no. 6464, pp. 447-453, 2019.

[34] Brent Daniel Mittelstadt; Patrick Allo; Mariarosaria Taddeo;Sandra Wachter;Luciano Floridi, "The ethics of algorithms: Mapping the debate," Sage Journal, vol. 3, no. 2, 2016.

[35] Mike Ananny;Kate Crawford, "Seeing without knowing: Limitations of the transparency ideal and its

application to algorithmic accountability," Sage Journal, vol. 20, no. 3, 2016.

[36] J. Christman, "Autonomy in Moral and Political Philosophy," Stanford Encyclopedia of Philosophy, 2018.

[37] Vincent C. Müller, "Autonomous Cognitive Systems in Real-World Environments: Less Control, More Flexibility and Better Interaction," Springer, vol. 4, p. 212–215, 2012.

[38] Stuart Russell, Human Compatible: Artificial Intelligence and the Problem of Control, Penguin Books, 2019.

[39] Stuart Russell;Daniel Dewey;Max Tegmark, "Research Priorities for Robust and Beneficial Artificial Intelligence," Association for the Advancement of Artificial Intelligence., vol. 36, no. 4, p. 105–114, 2015.

Article Information Form

Acknowledgments
We would like to extend our sincere gratitude to all the students from the three universities who participated in the questionnaire. Your valuable insights and time were crucial to the success of this research. Additionally, we would like to thank the heads of departments for their collaboration and support. Your cooperation and facilitation were instrumental in the smooth conduct of our study.

Author(s) Contributions
Enes Bajrami: Conducted the experimental analysis, interpreted the data, and drafted the manuscript. Responsible for designing the study methodology and ensuring the accuracy of the results presented.
Festim Halili & Florim Idrizi: Coordinated with the heads of departments at the participating universities to distribute the questionnaire link to students. Their efforts in securing collaboration from other institutions were vital to the data collection process.

Conflict of Interest Notice
The authors declare no conflict of interest.

Ethical Approval and Informed Consent
It is declared that during the preparation process of this study, scientific and ethical principles were followed, and all the studies benefited from are stated in the bibliography.

Availability of data and material
Not applicable

Plagiarism Statement
This article has been scanned by iThenticate ™.