SAUCIS

Sakarya University Journal of Computer and Information Sciences

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

SAKARYA UNIVERSITY

DOI: 10.35377/saucis…1257100

RESEARCH ARTICLE

Optimization of Several Deep CNN Models for Waste Classification

Mahir Kaya[1], Samet Ulutürk[1], Yasemin Çetin Kaya[1], Onur Altıntaş[1], Bülent Turan[1]

[1] Department of Computer Engineering, Faculty of Engineering and Architecture, Tokat Gaziosmanpaşa University, Tokat, Türkiye

Corresponding author:
Mahir Kaya,
Department of Computer Engineering,
Tokat Gaziosmanpaşa University
E-mail address: mahir.kaya@gop.edu.tr

Submitted: 28 February 2023
Revision Requested: 19 June 2023
Last Revision Received: 1 July 2023
Accepted: 04 July 2023
Published Online: 14 July 2023

Citation: Kaya M. et al. (2023). Optimization of Several Deep CNN Models for Waste Classification. *Sakarya University Journal of Computer and Information Sciences*. 6 (2)
https://doi.org/10.35377/saucis...1257100

ABSTRACT

With urbanization, population, and consumption on the rise, urban waste generation is steadily increasing. Consequently, waste management systems have become integral to city life, playing a critical role in resource efficiency and environmental protection. Inadequate waste management systems can adversely affect the environment, human health, and the economy. Accurate and rapid automatic waste classification poses a significant challenge in recycling. Deep learning models have achieved successful image classification in various fields recently. However, the optimal determination of many hyperparameters is crucial in these models. In this study, we developed a deep learning model that achieves the best classification performance by optimizing the depth, width, and other hyperparameters. Our six-layer Convolutional Neural Network (CNN) model with the lowest depth and width produced a successful result with an accuracy value of 89% and an F1 score of 88%. Moreover, several state-of-the-art CNN models such as VGG19, DenseNet169, ResNet101, Xception, InceptionV3, RegnetX008, RegnetY008, EfficientNetV2S trained with transfer learning and fine-tuning. Extensive experimental work has been done to find the optimal hyperparameters with GridSearch. Our most comprehensive DenseNet169 model, which we trained with fine-tuning, provided an accuracy value of 96.42% and an F1 score of 96%. These models can be successfully used in a variety of waste classification automation.

Keywords: CNN, Deep learning, Trash, Optimization

1. Introduction

According to World Bank data, global urban waste generation annually amounts to 2.01 billion tons. Urbanization and consumption trends indicate a projected 69% increase by 2050, surpassing 3.4 billion tons [1]. Consequently, waste management systems have become indispensable for cities, ensuring efficient resource utilization and environmental protection. Inadequate waste management systems harm the environment and the economy [2]. Recycling is crucial, ranking as the second-best environmentally friendly method according to the Environmental Protection Agency [3]. The European Union achieved a 56% recycling rate in 2016, with ongoing efforts to raise this figure. Human resources are used in some aspects of recycling, which reduces efficiency, increases costs, and harms the health of those who work in this industry [4]. Intelligent systems are being used to reduce or eliminate these issues.

One of the critical stages in intelligent waste management systems is accurate and rapid waste classification. Convolutional Neural Networks (CNNs), which have end-to-end learning capability, are widely used to classify and segment images in many fields. In CNNs, multiple convolutional and max pooling layers are added sequentially to extract features from raw images. In the final stage, fully connected layers are employed for classification purposes. [5]. During the training phase, learning takes place through the iterative update of a multitude of filters and fully connected layer weights, utilizing the backpropagation algorithm [6]. At the onset of training, we establish numerous hyperparameters for the model architecture. These hyperparameters encompass various aspects such as model depth, number of filters, filter sizes, dropout rates, optimizers, learning rates, epochs, batch sizes, and more. CNN models often face challenges in achieving successful classification in datasets with limited data and high inter-class similarity [7]. In CNN models, as the depth and width increase, overfitting is commonly encountered during the training phase, especially in datasets with limited labeled data. Despite exhibiting high accuracy on the training dataset, these models tend to perform poorly on previously unseen test datasets [8]. Avoiding overfitting and determining the optimal combination of hyperparameters play a crucial role in improving the classification performance of CNN models. Therefore, CNN models with various architectures and hyperparameters were used in the study for waste classification.

In this study, several models with varying depths, widths, and optimized hyperparameters were developed. An attempt was made to achieve the best model through hyperparameter optimization using GridSearch. Although the success rate of deep learning models increases as the depth and width increase, excessive depth can lead to gradient vanishing, which prevents the model from reaching the optimum [9, 10]. This problem is partially addressed in models such as ResNet [11] and DenseNet [12] through the use of residual connections, which allow the following inputs or blocks to receive information from the previous inputs. In waste management systems, real-time classification is performed, which makes the classification time of models crucial. While deep and complex models may achieve high classification accuracy, the time taken to classify a single image in real-time implementations can exceed expectations. To address this, we developed the best model using transfer learning and optimization techniques, which led to improved classification performance. Additionally, we successfully obtained the most effective shallow or small model in terms of both accuracy and prediction speed. The proposed model has the potential for successful utilization in diverse applications of waste classification automation.

The contributions of this study can be listed as follows:

- A novel waste classification CNN model has been developed to work in different embedded systems and be integrated into waste management systems.
- The performance of different architectures with various widths and depths in waste classification is presented.
- The model performance has been increased by optimizing many hyperparameters with GridSearch.
- With transfer learning, the performance of the state-of-the-art CNN models has been increased by fine-tuning and hyperparameter optimization methods.

Different CNN architectures have been optimized with extensive experimental studies, and the optimized models that make the most successful classification have been developed. These different CNN models can be integrated into various embedded systems and used practically in waste management systems.

2. Related works

CNN is a type of artificial neural network that processes images [11]. It extracts the features of the objects in the images and learns them using various learning algorithms. CNN provides effective results in many applications, including robotics, security cameras, license plate recognition, and face recognition. Since CNN is a technique that is widely used and has a high success rate in automatically classifying waste, the number of studies in this field is significant.

Transfer learning-based CNN models have shown successful results in image classification, especially when dealing with datasets with a limited number of samples. Bircanoğlu et al. [13] developed a waste classification model based on transfer learning. Several state-of-the-art CNN models were re-trained on the TrashNet dataset. The Densenet121 model gave the best result with 95% accuracy. The DenseNet121 model is reported to have an approximate CPU time of 649 ms. Wang et al. [14] proposed a system for waste management. The TrashNet dataset was expanded in the study, and the number of categories was increased from six to nine. MobileNetV3, MobileNetV2, InceptionV3, ResNet50, ResNet101, ResNet152, and Xception CNN architectures were used and compared for waste classification. In nine categories, the accuracy ranged from 91.9% to 94.6%. The MobileNetV3 architecture achieved the highest accuracy value of 94.26%. Furthermore, the MobileNetV3 architecture had the smallest size of 49.5 MB and the shortest duration of 261.7 ms. According to Aral et al. [15], waste recycling is critical for the global economy and climate balance, so classifying recyclable waste is critical for humanity. The architectures Densenet121, DenseNet169, InceptionResnetV2, MobileNet, and Xception were used in the study. The DenseNet121 architecture yielded the highest accuracy value of 95%. Zhang et al. [16] presented how smart systems can be used to classify waste, which is a crucial step toward achieving sustainable development for people. It was emphasized that smart systems should take the place of traditional waste classification because it is ineffective. With the transfer learning model, DenseNet169 CNN architecture was chosen as the most accurate, with an accuracy value of 82%. Gyawali et al. [17] proposed a CNN model for automated waste classification to assist recycling. The TrashNet dataset was expanded and used in the proposed model to eliminate human intervention. Using the Resnet18 architecture, an accuracy of 87.8% was achieved. Rabano et al. [18] stated that waste classification is the first step in recycling and reusing waste. For this purpose, MobileNet CNN architecture was chosen to classify waste. Transfer learning and TrashNet dataset were used. The accuracy value of 87.2% was reached on the test data. Feng et al. [19] proposed an enhanced GECM-EfficientNet model. In this model, they replaced the squeeze-and-excitation (SE) module from the EfficientNet architecture with the efficient channel attention (ECA) module. With the proposed model, they achieved a classification accuracy of 94.23% on the TrashNet test dataset.

Lin et al. [20] trained models using transfer learning with five different ResNet architectures on the TrashNet dataset. They used data augmentation techniques to increase the number of samples in the dataset before training the models. In this study, they achieved an accuracy of 88.8% and an F1 score of 88.9% with the RWNet-152 model. The ResNet architecture was one of the first architectures to address the vanishing gradients problem in CNN models by introducing residual connections.

In the existing studies, multilayer hybrid convolutional neural network-based customized CNN models have also been proposed. Shi et al. [21] emphasized that existing models for waste classification still struggle with issues like a poor success rate and a long working time. The study focused on the solution of these problems through the use of a Hybrid CNN network. CNN architecture is similar to VGGNet in structure, but there are fewer parameters. The accuracy rate was reported to be 92.6%. Yang et al. [22] proposed a model based on MLH-CNN. With the proposed model, they achieved a classification accuracy of 93.72% on the TrashNet test dataset. The accuracy of the proposed models was improved by applying image preprocessing techniques. Another custom CNN model was presented by Bobulski and Kubanek [23]. They conducted a study to increase the recycling rate by classifying plastic waste. In the study, the WaDaBa dataset was utilized. The system was developed to classify plastic waste using a microcomputer system with a color camera and image processing software. The study's success rate was 74%. Riba et al. [24] conducted a study to separate recyclable waste textile products. It aims to solve environmental problems while also producing high-quality recycling. For the study, a unique dataset was created. A new CNN architecture is proposed in the study to classify textile waste. The proposed system achieved 91.1% accuracy. Tran and Nguyen [25] proposed a customized CNN model consisting of five blocks that incorporate residual connections. The proposed model leverages the strengths of convolution layers, depthwise separable convolution layers, average pooling layers, batch normalization method, and two connectors to extract feature maps and optimize the network parameters. They achieved an accuracy of 90.71% on the TrashNet dataset.

Object detection-based waste management systems have also started to be widely implemented. Sallang et al. [26] created a smart trash can in their study. The Raspberry Pi 4 placed in this trash can was used to classify waste. TensorFlow Lite is used to create a new CNN architecture for the IoT system. The architecture was trained using a dataset created specifically for this study, and an accuracy of 87% was achieved. Melinte et al. [27] worked on improving the efficiency of object detectors by employing CNN architecture for waste classification in municipalities. The study's goal is to improve the sensitivity and performance of object detection devices, as well as their generalization and detection speed. The study made use of the TrashNet dataset. The accuracy of the R-CNN-created architecture is 95.76%. Nowakowski and Pamula [28] introduced two CNN architectures in their study for the classification of electrical and electronic waste. A mobile and web-based system was developed. After utilizing the dataset created specifically for this study, the first architecture achieved an accuracy of 96.7%, while the second architecture achieved an accuracy of 93.3%.

Alrayes et al. [7] proposed a Vision Transformer based on Multilayer Hybrid Convolution Neural Network for automatic waste classification. They achieved a classification accuracy of 95.8% on the TrashNet dataset. The Vision Transformer method has also become widely used in image classification. In this study, it has been stated that the Vision Transformer method outperforms transfer learning-based CNN methods in terms of performance. The iteration times during the training phase of the models were compared, but no information was provided regarding the prediction time of the models for classifying an image in real-time implementation.

Although many models have been developed in the literature to make automatic classifications in waste management systems, the success rates have not reached the desired level. Especially in waste classification, the similarity of glass and plastic wastes at some points affects the model performances. Further investigation is required to ascertain methods for improving the performance of the classifier when dealing with a dataset that has limited samples and significant similarities between different classes. In this study, architectures that will produce the optimum result have been obtained with extensive experimental studies.

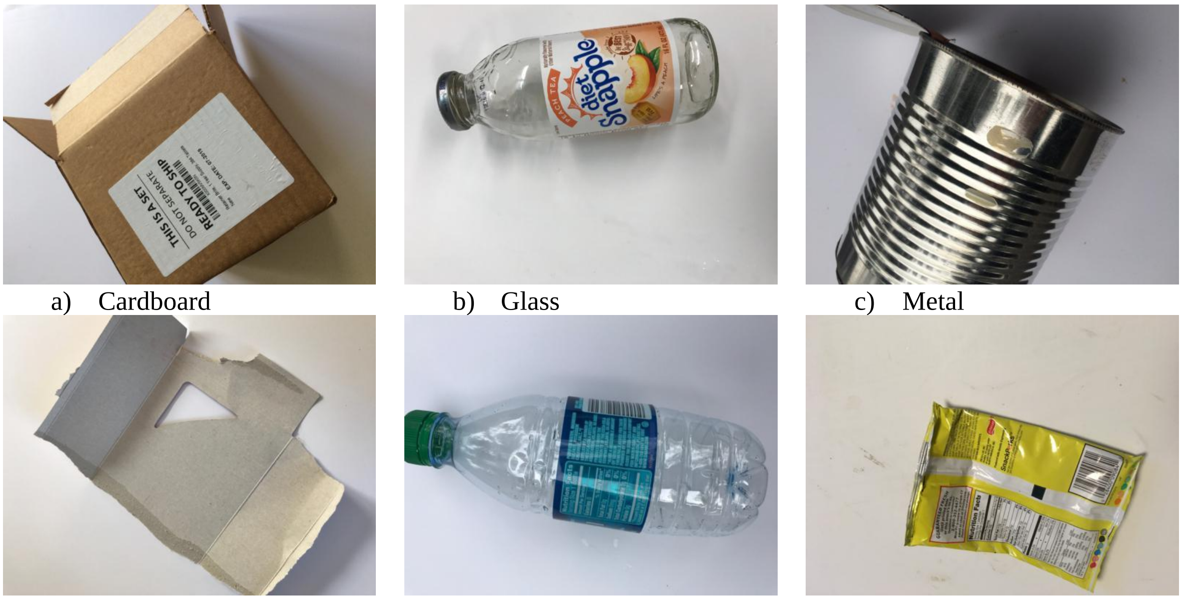
3. Materials and method

3.1 Dataset

The proposed study used the TrashNet [29] dataset. TrashNet data is divided into six categories: glass, paper, cardboard, plastic, metal, and other (garbage). The total number of images is 2527, with 501 in the glass category, 594 in the paper category, 403 in the cardboard category, 482 in the plastic category, 410 in the metal category, and 137 in the other category. The images in the dataset are 512x384 pixels in size. The dataset is 47 MB large. Figure 1 depicts some examples from the dataset.

The proposed model divides the dataset into training (80%) and test (20%) datasets. For the training, 323 cardboard, 401 glass, 328 metal, 476 paper, 386 plastic, and 110 garbage (other) images were used. Furthermore, 200 images representing 10% of the training images were used for validation. The test included 503 images, including 80 cardboard, 100 glass, 82 metal, 118 paper, 96 plastic, and 27 garbage (other). As a result, the proposed model's success was measured using data that had never been used before.

Data augmentation techniques were used to increase the number of images in the dataset. By doing this, the dataset was increased, and the model was kept from memorizing information during training. The employed techniques are as follows: 30-degree rotation (rotation_range), 0.2 percent shift (width_shift_range, height_shift_range), 0.2 shear_range, 0.2 zoom (zoom_range), and y-axis flip (horizontal flip). In this study, during the training phase, one of the specified data augmentation techniques was randomly applied to all images (the train dataset size) for each epoch. This approach ensures that the model sees different images in each epoch, thereby mitigating the problem of overfitting or memorizing the dataset. Data augmentation was not applied to the dataset before the training phase. In the test dataset, data augmentation techniques were not used.



a) Cardboard b) Glass c) Metal

d) Paper e) Plastic f) Other

Figure 1 Images from the TrashNet dataset

3.2 Convolutional neural network

Convolutional neural networks belong to a class of artificial neural networks. It is primarily employed in image processing and recognition applications. Using images as input, it learns the features of images through various layers. In this manner, it can perform image classification or object recognition.

CNNs have been used as image classifiers in most computer vision fields, requiring a simple and high-accuracy classifier [30]. In classical machine learning models, the problem-specific features were first determined manually. The feature vector was used to perform classification on the dataset. The biggest innovation that deep learning models add to this field is that feature vectors are generated automatically from the dataset during the training phase rather than manually. As a result, in CNN architectures, the number of filters and filter sizes in each convolutional layer is determined. During the training phase, the models will update these filter weights to determine the best features automatically for classification [6].

The AlexNet CNN was proposed in the ImageNet image recognition competition in 2012 and achieved the best performance [5]. This performance represents a breakthrough in computer vision. CNNs have since become the most widely used artificial neural network in image classification as a result of this research. CNN scans image segments and extracts features to recognize objects in those segments. Again, these features are used to define what the objects in the image are. CNN is built in layers, with each layer building on the previous layer's feature maps [5, 6]. It identifies objects in the image more accurately this way. A new CNN architecture is proposed in this study to classify waste.

3.3 Proposed model

Deep learning models can successfully classify the desired image according to different depth, width, and hyperparameter properties. Our aim in this study is to increase classification performance by designing these properties optimally. In the input layer of the proposed model, images were tested using 64x64, 128x128, 180x180, and 224x224 pixels. The most effective dimension for increasing the success rate was determined to be 180x180 pixels. As a result, 180x180 pixels images were used in the model's input layer. Figure 2 depicts the proposed model, which consists of four convolutional layers. The convolution layer’s filter numbers were determined to be 16, 32, 64, and 128, respectively. Each layer's filter (kernel) size was determined to be 3x3. Activation functions were used to learn about any continuous and complex relationship between network variables [31]. To avoid linearity in the network, the ReLu activation function was used after all convolution layers.

The pooling layer was used in the network to reduce the feature map dimensions. The largest value in the filter window determines the size of the filter window in max pooling; in average pooling, the average of all pixels in the window is kept as a single value in the output pixel [32]. The maximum pooling layer was added after each convolution layer in the proposed model. The BatchNormalization layer was added after the activation layer. However, it was discovered that this layer had no positive effect on the model's success and was removed. Flattening took place after the last pooling layer but before the fully connected layer. The multidimensional feature array was thus reduced to a one-dimensional array and fed into the fully connected layers [33]. As a result, the flattening process was carried out after the last max-pooling layer, with the addition of a flattening layer.

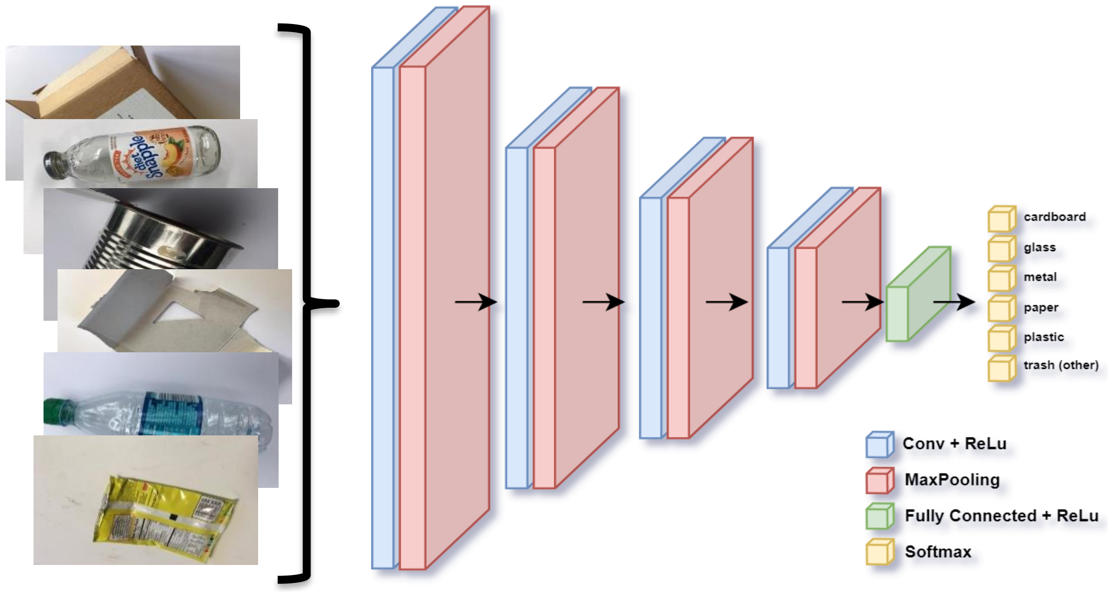


Figure 2 Proposed model

In CNN, the fully connected layer predicts the results using the features obtained by the convolution and pooling layers [33-35]. That is, the previous layers' output is used as the input in the fully connected layers. In this manner, it predicts the outcomes based on the features provided as input. The fully connected layer was represented by 512 units in our proposed model. The last layer added was the output layer. A dense layer with six neurons was added because the model will classify into six categories. The nonlinear Softmax activation function, which is used for multi-classification, was used in this layer. Softmax produces an output indicating the likelihood that the given input belongs to a class.

3.4 Performance metrics

The Confusion Matrix [36, 37] is a matrix created to interpret the results of a created model and to cross-examine the relationships between the actual and predicted values. This matrix contains four parameters.

• True Positive (TP): If a circumstance that is generally positive in the estimating process is projected to be positive.

• True Negative (TN): If the current situation is negative and the forecast is negative.

• False Positive (FP): If the current state is negative, but the estimating system expects a positive state.

• False Negative (FN): If the current state is positive and the estimator produces a negative outcome.

Accuracy, precision, recall, and F1-score metrics were used in the study [38-40]. Performance metrics are given in Equations 1-4, which are calculated from the confusion matrix.

$$\boldsymbol{Accuracy} = \frac{\boldsymbol{TP + TN}}{(\boldsymbol{TP + TN + FP + FN})} \tag{1}$$

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

$$F1\ Score = 2 * \frac{Recall * Precision}{Recall + Precision} \tag{4}$$

In cases where the class distribution of a dataset is imbalanced, it is necessary to consider the F1-score metric alongside accuracy for comparative purposes. To thoroughly assess the performance of a model, it is important to evaluate both precision and recall. The F1 score is a valuable metric that takes into account both of these measures. To summarize the model's performance in a more balanced manner, we utilize the F1-score, which is the harmonic mean of precision and recall values.

4. Experimentation and results

The TrashNet dataset was employed, utilizing pre-trained architectures from the ImageNet dataset. Initially, these architectures were loaded using the Keras deep learning framework, an open-source tool, with TensorFlow serving as the backend. Subsequently, the models were fine-tuned specifically for the TrashNet dataset. The experimental setup involved a Standard PC equipped with 16 GB RAM, an NVIDIA GeForce GTX 1080 Ti GPU with 11 GB memory, and an Intel i5-8400 processor operating at 2.80 GHz.

4.1 Hyperparameter optimization of the CNN architectures

In this study, many models with varying depth, width, and other hyperparameter values were developed. Table 1 displays the investigated hyperparameters and values. Table 2 shows the developed models with an accuracy value greater than 80% and the hyperparameter values for these models.

Table 1 Hiperparameters and values

Hyperparameter	Values
Number of convolutional layers (NCL)	4, 5, 6, 7, 9, 11, 13, 15
Number of filters (NF)	16, 32, 64, 128, 256, 512
Input Image Dimension, pixels (IID)	64, 128, 180, 224, 256, 512
Kernel size (KS)	3, 5, 7
Number of dense layers (NDS)	1, 2, 3, 4, 5
Epoch (Epo)	120, 150, 200, 300
Batch size (BS)	8, 16, 32
Optimizer (Opt)	Adam, Nadam, RmsProp, SGD, Adamax, Adadelta

The optimization strategy consists of generating CNNs with predetermined hyperparameters, training them iteratively, and identifying the optimal set of hyperparameters among these constructed networks. CNNs belong to the category of deep learning techniques that can learn end-to-end [5]. In the training phase, CNNs employ the backpropagation algorithm to update their filter weights and facilitate the learning process [6]. The training phase of CNNs is computationally expensive. In the case of waste classification, training a CNN network takes approximately 20 minutes. Considering all the hyperparameters in Table 1, there are a total of 181,440 different combinations of CNN networks. Training and comparing all these combinations would require a significant amount of time. Due to the large number of combinations, the Random Search method, which works on a limited number of combinations, can be applied. However, since Random Search does not remember past model results, we applied a different method. In this study, a prioritized Manuel Search approach was adopted by initially focusing on selected combinations based on our experience. Approximately 500 different CNN networks were trained, and out of these, 21 were reported.

In CNN architectures, increasing the depth (number of convolutional layers) and width (number of filters) hyperparameters enhances the model's learning capacity during the training phase [6]. However, as the models have more network weight parameters than the limited dataset, the risk of overfitting arises. To mitigate the problem of overfitting, regularization techniques such as L2 regularization, dropout, batch normalization, and data augmentation are employed [8]. However, these methods may not be sufficient for limited datasets. CNNs extract features from raw input images and perform classification tasks [6]. When we increase the size of input images, we often require deeper networks for better learning capacity.

In CNN architectures, the early layers learn basic features such as edges and color blobs, while the deeper layers learn more complex structures specific to the dataset [10]. In this study, we primarily focused on determining the depth of the model, the number of filters in each layer, and the size of the kernels for waste classification. Based on the obtained basic CNN architectures, we then observed the effects of other hyperparameters like optimizers, epochs, and batch sizes to achieve the optimal model. Increasing the depth and number of filters in a model leads to a higher success rate during the training phase. However, due to overfitting in the training phase, the performance of the model on unseen test datasets decreases. Deep models require a substantial amount of data to learn and generalize statistical patterns from the dataset during the training phase. Therefore, pre-trained models trained on large-scale datasets (such as ImageNet) with millions of samples have been successfully utilized through transfer learning in solving current problems.

Table 2 Experimental studies

M No	NCL	NF	KS	IID	NDL	Epoch	BS	Opt	Acc	F1 score	CPU time (ms)
1	5	16, 16, 32, 32, 128	3, 3, 3, 3, 3	180	2	150	16	SGD	0.80	0.80	334.80
2	7	16,16,16,32,64,128, 256	3, 3, 3, 5, 5, 5, 7	224	3	150	16	SGD	0.81	0.81	443.76
3	5	16, 32, 32, 64, 64	3, 3, 5, 5, 5	128	4	150	16	Nadam	0.81	0.80	261.22
4	6	16, 32, 32, 64, 64, 128	3, 3, 3, 5, 7, 7	128	4	150	16	Nadam	0.82	0.81	301.89
5	6	16, 16, 32, 64, 128, 256	3, 3, 3, 3, 3, 7	224	3	150	16	SGD	0.82	0.82	410.76
6	6	16, 32, 32, 64, 128, 256	3, 3, 3, 3, 7, 7	224	3	150	16	SGD	0.82	0.81	442.71
7	6	8, 16, 32, 64, 128, 256	3, 3, 3, 3, 7, 7	224	3	150	16	SGD	0.83	0.83	432.75
8	5	8, 16, 32, 64, 128	3, 3, 3, 3, 3	256	4	120	16	Adam	0.83	0.83	401.56
9	6	16, 16, 32, 64, 128, 256	3, 3, 3, 3, 3, 7	256	3	150	16	SGD	0.84	0.83	421.47
10	7	16, 16,16,32,64,128, 128	3, 3, 5, 5, 5, 7, 7	128	2	150	16	Adam	0.84	0.83	337.80
11	6	16, 32, 32, 32, 64, 128	3, 3, 3, 3, 5, 5	224	3	150	16	RmsProp	0.84	0.84	379.88
12	5	16, 16, 32, 32, 128	3, 3, 3, 5, 7	180	3	150	32	Adam	0.84	0.83	358.13
13	5	16, 16, 32, 32, 128	3, 3, 3, 5, 5	180	2	150	16	SGD	0.84	0.83	340.62
14	6	16, 16, 32, 64, 128, 256	3, 3, 3, 3, 3, 7	224	3	150	16	SGD	0.85	0.84	410.73
15	5	16, 32, 64, 128, 256	3, 3, 3, 3, 7	224	3	150	16	SGD	0.86	0.86	332.77
16	5	16, 32, 64, 128, 256	3, 3, 3, 3, 7	224	3	150	16	Adamax	0.86	0.86	349.82
17	6	16, 32, 64, 128, 256, 256	3, 3, 3, 3, 7, 7	224	3	150	16	Adamax	0.86	0.86	426.80
18	5	16, 32, 64, 128, 256	7, 3, 3, 5, 7	180	3	300	16	Adamax	0.87	0.87	351.76
19	5	16, 32, 64, 128, 256	7, 3, 3, 5, 7	180	3	300	16	Adam	0.87	0.87	370.76
20	5	16, 32, 64, 128, 256	7, 3, 3, 5, 7	180	1	300	16	Nadam	0.88	0.87	248.63
21	4	16, 32, 64, 128	3, 3, 3, 3	180	1	300	16	Nadam	0.89	0.88	239.86

M No: Model No, Acc: Accuracy, and other abbreviations are given in Table 1.

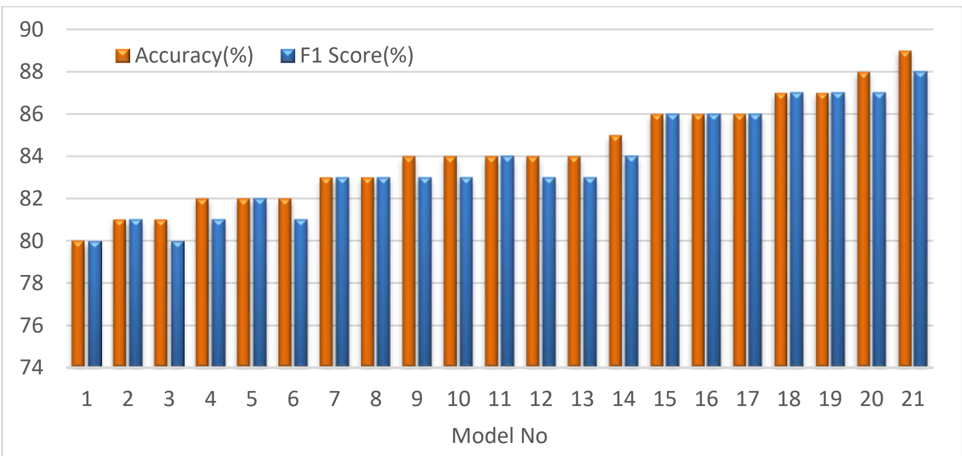


Figure 3 Accuracy and F1 score values of experimental studies for custom CNN models

The following study was carried out with the best-performing models among the derived models during the experimental studies shown in Table 2. The real-time classification time of models is important for waste classification. The best model, Model 21, achieved a classification time of 239.86 ms CPU time for a single image. The accuracy and F1 scores of the experimental studies can be analyzed in Figure 3. In the final proposed model (Model 21), four convolution layers were used to determine the best hyperparameters. Convolution layer filter numbers are 16, 32, 64, and 128, respectively. The size of the filters used in the layers is 3x3 on condition that all layers are equal. A 2x2 pooling (MaxPooling) layer was added after each conv layer. Because it produces better results in convolution layers, the ReLu activation function was used. In the classifier layer, the Softmax activation function was used. The epoch number was set to 300. Furthermore, the best performance was obtained with DenseNet169 architecture in the transfer learning study, with an accuracy of 96.42% and an F1 score of 96%.

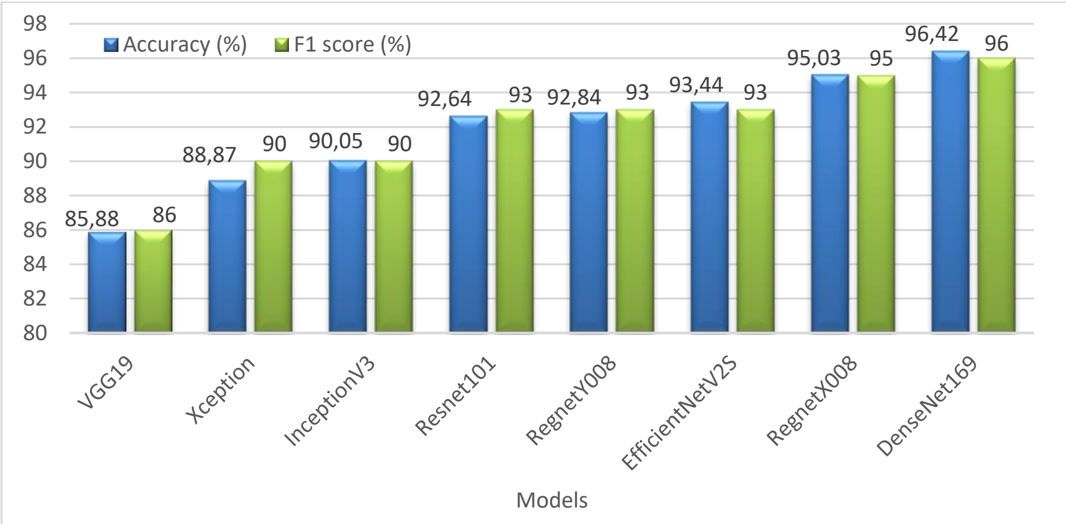


Figure 4 Accuracy and F1-scores of the state-of-the-art CNN models (with transfer learning and fine-tuning)

Figure 4 presents the accuracy and F1 scores of the state-of-the-art CNN models. Several state-of-the-art CNN models were trained through hyperparameter optimization. We keep the convolution layers of these models and remove the layers after the last convolution layer. After the last convolution layer, a Global Average Pooling, two dense and dropout layers are added. Finally, since we made a six-category classification, a dense layer with six neurons was added. Dense layers were optimized using different neuron numbers between 128 and 1024. In the dropout layer, the previous neurons were ignored with a ratio between 0.2 and 0.7, and the optimum ratio was found. In DenseNet169 architecture, the last fully connected layers were eliminated, a 256-node fully connected layer was created, and then a 128-node fully connected layer was added. A dropout with the 0.2 ratio was added after the 256-node fully connected layer. As a result, this architecture reduced the number of parameters from 12 million to about 5 million.

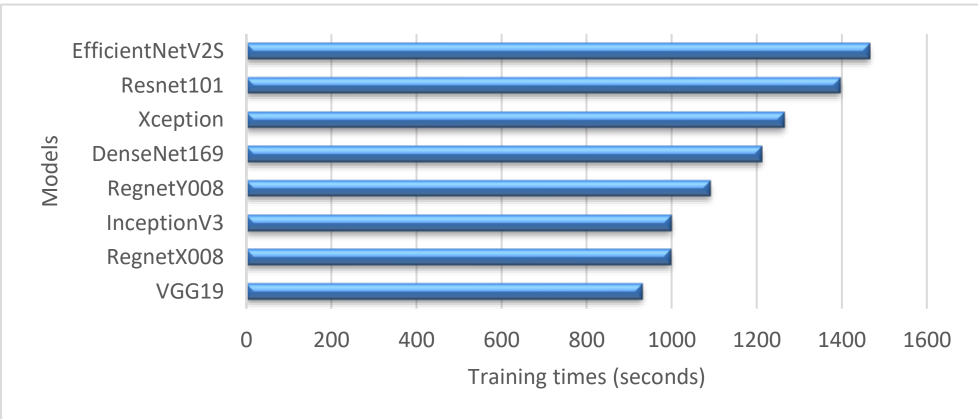


Figure 5 Training times of the state-of-the-art CNN models (with transfer learning and fine-tuning)

Figure 5 displays the training times of models on the TrashNet dataset using transfer learning. Since a large number of combinations were attempted in this study and the aim was to find the most optimal hyperparameter combination, training times became significant. Training the DenseNet169 model for 50 epochs takes approximately 20 minutes.

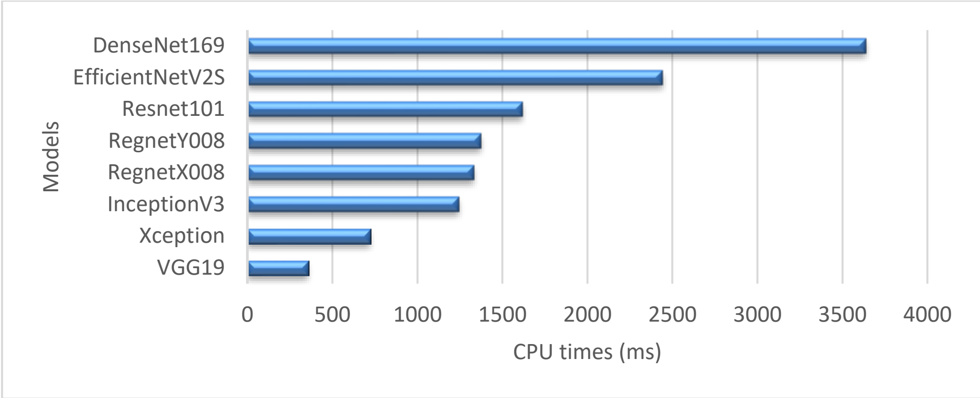


Figure 6 Prediction times (one image) of the state-of-the-art CNN models (with transfer learning and fine-tuning)

When examining Figure 6, it can be observed that the DenseNet169 model, which is the best-performing model, has a significantly longer inference time for predicting a single image compared to Model 21 in Table 2. The inference time for the DenseNet169 model to classify a single image is approximately 3.6 seconds. This duration is higher than the expected values for real-time implementations. Therefore, the proposed Model 21, which even predicts the class of an image in a shorter time (0.24 seconds) than the VGG19 model (0.36 seconds), can be utilized in waste management systems. Additionally, cross-platform support can be added as work moves forward. In this way, versions such as mobile and server-based systems can be derived [41, 42].

4.2 Confusion matrix

The confusion matrix of the proposed model is shown in Figure 7. When examined as an example in Figure 7, out of a total of 80 cardboard images in the test data, our model correctly predicted 72 of them. These 72 images correctly classified as Cardboard represent our true positive (TP) value. When analyzing the true labels in the horizontal row for Cardboard, our model made erroneous predictions for 4 images, classifying them as Glass instead of Cardboard. Similarly, it misclassified 2 images as Paper and 2 images as Plastic instead of correctly identifying them as Cardboard. These 8 images (4+2+2=8), which our model misclassified as Cardboard, represent the false negative (FN) value for our Cardboard class. Upon reviewing the column values for the Cardboard class in Figure 7, we can see the predictions made by our model. In these columns, our model erroneously labeled 3 images as Cardboard instead of Glass. Similarly, it misclassified 2 images as Cardboard instead of Plastic and 1 image as Cardboard instead of Trash. These 6 images (3+2+1=6) correspond to the false positive (FP) values for the Cardboard class. The same procedure is followed to interpret the confusion matrix values for the remaining classes, enabling the calculation of total FN and FP values for each class in Table 3. Precision, recall, and F1-score values for each class are computed by substituting the TP, FN, and FP values obtained from the confusion matrix into Equations 2-4.

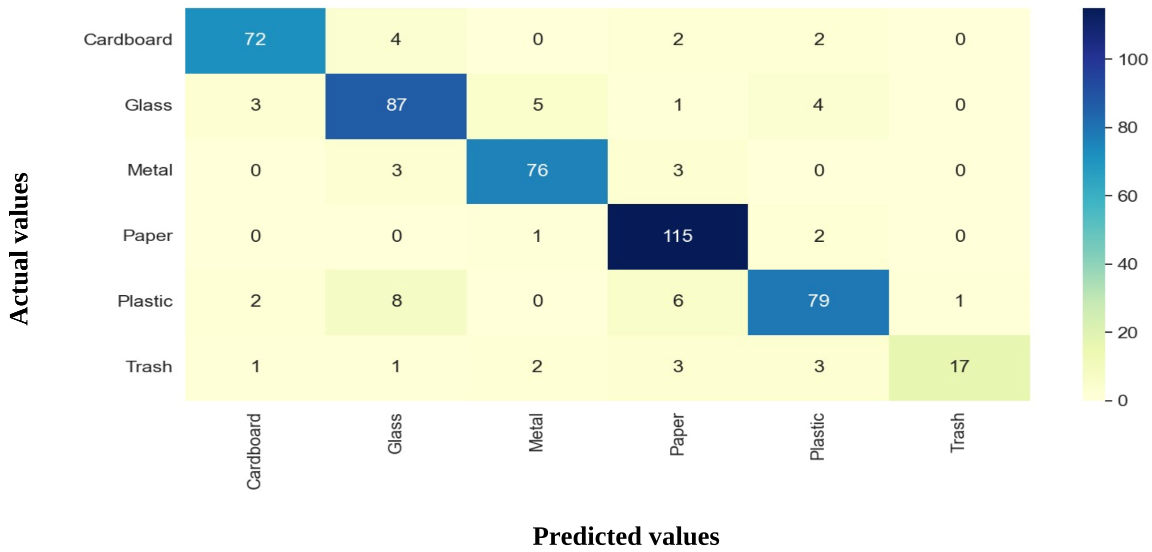


Figure 7 Proposed model confusion matrix

The confusion matrix of the study with the DenseNet169 architecture using the transfer learning method is presented in Figure 8. When the confusion matrix in both models is examined, it is seen that the most inaccurate cases are caused by the glass and plastic classifications.

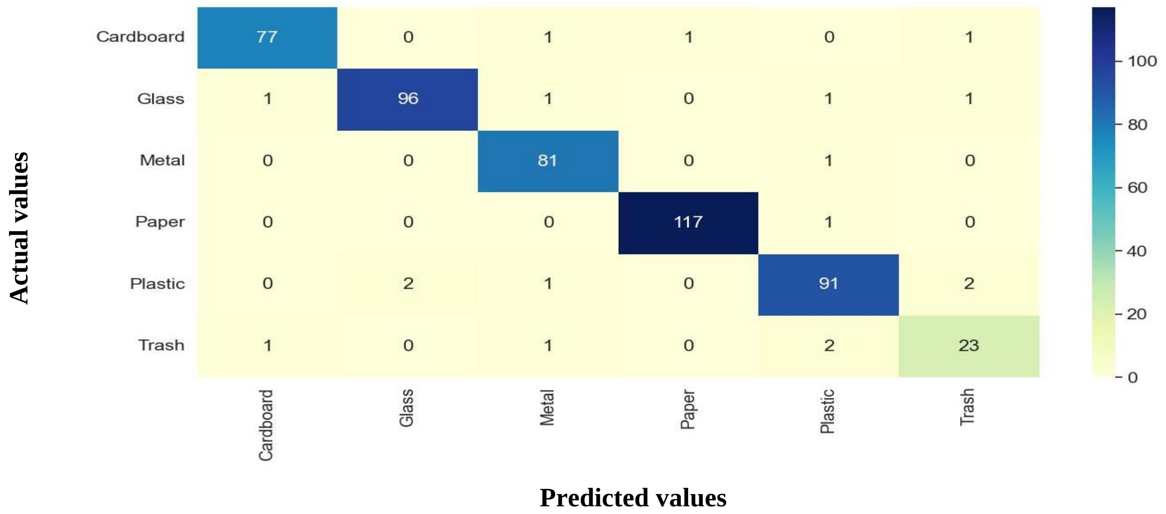


Figure 8 DenseNet169 confusion matrix

When the confusion matrix in both models is examined, the most incorrect cases are found in the glass and plastic classification. These two types of trash are quite similar to each other in some images. For this reason, it can be seen that the models are unstable in classifying them. For these two types of trash, solutions can be offered with data augmentation or handcrafted features. These handcrafted features can be integrated with CNN architecture.

4.3 Performance metrics

The proposed model was tested with 503 images. The model's performance metrics were measured as an accuracy of 88.66% and an F1 score of 88%. By utilizing the confusion matrix (see Section 4.2), we can substitute the TP, FN, and FP values obtained for each class into Equations 2-4 to determine the precision, recall, and F1-score values for each class. When we represent the F1-score using the True Positive (TP), False Positive (FP), and False Negative (FN) values, we obtain the following equation: F1 = TP / (TP + 0.5 (FP+FN)). Table 3 provides the calculated performance metric values for each class in multiclass datasets, utilizing the TP, FP, and FN values obtained from the confusion matrix table.

Table 3 Calculating precision, recall, and F1-score values for each class using the confusion matrix for model 21.

	TP	**FP**	**FN**	**Precision**	**Recall**	**F1-score**	**Support**
Cardboard	72	6	8	72/(72+6) = 0.92	72/(72+8) = 0.90	72 / (72 + 0.5 (6+8)) = 0.91	80 / 503 = 0.16
Glass	87	16	13	87/(87+16) = 0.84	87/(87+13) = 0.87	87 / (87 + 0.5 (16+13)) = 0.86	100 /503 = 0.20
Metal	76	8	6	76/(76+8) = 0.90	76/(76+6) = 0.93	76 / (76 + 0.5 (8+6)) = 0.92	82 /503 = 0.16
Paper	115	15	3	115/(115+15) = 0.88	115/(115+3) = 0.97	115 / (115 + 0.5 (15+3)) = 0.93	118/503 = 0.23
Plastic	79	11	17	79/(79+11) = 0.88	79/(79+17) = 0.82	79 / (79 + 0.5 (11+17)) = 0.85	96/503 = 0.19
Trash (Other)	17	1	10	17/(17+1) = 0.94	17/(17+10) = 0.63	17 / (17 + 0.5 (1 + 10)) = 0.76	27/503 = 0.05
Accuracy						0.89	503
Macro avg				0.90	0.85	0.87	503
Weighted avg				0.89	0.89	0.88	503

The results of the measured metrics are given in Table 3. To calculate the macro-averaged precision, recall, and F1 scores, the arithmetic mean (or unweighted mean) of the per-class F1 scores is computed. To calculate the weighted-averaged precision, recall, and F1 scores, the mean of all the per-class F1 scores is computed, considering each class's support. The individual metric value of each class is weighted according to the ratio of the number of images with their true labels. For example, according to the macro average, the F1 score value is calculated as (0.91+0.86+0.92+0.93+0.85+0.76)/6 = 0.87.

The performance metric results of the study with DenseNet169 were given in Table 4.

Table 4. DenseNet169 performance metrics

	Precision	Recall	F1-Score	Number of images
Cardboard	0.97	0.96	0.97	80
Glass	0.98	0.96	0.97	100
Metal	0.95	0.99	0.97	82
Paper	0.99	0.99	0.99	118
Plastic	0.95	0.95	0.95	96
Trash (Other)	0.85	0.85	0.85	27
-	-	-	-	-
Accuracy			0.96	503
Macro avg	0.95	0.95	0.95	503
Weighted avg	0.96	0.96	0.96	503

4.4 Accuracy and loss charts

When the accuracy graph shown in Figure 9 is examined, the training and test accuracy curves increase gradually. The validation curve follows the training curve somewhat from below. Accordingly, a small amount of memorization is observed, but the training of the model is successful as the validation curve follows the training curve in a parallel manner and there is no decrease in the validation curve in the following stages.

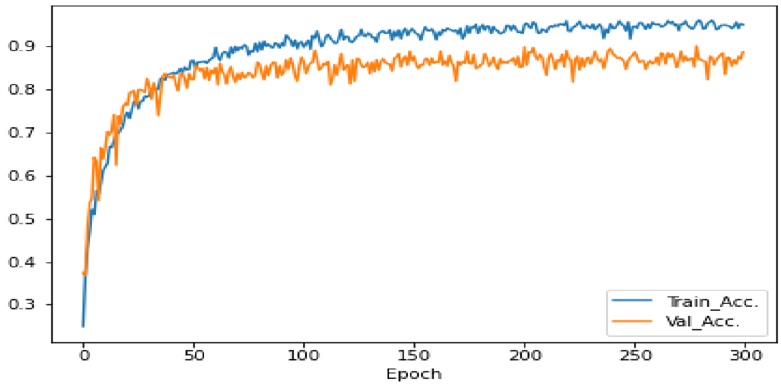


Figure 9. Training and test accuracy graph

In the loss graph in Figure 10, it is seen that the loss value decreases during the training and validation phases. In addition, the loss value in the validation phase follows the training phase from above. It has been tried to prevent memorization with Dropout methods.

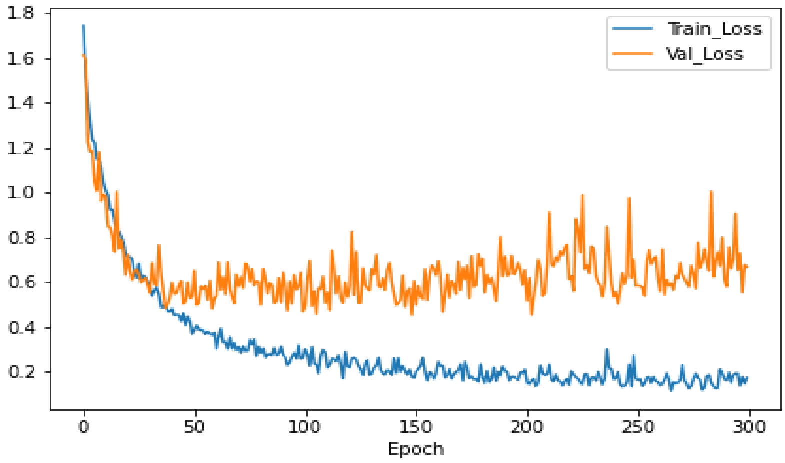


Figure 10. Training and test loss graph

Table 5 presents a comparison of the proposed model with existing studies. Most of the existing models have increased their model accuracy by transfer learning. Since state-of-the-art CNN models are very comprehensive models, good performances can be achieved in the desired dataset with the right hyperparameters. In this study, we achieved better results than existing studies with hyperparameter optimization. All existing studies in Table 5 have used the TrashNet dataset, and the performance values on the TrashNet test dataset have been presented.

Table 5 Comparison of the recommended model with existing CNN methods using the TrashNet dataset

Model	Year	Accuracy (%)	F1-Score (%)
Rabano et al. [18]	2018	87.2	-
Gyawali et al. [17]	2020	87.8	-
Lin et al. [20]	2022	88.8	88.9
Tran and Nguyen [25]	2022	90.71	-
Shi et al. [21]	2021	92.6	91
Yang et al. [22]	2022	93.72	-
Feng et al. [19]	2022	94.23	-
Wang et al. [14]	2021	94.24	94
Alrayes et al. [7]	2023	94.7	-
Bircanoğlu et al. [13]	2018	95	-
Aral et al. [15]	2018	95	-
Proposed Model	2023	96.42	96

5. Conclusion

Urban waste is a major issue in many countries around the world. Recycling is regarded as the most effective method of reducing urban waste. This study aims to enhance the effectiveness of waste classification in recycling through the optimization of various CNN models. In the study, a new CNN architecture was proposed for the classification of wastes in six categories using the TrashNet dataset. The proposed model has achieved an accuracy of 88.66% and an F1 score of 88%. Using the DenseNet169 architecture and transfer learning, an accuracy of 96.42% and an F1 score of 96% were achieved. This research can help city governments and recycling facilities classify waste and create an efficient waste management system. Thanks to the small size and fast operation of the model, waste classification can be performed without the need to transport it to facilities. This can be achieved using smart garbage containers equipped with devices like Raspberry Pi. Additionally, reducing the need for manual labor in recycling, it will contribute to safeguarding the health of individuals involved in this sector. New models will continue to be evaluated in future studies. We also aim to expand the dataset to categorize waste into more specific categories and increase the accuracy rate.

References

[1] S. Kaza, L. C. Yao, P. Bhada-Tata and F. Van Woerden, "A Global Snapshot of Solid Waste Management to 2050," 2018, [Online]. Available: https://elibrary.worldbank.org/doi/abs/10.1596/978-1-4648-1329-0. [Accessed: 15-Dec-2022].

[2] D. Hoornweg and P. Bhada-Tata, *What a Waste: A Global Review of Solid Waste Management,* World Bank, Washington DC USA, 2012.

[3] R. E. Sanderson, *Environmental Protection Agency Office of Federal Activities' Guidance on Incorporating EPA's Pollution Prevention Strategy into the Environmental Review Process*, EPA, Washington, DC, USA, 1993.

[4] O. Adedeji and Z. Wang, "Intelligent Waste Classification System Using Deep Learning Convolutional Neural Network," *Procedia Manufacturing*, vol. 35, pp. 607-612, 2019.

[5] A. Krizhevsky, I. Sutskever and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Proc - Neural Information Processing System Conference*, pp. 1-9, 2012.

[6] Y. LeCun, Y. Bengio, & G. Hinton. "Deep learning," *Natur*e, vol. 521, pp. 436-444, 2015.

[7] F. S. Alrayes et al., "Waste classification using vision transformer based on multilayer hybrid convolution neural network," *Urban Climate*, vol. 49, pp. 1-14, 2023.

[8] 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.1, pp. 1929-1958, 2014.

[9] C. Tan, F. Sun, T. Kong, W. Zhang and C. Y. &. C. Liu, "A survey on deep transfer learning," *Proc. - 27th International Conference on Artificial Neural Networks,* pp. 270-279, 2018.

[10] J. Yosinski, C. Jeff , B. Yoshua ve L. Hod, "How transferable are features in deep neural networks?," *Advances in neural information processing systems*, 2014.

[11] K. He, X. Zhang, S. Ren and J. Sun, "Deep residual learning for image recognition" *Proc – IEEE conference onComputer Vision and Pattern Recognition*, pp. 770-778, 2016.

[12] G. Huang, Z. Liu, L. v. d. Maaten and K. Q. Weinberger, "Densely Connected Convolutional Networks," *Proc - IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4700-4708, 2017.

[13] C. Bircanoğlu, M. Atay, F. Beşer, Ö. Genç, & M. A. Kızrak, "RecycleNet: Intelligent waste sorting using deep neural networks," *Proc - 2018 Innovations in intelligent systems and applications,* pp. 1-7, 2018.

[14] C. Wang, J. Qin, C. Qu, X. Ran, C. L. b and B. Chen, "A Smart Municipal Waste Management System Based on Deep-Learning and Internet of Things," *Waste Management*, vol. 135, pp. 20-29, 2021.
[15] R. A. Aral, Ş. R. Keskin, M. Kaya and M. Hacıömeroğlu, "Classification of TrashNet Dataset Based on Deep Learning Models," *Proc - International Conference on Big Data*, pp. 2058-2062, 2018.
[16] Q. Zhang, Q. Yang, X. Zhang, Q. Bao, J. Su and X. Liu, "Waste image classification based on transfer learning and convolutional neural network," *Waste Management*, vol. 135, pp. 150-157, 2021.
[17] D. Gyawali, A. Regmi, A. Shakya, A. Gautam and S. Shrestha, "Comparative Analysis of Multiple Deep CNN Models for Waste Classification," 2020, [Online]. Available: https://arxiv.org/abs/2004.02168. [Accessed: 10-Dec-2022].
[18] S. L. Rabano, M. K. Cabatuan, E. Sybingco, E. P. Dadios and E. J. Calilung, "Common Garbage Classification Using MobileNet," *Proc - IEEE 10th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management*, pp. 1-4, 2018.
[19] Z. Feng, Yang, J., Chen, L., Chen, Z., & Li, L., "An Intelligent Waste-Sorting and Recycling Device Based on Improved EfficientNet," *International Journal of Environmental Research and Public Health*, vol. 19, no. 23, pp. 1-18, 2022.
[20] K. Lin, Zhao et al, "Applying a deep residual network coupling with transfer learning for recyclable waste sorting," *Environmental Science and Pollution Research*, vol. 29, no. 60, pp. 91081-91095, 2022.
[21] C. Shi, C. Tan, T. Wang and L. Wang, "A Waste Classification Method Based on a Multilayer Hybrid Convolution Neural Network," *Applied Science*, vol. 11, no 18, pp. 1-19, 2021.
[22] Z. Yang, Xia, Z., Yang, G., & Lv, Y. "A Garbage Classification Method Based on a Small Convolution Neural Network," *Sustainability*, vol. 14, no. 22, pp. 1-16, 2022.
[23] J. Bobulski and M. Kubanek, "Deep Learning for Plastic Waste Classification System," *Applied Computational Intelligence and Soft Computing*, vol. 2021, pp. 1-7, 2021.
[24] J.-R. Riba, R. Cantero, P. Riba-Mosoll and R. Puig, "Post-Consumer Textile Waste Classification through Near-Infrared Spectroscopy, using an Advanced Deep Learning Approach," *Polymers*, vol. 14, no. 12, pp. 1-14, 2022.
[25] B. G. Tran, & D. L. Nguyen. “Simple and Efficient Convolutional Neural Network for Trash Classification,” *Proc - Annals of Computer Science and Information Systems*, pp. 255-260, 2022.
[26] N. C. A. Sallang, M. T. Islam, M. S. Islam and H. Arshad, " A CNN-Based Smart Waste Management System Using TensorFlow Lite and LoRa-GPS Shield in Internet of Things Environment," *IEEE Access*, vol. 9, pp. 153560-153574, 2021.
[27] D. O. Melinte, A.-M. Travediu and D. N. Dumitriu, "Deep Convolutional Neural Networks Object Detector for Real-Time Waste Identification," *Applied Sciences*, vol. 10, no. 20, pp. 1-18, 2020.
[28] P. Nowakowski and T. Pamula, "Application of Deep Learning Object Classifier to Improve E-waste Collection Planning" *Waste Management*, vol. 109, pp. 1-9, 2020.
[29] M. Yang, and G. Thung, "Classification of trash for recyclability status." CS229 project report 2016.1 (2016): 3.
[30] F. Hu, G.-S. Xia, J. Hu and L. Zhang, "Transfering Deep Convolutional Neural Networks for the Scene Classification of High-Resolution Remote Sensing Imagery," *Remote Sensing*, vol. 7, no. 11, pp. 14680-14707, 2011.
[31] R. Jain, P. Nagrath, G. Kataria, V. S. Kaushik, & D. J. Hemanth, "Pneumonia detection in chest X-ray images using convolutional neural networks and transfer learning," *Measurement*, vol. 165, pp. 1-10, 2020.
[32] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, Y. Duan, O. Al-Shamma, J. Santamaría, M. A. Fadhel and M. A.-A. &. L. Farhan, "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," *Journal of Big Data*, vol. 8, pp. 1-74, 2021.
[33] S. K. Sundararajan, B. Sankaragomathi ve D. S. Priya, "Deep Belief CNN Feature Representation Based Content Based Image Retrieval for Medical Images," *Journal of Medical Systems*, vol. 43, pp. 1-9, 2019.
[34] G. Li and N. Li, "Customs classification for cross-border e-commerce based on text-image adaptive convolutional neural network," *Electronic Commerce Research*, vol. 19, pp. 779-800, 2019.
[35] X. Y. Wu, "A hand gesture recognition algorithm based on DC-CNN," *Multimedia Tools and Applications*, vol. 79, pp. 9193-9205, 2020.
[36] S. V. Stehman, "Selecting and interpreting measures of thematic classification accuracy," *Remote Sensing of Environment*, vol. 62, no .1, pp. 77-89, 1997.
[37] S. M. Piryonesi and T. E. El-Diraby, "Data Analytics in Asset Management: Cost-Effective Prediction of the Pavement Condition Index," *Journal of Infrastructure Systems*, vol. 26, no. 1, pp. 1-25, 2020.
[38] D. M. W. Powers, "Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness & Correlation," *Journal of Machine Learning Technologies*, vol. 2, pp. 37-63, 2011.
[39] K. M. Ting, C. Sammut and G. I. Webb, *Encyclopedia of machine learning*, New York: Springer Science & Business Media, 2011.
[40] M. Talo, U. B. Baloglu, Ö. Yıldırım and U. R. Acharya, "Application of deep transfer learning for automated brain abnormality classification using MR images," *Cognitive Systems Research*, vol. 54, pp. 176-188, 2019.
[41] Y. Çetin-Kaya, M. Kaya & A. Akdağ, "Route Optimization for Medication Delivery of Covid-19 Patients with Drones," *Gazi University Journal of Science Part C: Design and Technology*, vol. 9, no. 3, pp. 478-491, 2021.
[42] M. Kaya, and Y. Çetin-Kaya, "Seamless computation offloading for mobile applications using an online learning algorithm," *Computing*, vol. 103, no.5, pp. 771-799, 2021.

Conflict of interest
The author declares that there are no potential conflicts of interest.

Funding
This research did not receive a specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Data availability
The datasets generated during and/or analyzed during the current study are available from the corresponding author upon reasonable request.

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.