Sakarya University Journal of Computer and Information Sciences
http://saucis.sakarya.edu.tr/

DOI: 10.35377/saucis...1170902 RESEARCH ARTICLE

Deep Learning-based Road Segmentation & Pedestrian Detection System for Intelligent Vehicles

Gozde Yolcu Oztel [1], Ismail Oztel [2]

[1]Software Engineering Department, Faculty of Computer and Information Sciences, Sakarya University, Sakarya, Türkiye
[2]Computer Engineering Department, Faculty of Computer and Information Sciences, Sakarya University, Sakarya, Türkiye

Corresponding author:
Gozde Yolcu Oztel, Software Engineering Department, Faculty of Computer and Information Sciences, Sakarya University, Sakarya, Türkiye
E-mail address:
gyolcu@sakarya.edu.tr

Received: 4 September 2022
Revised: 23 February 2023
Accepted: 24 February 2023
Published Online: 30 April 2023

Citation: Oztel G and Oztel I.. (2023). Deep Learning-based Road Segmentation & Pedestrian Detection System for Intelligent Vehicles. *Sakarya University Journal of Computer and Information Sciences*. 6(1) https://doi.org/10.35377/saucis...1170902

ABSTRACT

Correctly determining the driving area and pedestrians is crucial for intelligent vehicles to reduce fatal road accident risk. But these are challenging tasks in the computer vision field. Various weather, road conditions, etc., make them difficult. This paper presents a vision-based road segmentation and pedestrian detection system. First, the roads are segmented using a deep learning-based consecutive triple filter size (CTFS) approach. Then, pedestrians on the segmented roads are detected using an object detection approach. The CTFS approach can create feature maps for small and big features. According to the experiments, the segmentation and object detection results have achieved successful results compared to the literature. The Jaccard index value is 95.84% for segmentation and the average precision value is 65.50% for the people detection task. The proposed system is a low-cost road segmentation and pedestrian detection system for intelligent vehicles.

Keywords: Pedestrian detection, road segmentation, convolutional neural networks, intelligent vehicles, deep learning

1. Introduction

Intelligent transportation systems have been very popular recently. These systems allow efficient and safe planning of the traffic. According to International Road Federation, the goals of Intelligent Transportation Technologies are as follows: (1) Safe roads and driving. (2) Providing sustainable road transportation. (3) Collecting the data. (4) Transferring, processing, and analyzing the data. (5) Smart decision. These technologies can be applied in many fields like vehicle navigation [1], [2], traffic signal control systems [3], [4], [5], and pedestrian detection systems [6], [7]. This technology can be helpful for all humanity, especially visually impaired and disabled people.

To ensure safety, first, the intelligent vehicle's road must be correctly determined. On the other hand, road segmentation is difficult to work in the computer vision field. Because different road and weather conditions, etc., make these tasks difficult. Potholes on the roads may reduce the segmentation performance. Also, for efficient traffic management, roads are designed with various patterns like rectangular, radial, and hexagonal. So, the roads contain small and big features in terms of computer vision. Correctly mapping these features is significant for a high-performance system.

In order to prevent loss of life, the pedestrians on the road must also be correctly determined. Pedestrian detection on the road is also a challenging task because pedestrians may wear clothes in a wide variety of colors. Also, some environmental factors such as light and/or weather conditions, and complex backgrounds can camouflage the pedestrians. In addition, in some environments, the pedestrian body may not be completely observed due to the occlusions.

This study presents a deep learning-based road segmentation and pedestrian detection system. The system first segments road images taken by a vehicle-mounted camera. Then, the system starts to detect pedestrians on the segmented road images. The main contribution of the study is given below:

1) The low-cost proposed system allows robust road segmentation and pedestrian detection tasks using a simple camera.

2) A deep learning-based multi-task system is developed, which applies both segmentation and detection tasks.

The system architecture is shown in Figure 1. As can be seen in the figure, the system includes two subsystems (a and b). The first subsystem(a) has been trained to segment road images as road vs. background. In this stage, a CNN-based algorithm has been developed. The second subsystem(b) has been trained for the human detection task. For this purpose, a pre-trained YOLOv7 network has been used. In the realization stage, the system takes an image using a vehicle-mounted camera. Then, this image is used as an input for both trained subsystems, and results have been obtained. Both system results have been combined in a single final image.

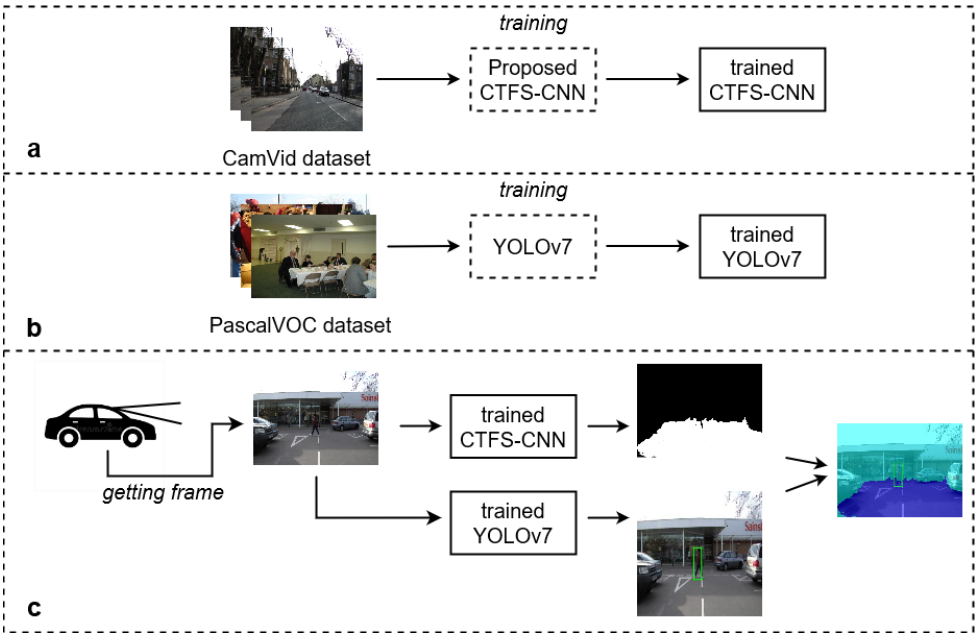

Figure 1 The system architecture: training step for segmentation task (a) and human detection task (b), system pipeline (c)

2. Related Works

2.1 Road segmentation

Many road segmentation studies used radar [8], laser scanners [9], stereovision [10], etc. for detecting road markings or boundaries.

Vision-based road segmentation studies are mainly based on three approaches, namely feature-based, model-based, and neural network-based approaches [11].

Feature-based approaches use color, texture, and edge information to detect and segment the road area [12], [13]. Although this method has the advantage of not requiring prior knowledge of the road shape, it is quite disadvantageous in road conditions such as shadow and water [14]. According to [15], studying road detection using texture information has two disadvantages. One of them is the strong perspective effect of road scenes. Also, random aperiodic textures usually are presented by roads, and these are not easily characterizable.

Model-based approaches build a road model taking into account the shape of the road. They extract the lanes with edge detection and match the lanes with the road model. The disadvantage of this approach is establishing a road model is quite difficult [11].

Neural network-based approaches perform using input and output data. If there is sufficient input data, it can produce successful results.

Recently, deep learning-based approaches have been studied in road segmentation tasks. In [16], the authors modified the structure of a deep network for an effective process in the case of memory and running time. In [15], the authors proposed a CNN approach to segment road scene images. In [17], the authors proposed a Siamese deep neural network based on FCN-8s to detect the road region. They collected the data from a LIDAR sensor and a monocular camera. In [17], the authors proposed a supervised deep Auto-Encoder model for road segmentation tasks. In [18], the authors proposed a CNN model to distinguish different image patches.

2.2 Human detection

Human detection systems are basically developed in three steps. Firstly, the regions which are potentially covered by human components are extracted. Secondly, extracted regions are described. The last step is the classification process for human vs. non-human areas.

In the literature, three basic feature extraction methods have been used for human detection. These are shape-based, appearance-based, and motion-based methods [19]. Shape-based methods use edge-based features for detecting human objects. In appearance-based methods, color and texture information is used. If the object's motion patterns are different, motion monitoring can be used to discriminate objects. For detecting human motion, firstly, temporal features are determined using the temporal difference or optical flows. If the human features are detected, the classification step starts [19].

Also, deep learning has been mostly used for human detection tasks [20][21]. In the proposed study, the pedestrian detection task has been performed using a deep transfer learning approach.

3. Methodology

Recently, deep learning has been popular owing to the increasing data sources and improvement of powerful computer equipment. Convolutional Neural Network (CNN) is a deep learning method that shows successful results in computer vision problems. CNNs can be designed using a different combination of convolution, pooling, ReLU, fully connected, softmax etc. layers.

In convolution layers, filters operate on the input images. The mathematical definition of convolution is given in Equation (1) [22].

$$S(i,j) = (I * K)(i,j) = \sum_{m}\sum_{n} I(i+m, j+n)K(m,n) \quad (1)$$

In the Equation: K is a kernel, I is an input image, S is the production result after the convolution operation, and i, j, m, n are the index values.

The pooling layer is not an obligatory layer. It can be used to reduce the input size. It causes a reduction in the size of width and height [23]. Thus, less processing load is provided for the next layers. Also, the pooling layer can help reduce overfitting.

In [24], the authors suggest using a non-linear activation function for quick learning in CNNs. In this paper, ReLU is used for this purpose. The Mathematical Equation of ReLU is given in Equation (2). In the Equation, x is the input of the activation function.

$$f(x) = max(0, x) \quad (2)$$

In order to avoid overfitting in the CNNs, the dropout layer is used [25]. In this layer, some nodes are deleted. Thus, dependence on a particular neuron is prevented. The fully connected layer connects to all nodes in the previous layer. It is used before the classification layer.

The CNNs are trained based on the feed-forward technique. In this technique, information is transmitted through the network. After that, the error value is calculated based on the produced result and the corresponding target. In the proposed study, the error is calculated using the Least Mean Square Error (LMSE). The mathematical definition of LMSE is given in Equation (3) [22].

$$J(w) = \frac{1}{2}\sum_{k=1}^{c}(t_k - z_k)^2 \quad (3)$$

In the Equation, t is the target vector, z is the produced result vector, t and z are vectors with the length of c, w represents the weights, and k represents the index.

In order to increase the network segmentation performance based on the error value Backpropagation Algorithm is used.

3.1 Road segmentation using Consecutive Triple Filter Size Approach - CTFS

Roads may include different patterns such as rectangular, radial, etc. Thus, they have different sizes of features. For effective road segmentation, these features must be correctly detected. For this purpose, in the proposed system, a CNN architecture is designed that contains consecutive 3x3, 5x5, and 7x7 filters in convolution blocks. The structure of convolutional layers with the CTFS approach is shown in Figure 2. In the proposed network, the properties of all consecutive triple convolution blocks are the same. These properties are given in Table 1.

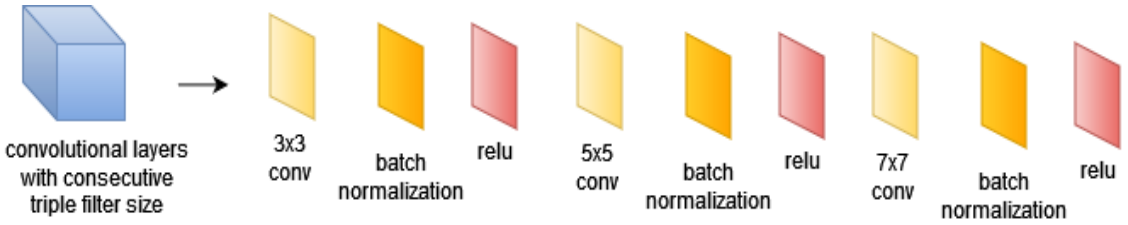

Figure 2 The structure of convolutional layers with CTFS

Table 1 Layer properties of proposed CTFS blocks

Type of layer	#Filter	Filters	Padding
convolution	64	3x3	1
convolution	64	5x5	2
convolution	64	7x7	3

The proposed CNN architecture contains 2 maxpool layers with 2x2, 5 CTFS blocks, 2 transposed-convolution layers with 4x4 size 64 filters, 1 fully connected layer, 1 normalization layer, 1 softmax and 1 pixel classification layer. It also includes 15 batch normalization layers after each convolution layer, and 15 relu layers after each batch normalization layer. The proposed CNN architecture is illustrated in Figure 3.

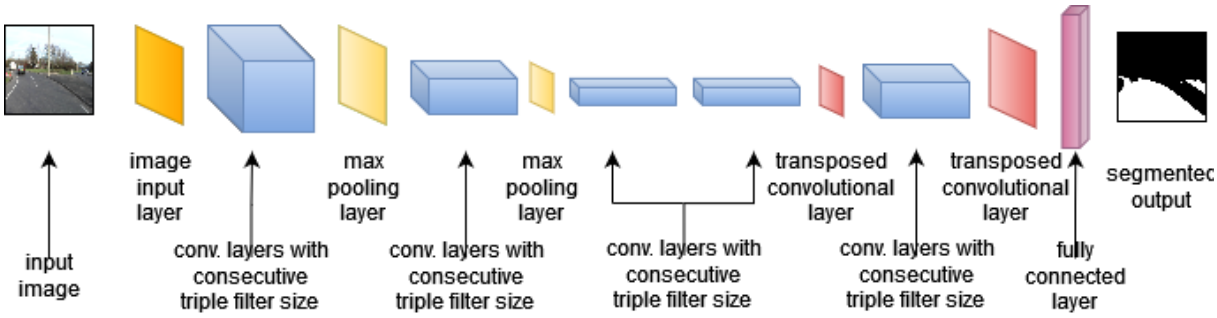

Figure 3 Proposed CNN architecture

Road segmentation is considered as a binary classification problem. Thus, the image pixels have been classified as background vs. road. As the learning rate of 0.001 is used.

The proposed network also includes an encoder-decoder subnetwork. With the encoder, the inputs are downsampled; then, with the decoder, the output of the encoder is upsampled.

3.2 Pedestrian detection using YOLOv7

The YOLO [26], [27] is an object detection network. In the YOLO, the detection problem is structured as a regression problem. The bounding box coordinates and probabilities of each class are produced through regression.

In this network, training images are divided into S× S grids. If the center of the target ground truth is in a grid; this grid is responsible for detecting the target.

Grids predict bounding boxes, their confidence scores, and conditional probabilities of the classes [28]. The mathematical definition of confidence is shown in Equation (4) [28]:

$$confidence = p_r(object) x IoU_{pred}^{truth}, p_r(object) \in \{0,1\} \quad (4)$$

If the target is in a grid p_r(object)=1; otherwise, it becomes 0. In order to represent the confidence between the reference and the predicted bounding box, IoU_{pred}^{truth} is used. The confidence shows if the grid includes objects. If it includes objects, the confidence also represents the predicted bounding box accuracy.

In July 2022, the YOLO family introduced a new version called YOLOv7. [29] claims that this version is the fastest and most accurate real-time object detector to date. The E-ELAN is the computational block in the YOLOv7 backbone. This E-ELAN architecture enables the framework to learn better. It has been designed by analyzing the performance factors such as impact speed and accuracy. YOLOv7 is based on a compound model scaling approach. In YOLOv7, width and depth are scaled in coherence for concatenation-based models.

After the training stage, the re-parameterization method is used to improve the model. Due to this operation, training time may increase, but results can be better. There are 2 types of re-parametrizations: Model level and Module level. For model level re-parametrization, there are 2 options: (1) Train multiple models with different training data but the same settings. Then average weights are calculated to obtain the final model. (2) Calculate the average weights of models at different epochs. In Module level re-parameterization, the model training process is divided into multiple modules. The outputs are combined to obtain the final model [29].

Classic YOLO architecture includes a backbone, a neck, and a head. The head includes the predicted outputs. Differently, YOLOv7 has multiple heads. The head responsible for the final output is called as Lead Head. The head which helps with training in the middle layers is called as Auxiliary Head. Using an assistant loss, the auxiliary heads' weights are updated. Thus, the model learns better [29].

4. Experimental Results

4.1 Database

For the road segmentation task, The Cambridge-driving Labeled Video Database (CamVid) [30] was used. The dataset contains 701 images that were taken by a driving automobile. The database has the original frames and their corresponding labeled frames. There are few images of people on the road in this database. Thus, for pedestrian detection, training could not be done in this database.

The pedestrian detection system has been trained using the last version of the Pascal VOC dataset [31]. This dataset includes four main labels. These are person, vehicle, animal, and indoor. For the pedestrian detection task, the person labels with corresponding coordinate information have been selected. The pedestrian detection system has also been visually tested using CamVid images.

The Pascal VOC dataset contains data with 20 labels. The Roboflow platform [32] was used to use only 'person' labeled data. Also, a YOLOv7 repository [33] was used to train and test the process for person detection.

4.2 Experiments

In the proposed system, firstly road segmentation task has been applied. Using the proposed CTFS approach, the system has been trained and tested with CamVid road images and their ground truths.

Then, the pedestrian detection task has been applied. For pedestrian detection tasks, the system has been trained using YOLOv7. YOLOv7 has been trained with the Pascal VOC dataset for the human detection task. This trained network has been used for CamVid.

Some visual results from CamVid have been shown in Figure 4. In this figure, columns a, and b show original images and their ground truths, respectively. Column c illustrates the segmentation results. Column d shows pedestrian detection results. Finally, column e shows the complete system result. As seen, the system produced visually successful results.

In Figure 5, the automated road segmentation result for a test image has been compared to its ground truth. In this figure, black pixels indicate true negatives (TN), yellow pixels show true positives (TP), red pixels represent false negatives (FP) and green pixels are true negatives (FN). As seen in the figure, FP and FN are mostly produced in boundary fields that separate the two classes. Because the ground truths are manually created in the database, a small margin of error in these fields can be expected.

To better examine the benefits of the proposed consecutive triple filter size approach, an object detection method has been applied to the same road segmentation task. Vgg16 [34] pre-trained network has been adapted to the road segmentation task and retrained. Finally, the results have been compared.

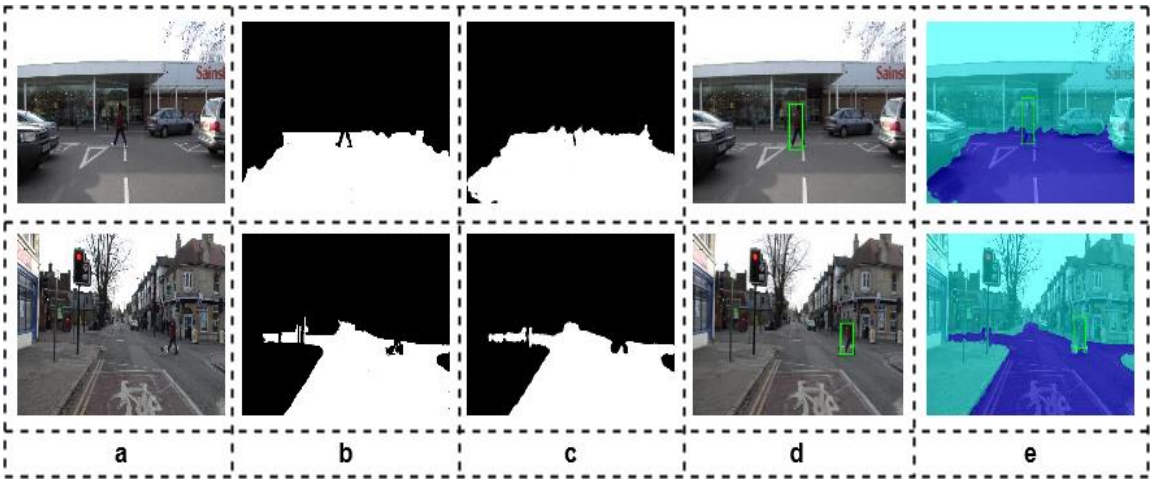

Figure 4 Visual results from the pedestrian detection system. a: original image b: ground truth for road segmentation c: road segmentation result d: pedestrian detection result e: complete system result

To evaluate the proposed CTFS approach, different criteria indexes have been used. These are False Positive Rate (FPR), TPR (True Positive Rate), and IoU (Intersection-Over-Union) metrics. To evaluate the pedestrian detection part; the Average precision (AP) metric has been used. Their mathematical definition is given in Equations (5), (6), (7), (8) and (9).

$$FPR = \frac{FP}{FP + TN} \tag{5}$$

$$TPR\ (Recall) = \frac{TP}{TP + FN} \tag{6}$$

$$IoU = \frac{TP}{TP + FP + FN} \tag{7}$$

$$Precision(P) = \frac{TP}{TP + FP} \tag{8}$$

$$AP = \sum_n (R_n - R_{n-1)} P_n \tag{9}$$

where P_n and R_n are the precision and recall at the nth threshold.

Figure 5 Visual results from the pedestrian detection system

Table 2 compares the success rate of the CTFS approach with other studies that use the same CamVid dataset. In the different literature studies, different number of classes were used according to their own scenario. The success rates which are reported

in Table 2 belong to the" road" label success rate. These are obtained from the study's confusion matrices. As can be seen in Table 2, the proposed study shows the best performance in terms of the FPR index. In addition, the proposed system outperformed most of the approaches and ranked second behind Vgg16 in terms of IoU and TPR indexes. Vgg16 is a pre-trained network, and it was trained with more than one million images. In Vgg16, just 3x3 filter size was used. Although the proposed CNN architecture was trained with fewer data (1200 images), it produced a close result to Vgg16. Also, for higher quality, the number of data can be increased, and the methods presented in [35] can be implemented to speed up the runtime.

Table 2 Performance comparison of the road segmentation subsystem to other studies that used the same CamVid dataset.

Approach	FPR	TPR	IoU	Inference time
K means [36]	15.80	78.6	63.50	0.25 s
Supervised Deep AE [36]	3.30	97.1	95.40	0.03s
Resnet [37]	-	-	75.8	-
HyperSeg [38]	-	-	78.1	-
Vgg16 (transfer learning)	3.05	97.94	95.98	1.45s
Proposed approach	2.34	97.86	95.84	0.88 s

The comparison table of the pedestrian detection part is shown in Table 3. As can be seen in this table, the proposed pedestrian subsystem gives successful results in terms of the average precision index. Also, the precision-recall curve of the pedestrian detection part is given in Figure 6. A high area under the curve shows high recall and high precision.

Table 3 Performance comparison of the people detection subsystem to other studies that used the same Pascal VOC dataset.

Method	Av. precision	Inference time
HOG III Feature [39]	52.10	2s
G Feature [40]	51.30	-
Fusion of G and T Feature [41]	52.10	-
Faster R-CNN [42]	65.00	-
YOLOv7	65.50	0.2ms

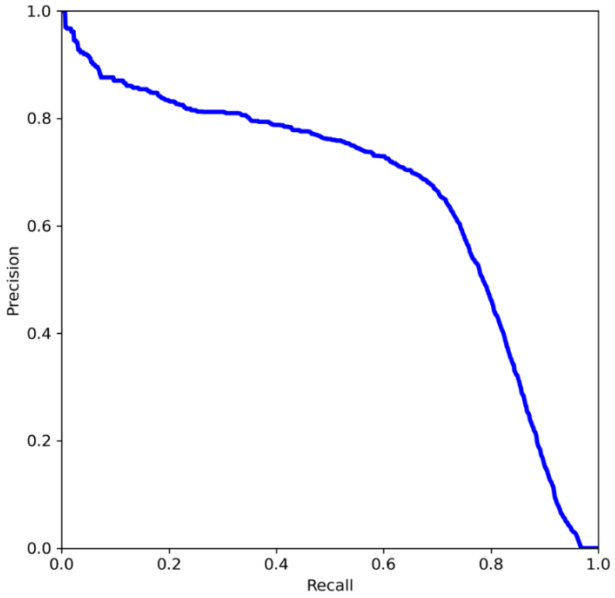

Figure 6 Precision-recall curve of the pedestrian detection task

5. Conclusions

In this study, a deep learning-based road segmentation and pedestrian detection system for intelligent vehicles has been presented. The system includes two main steps. In the first step, the system segments road images and detects roads. In the second step, it detects pedestrians on the roads.

For the road segmentation task, a deep learning-based consecutive triple filter size approach has been developed. This model has been trained and tested using the CamVid dataset. Owing to consecutive 3x3, 5x5 and 7x7 filter blocks, small and big features have been mapped. Also, Vgg16 was retrained for the road segmentation task using the transfer learning method.

Then, the results of both networks were compared. Although Vgg16 was trained using more than one million data, the proposed approach reached close performance using just 1200 data.

After that, the YOLOv7 network has been used for pedestrian detection tasks on the road. In this study, YOLOv7 network has been retrained with the Pascal VOC dataset to perform human detection tasks. Additionally, the trained network has been tested using the CamVid dataset.

Both road segmentation and pedestrian detection subsystems have been evaluated using various metrics. They also compared to literature studies that used the same database. The comparison tables have been reported. They produced promising results compared to the literature. The visual results of the system have also been reported.

In this study, it is focused on artificial intelligence and software to realize the system. This system can be integrated into any embedded system. With a simple camera fixed to the vehicle, the system can take images and transfer these images to a system and produce instant results. We are planning to expand this system for road anomaly detection in the near future.

References

[1] K.-W. Chiang and Y.-W. Huang, "An intelligent navigator for seamless INS/GPS integrated land vehicle navigation applications," *Appl Soft Comput*, vol. 8, no. 1, pp. 722–733, Jan. 2008, doi: 10.1016/j.asoc.2007.05.010.

[2] X. Zhang and M. M. Khan, "Intelligent Vehicle Navigation and Traffic System," in *Principles of Intelligent Automobiles*, Singapore: Springer Singapore, 2019, pp. 175–209. doi: 10.1007/978-981-13-2484-0_5.

[3] J. Jin and X. Ma, "A group-based traffic signal control with adaptive learning ability," *Eng Appl Artif Intell*, vol. 65, pp. 282–293, Oct. 2017, doi: 10.1016/j.engappai.2017.07.022.

[4] J.-Z. Yuan, H. Chen, B. Zhao, and Y. Xu, "Estimation of Vehicle Pose and Position with Monocular Camera at Urban Road Intersections," *J Comput Sci Technol*, vol. 32, no. 6, pp. 1150–1161, Nov. 2017, doi: 10.1007/s11390-017-1790-3.

[5] C. Ma, W. Hao, A. Wang, and H. Zhao, "Developing a Coordinated Signal Control System for Urban Ring Road Under the Vehicle-Infrastructure Connected Environment," *IEEE Access*, vol. 6, pp. 52471–52478, 2018, doi: 10.1109/ACCESS.2018.2869890.

[6] S. Zhang, R. Benenson, M. Omran, J. Hosang, and B. Schiele, "Towards Reaching Human Performance in Pedestrian Detection," *IEEE Trans Pattern Anal Mach Intell*, vol. 40, no. 4, pp. 973–986, Apr. 2018, doi: 10.1109/TPAMI.2017.2700460.

[7] J. Li, X. Liang, S. Shen, T. Xu, J. Feng, and S. Yan, "Scale-aware Fast R-CNN for Pedestrian Detection," *IEEE Trans Multimedia*, pp. 1–1, 2017, doi: 10.1109/TMM.2017.2759508.

[8] B. Ma, S. Lakshmanan, and A. O. Hero, "Simultaneous detection of lane and pavement boundaries using model-based multisensor fusion," *IEEE Transactions on Intelligent Transportation Systems*, vol. 1, no. 3, pp. 135–147, 2000, doi: 10.1109/6979.892150.

[9] J. Sparbert, K. Dietmayer, and D. Streller, "Lane detection and street type classification using laser range images," in *ITSC 2001. 2001 IEEE Intelligent Transportation Systems. Proceedings (Cat. No.01TH8585)*, pp. 454–459. doi: 10.1109/ITSC.2001.948700.

[10] M. Bertozzi and A. Broggi, "GOLD: a parallel real-time stereo vision system for generic obstacle and lane detection," *IEEE Transactions on Image Processing*, vol. 7, no. 1, pp. 62–81, 1998, doi: 10.1109/83.650851.

[11] Y. Wang, E. K. Teoh, and D. Shen, "Lane detection and tracking using B-Snake," *Image Vis Comput*, vol. 22, no. 4, pp. 269–280, Apr. 2004, doi: 10.1016/j.imavis.2003.10.003.

[12] Luo-Wei Tsai, Jun-Wei Hsieh, Chi-Hung Chuang, and Kuo-Chin Fan, "Lane detection using directional random walks," in *2008 IEEE Intelligent Vehicles Symposium*, Jun. 2008, pp. 303–306. doi: 10.1109/IVS.2008.4621271.

[13] Q. Li, N. Zheng, and H. Cheng, "Springrobot: A Prototype Autonomous Vehicle and Its Algorithms for Lane Detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 5, no. 4, pp. 300–308, Dec. 2004, doi: 10.1109/TITS.2004.838220.

[14] Y. Wang, E. K. Teoh, and D. Shen, "Lane detection and tracking using B-Snake," *Image Vis Comput*, vol. 22, no. 4, pp. 269–280, Apr. 2004, doi: 10.1016/j.imavis.2003.10.003.

[15] J. M. Alvarez, T. Gevers, Y. LeCun, and A. M. Lopez, "Road Scene Segmentation from a Single Image," 2012, pp. 376–389. doi: 10.1007/978-3-642-33786-4_28.

[16] G. L. Oliveira, W. Burgard, and T. Brox, "Efficient deep models for monocular road segmentation," in *2016 IEEE/RSJ*

International Conference on Intelligent Robots and Systems (IROS), Oct. 2016, pp. 4885–4891. doi: 10.1109/IROS.2016.7759717.

[17] H. Liu, X. Han, X. Li, Y. Yao, P. Huang, and Z. Tang, "Deep representation learning for road detection using Siamese network," *Multimed Tools Appl*, vol. 78, no. 17, pp. 24269–24283, Sep. 2019, doi: 10.1007/s11042-018-6986-1.

[18] C.-A. Brust, S. Sickert, M. Simon, E. Rodner, and J. Denzler, "Convolutional Patch Networks with Spatial Prior for Road Detection and Urban Scene Understanding," Feb. 2015.

[19] D. T. Nguyen, W. Li, and P. O. Ogunbona, "Human detection from images and videos: A survey," *Pattern Recognit*, vol. 51, pp. 148–175, Mar. 2016, doi: 10.1016/j.patcog.2015.08.027.

[20] Y. Kim and T. Moon, "Human Detection and Activity Classification Based on Micro-Doppler Signatures Using Deep Convolutional Neural Networks," *IEEE Geoscience and Remote Sensing Letters*, vol. 13, no. 1, pp. 8–12, Jan. 2016, doi: 10.1109/LGRS.2015.2491329.

[21] W. Ouyang and X. Wang, "Joint deep learning for pedestrian detection," *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2056–2063, 2013, doi: 10.1109/ICCV.2013.257.

[22] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT Press, 2016.

[23] S. Srinivas, R. K. Sarvadevabhatla, K. R. Mopuri, N. Prabhu, S. S. S. Kruthiventi, and R. V. Babu, "A Taxonomy of Deep Convolutional Neural Nets for Computer Vision," *Front Robot AI*, Jan. 2016, doi: 10.3389/frobt.2015.00036.

[24] V. Nair and G. E. Hinton, "Rectified Linear Units Improve Restricted Boltzmann Machines," in *Proceedings of the 27th International Conference on International Conference on Machine Learning*, 2010, pp. 807–814.

[25] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," *Journal of Machine Learning Research*, vol. 15, pp. 1929–1958, 2014, doi: 10.1214/12-AOS1000.

[26] R. Padilla, S. L. Netto, and E. A. B. da Silva, "A Survey on Performance Metrics for Object-Detection Algorithms," in *2020 International Conference on Systems, Signals and Image Processing (IWSSIP)*, Jul. 2020, pp. 237–242. doi: 10.1109/IWSSIP48289.2020.9145130.

[27] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 779–788. doi: 10.1109/CVPR.2016.91.

[28] Y. Tian, G. Yang, Z. Wang, H. Wang, E. Li, and Z. Liang, "Apple detection during different growth stages in orchards using the improved YOLO-V3 model," *Comput Electron Agric*, vol. 157, pp. 417–426, Feb. 2019, doi: 10.1016/j.compag.2019.01.012.

[29] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," Jul. 2022.

[30] 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, Dec. 2017, doi: 10.1109/TPAMI.2016.2644615.

[31] M. Everingham, L. Van~Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes Challenge 2012 (VOC2012) Results."

[32] B. , Dwyer, J. , Nelson, J. , Solawetz, and et. al., "Roboflow (Version 1.0)," *https://roboflow.com.*, 2022.

[33] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," Jul. 2022.

[34] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," in *International Conference on Learning Representations*, Sep. 2015.

[35] Ö. Dülger, H. Oğuztüzün, and M. Demirekler, "Memory Coalescing Implementation of Metropolis Resampling on Graphics Processing Unit," *J Signal Process Syst*, vol. 90, no. 3, pp. 433–447, Mar. 2018, doi: 10.1007/s11265-017-1254-6.

[36] X. Song, T. Rui, S. Zhang, J. Fei, and X. Wang, "The Road Segmentation Method Based on the Deep Auto-Encoder with Supervised Learning," *Computers & Electrical Engineering*, vol. 68, pp. 381–388, 2018, doi: 10.1007/978-3-319-69877-9_28.

[37] B. L. Priya, S. Jayalakshmy, G. Idayachandran, and S. Kumaran, "Performance Analysis of Semantic Segmentation using Optimized CNN based SegNet," in *2022 International Conference on Smart Technologies and Systems for Next*

Generation Computing (ICSTSN), Mar. 2022, pp. 1–5. doi: 10.1109/ICSTSN53084.2022.9761293.

[38] W. Nagai, T. Katayama, T. Song, and T. Shimamoto, "High Efficiency Dataset Generation for Semantic Video Segmentation on Road Intersection," in *2022 37th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC)*, Jul. 2022, pp. 1–4. doi: 10.1109/ITC-CSCC55581.2022.9894901.

[39] Yunsheng Jiang and Jinwen Ma, "Combination features and models for human detection," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, pp. 240–248. doi: 10.1109/CVPR.2015.7298620.

[40] H. Htet Lin, "Person detection based on fusion histogram of gradients with texture (FHGT) local features," *International Journal of Research in Computer Scientific (IJRCS)*, vol. 5, no. 4, 2018.

[41] H. Htet Lin, "Smart Feature Fusion and Model for Human Detection," *Review of Computer Engineering Research*, vol. 7, no. 1, pp. 38–46, 2020, doi: 10.18488/journal.76.2020.71.38.46.

[42] I. Oztel, "Human Detection System using Different Depths of the Resnet-50 in Faster R-CNN," in *2020 4th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, Oct. 2020, pp. 1–5. doi: 10.1109/ISMSIT50672.2020.9255109.

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 ™.