International Journal of Innovative
Computing, Information and Control
Volume **17**, Number **6**, December **2021**

ICIC International ©2021 ISSN 1349-4198
pp. **2201–2216**

ENHANCED VEHICLE CLASSIFICATION USING TRANSFER LEARNING AND A NOVEL DUPLICATION-BASED DATA AUGMENTATION TECHNIQUE

Abdelrahman Harras*, Akinori Tsuji, Stephen Karungaru and Kenji Terada

Graduate School of Advanced Technology and Science
Tokushima University
2-1, Minamijyosanjima-cho, Tokushima 770-8506, Japan
*Corresponding author: c501847007@tokushima-u.ac.jp
{ a-tsuji; terada }@is.tokushima-u.ac.jp; karungaru@tokushima-u.ac.jp

Received June 2021; revised September 2021

Abstract. *Vehicles classification has a significant role in transportation systems for monitoring traffic conditions. In such systems, there is a need not only to detect vehicles but also to categorize vehicle types in surveillance and security applications. We propose a robust vehicle classification method using data duplication and transfer learning in deep neural networks. In our method, after the pre-trained learning model has been re-trained and validated using a small data set, vehicles are classified into 6 categories: Crossover, Sedan, Hatchback, Pickup, Van, and Minivan. To enhance the classification accuracy, we adjust the training process by applying data augmentation by duplicating the training data. In the training process, each type of vehicles is used multiple times in every epoch to gain optimal learning results. In the proposed method, transfer learning is applied using ResNet-50 to obtaining a highly effective learning model. To evaluate the proposed method, a testing dataset, including 640 images of 6 vehicle types was used. In the experiments, we used a Stanford-based dataset, with 8,000 vehicle images, to retest the model to ensure its generalization ability. From the experimental results, our proposed method indicated that the classification performance has increased from 92.68% to 99.70%.*
Keywords: Vehicle classification, Data augmentation, Transfer learning, Deep neural networks

1. **Introduction.** Vehicle classification is one of the solutions for the problem of traffic overcrowding through which large amounts of data from roads are collected and analyzed for proper decision-making. In general, these tasks are previously performed by a human, and now they are changed to be fully automated in artificial intelligence (AI) applications. Deep neural networks (DNN), which is a form of AI application, are widely used in vehicle classification solutions. Vehicle type recognition is required as part of the automatic toll, where the system should be able to characterize which type of vehicles (car, van, truck, etc.) goes to the tollbooth to assess the price the driver should pay. It is also required in video surveillance applications used to monitor pollution peaks. In these applications, recognizing the vehicle's type allows for better regulation by setting up different routes according to their size, and accordingly, to their related pollution. Moreover, it is important to categorize vehicles according to their types in applications concerned with driving assistance, automatic parking, and autonomous driving. In this context, the estimation of the orientation and 3D localization of vehicles around the smart vehicle is required to predict its appropriate path and speed. Additionally, in security implementations, it is important

DOI: 10.24507/ijicic.17.06.2201

to monitor and classify vehicles around the clock in sensitive areas, like airports, cantonment areas, and secure locations. In such areas, it is only allowed for specific vehicles to park or move while other types are not. In these implementations, a pre-trained DNN is used to automatically capture classes of interest in the upcoming stream of vehicles and make decisions accordingly. However, the generalization of DNN based models depends on the amount of data they have trained on. More training data is necessary to make a more robust learning model. It is a real challenge to get enough labeled data to train DNN models because it is of high cost and time-consuming. Increasing data artificially is one of the most suitable solutions for this problem as, instead of trying to find and label more data, we build new training data based on what we have. In our research, we investigate how to effectively increase the training dataset artificially by creating new data from the existing dataset. The most well-known technique is data augmentation in which modified images are created from the available data or images. It is used in most DNN based solutions and with the most complex and powerful algorithms regardless of the data kind. In [1] for instance, plant leaf recognition is among those diversified DNN based implementations that use data augmentation techniques. The size of the plant leaf training dataset was increased by 25 times by using simple techniques like rotation, blur, contrast, scaling, illumination, and transformation. In this regard, the objective of our research is to develop a high-performance method to classify vehicles in an easy and fast way by training a pre-trained DNN on an effective dataset that is artificially enlarged by data augmentation. The proposed data augmentation technique is a new approach. The training dataset is augmented by duplicating its instances for getting highly effective training. In this work, transfer learning is used for changing the final layers of the pre-trained network. The final layers are removed and the networks are re-trained on a small amount of dataset including only vehicle classes of interest. Therefore, it learns only these classes' features without the features of other classes learned earlier. In addition, during the re-training process, the data augmentation technique is used to create image variations that improve the training performance and the ability of the trained model to be generalized. This technique is enhanced by duplicating vehicle image instances multiple times in the training dataset. Accordingly, the duplication assists the simple random augmentation such as cropping, flipping, and rotating in each epoch during the training. In the proposed method, the optimum duplication number, which generates the highest classification performance, was empirically identified.

This work contributes to enhancing vehicle classification in DNNs by implementing a novel technique for augmenting the training dataset by implementing multiple duplications to its instances. Thus, it transforms the data duplication phenomenon from being very negative to be a good one. This technique increases the training data efficiency and thus reduces significantly the performance gap between lab test results and real-world ones. Besides, in this work, a good example of using a limited number of objects to train a DNN model effectively, through using transfer learning, was demonstrated. The rest of the paper is structured as follows. Section 2 provides a literature review for the current research on vehicle classification with DNN. Section 3 shows the description and procedures of the proposed method. Section 4 presents the experimental results, comparison of existing methods, discussions, and contributions of the work. Section 5 presents the conclusion and the future work.

2. **Literature Review.** Vehicle classification has been tackled from different perspectives summarized in Table 1. In [2], the transfer learning-based vehicle classification system, by using a convolutional neural network (CNN) pre-trained on a large-scale dataset was proposed. The system is divided into two stages. The vehicle area is detected by Haar-

TABLE 1. Summary of vehicle classification research

Reference	Technology	Technique	Dataset
Transfer Learning-Based Vehicle Classification [2]	Haar algorithm, GoogLeNet, Transfer learning	Vehicle area is detected by Haar-like features, and then GoogLeNet is used for vehicle classification.	Limited scale vehicle dataset
Indonesia Toll Road Vehicle Classification Using Transfer Learning with Pre-Trained ResNet Models [3]	ResNet, Transfer learning	The images are classified manually based on number of axles used to distinguish between vehicles.	Dataset collected by a smart phone camera
Vehicle Type Classification Using a Semi-Supervised Convolutional Neural Network [4]	Semi-supervised CNN	Sparse Laplacian filter is used to capture vehicles. A trained softmax layer is used to get the probability of each vehicle type.	BIT-Vehicle dataset
Convolutional Neural Network Based Vehicle Classification in Adverse Illuminous Conditions for Intelligent Transportation Systems [5]	ResNet, Transfer learning	Improving ResNet is used by adding a new classification block.	Custom dataset including 10,000 images of six vehicle classes
Vehicle Classification for Large-Scale Traffic Surveillance Videos Using Convolutional Neural Networks [6]	GoogLeNet, Transfer learning	Fine tune the CNN model.	Custom vehicle dataset including 13,700 images of six classes
ResNet-Based Vehicle Classification and Localization in Traffic Surveillance Systems [7]	ResNet 18	A technique called joint fine-tuning (JF) and a dropping CNN (DropCNN) method as a synergy to the JF.	MIO-TCD traffic dataset
An Ensemble Deep Learning Method for Vehicle Type Classification on Visual Traffic Surveillance Sensors [8]	CNN, Data augmentation	Apply data augmentation with balanced sampling and use the augmented training dataset to construct a different architecture with new parameters.	MIO-TCD traffic dataset
Improving Deep Ensemble Vehicle Classification by Using Selected Adversarial Samples [9]	Generative Adversarial Nets (GAN), data augmentation	Integrate DNNs with data augmentation in 3 successive phases.	MIO-TCD traffic dataset

like features on the roadway video, and then GoogLeNet is used as a transfer learning-based model for vehicle classification. They reached higher accuracy than the conventional method without transfer learning. Authors in [3] proposed a solution to conduct vehicle classification based on type and number of axles and categorized them into five groups. ResNet CNN and transfer learning have been used to fine-tune the model to enhance classification accuracy. A real-time vehicle classification solution was proposed in [4], through which they used a sparse Laplacian filter to capture vehicles' information and a trained softmax layer to get the probability of each vehicle type. They claimed that their method works well in complex scenes. It was tested on the BIT-Vehicle dataset and achieved a day

accuracy of 96.1% and a night accuracy of 89.6%. A real-time CNN-based vehicle classification system was proposed in [5]. The six-vehicle classes were categorized in adverse illuminating conditions. A performance comparison, in terms of accuracy and convergence, was conducted between 6 pre-trained CNNs which are fine-tuned and tested. It was concluded with ResNet which was further improved by adding a new classification block to it for better performance. Their method achieved 99.68%, 99.65%, and 99.56% accuracy, precision, and F1-score respectively on a custom dataset. In [6], the proposed system through which vehicle classification is conducted in two phases. GoogLeNet is used in the first pre-training phase to get the initial model with its appropriate connection weights. Then the model is fine-tuned in the 2nd phase, on a 13,700 dataset of six categories captured from real highway surveillance videos. The average classification accuracy is up to 98.26%. In [7], a ResNet-based vehicle classification and localization method was presented. The classification performance was improved using a technique called joint fine-tuning (JF) and a dropping CNN (DropCNN) method as a synergy to the JF. In performing the localization task, they used a region-based detector combined with a feature extraction network constructed from ResNet50 and 101 layers. They claimed that they achieved the highest accuracy among several state-of-the-art methods. From another viewpoint, where the objective was to confront the issue of imbalanced data collected from traffic surveillance in vehicle classification methods, a method was proposed in [8] for classifying imbalanced data by integrating CNN with balanced sampling. The method has two phases. In the first phase, the unbalanced dataset problem is mitigated by applying data augmentation with balanced sampling. In the second, they used the augmented training dataset to construct an ensemble of CNN's different architecture models with new parameters. They claimed that their proposed method enhances the overall accuracy of all categories as compared with the baseline algorithms. The authors in [9] proposed a method based on Generative Adversarial Nets (GAN) through which they integrated DNNs with data augmentation. In the first, they generated adversarial samples for the rare classes by using the original dataset to train several GANs. In the second, they filtered out the low-quality adversarial samples from an ensemble of different CNN models after being trained on the original imbalanced dataset. Finally, they refined the model by compiling the selected adversarial samples with the augmented dataset. Their method increases the classification performance of some categories while maintaining a high overall accuracy compared with the existing methods.

In the previous literature review, the deep learning-based vehicle classification models are diversified. They use different methods to achieve significantly higher accuracy with different state-of-the-art technologies. In this regard, the reviewed models in [2,3,5-7] have used a combination of transfer learning and deep learning while others are deep learning-based ones. In [5,8,9], they are using data augmentation techniques to artificially modify or increase the training data to improve the model performance. In [2-4,6], they used a custom dataset for training which could be a source of performance degradation because of its limited size as compared with the real-world data. In these models, we expect that there is still a noticeable performance gap between lab test results and real-world results caused by a lack of efficient training data. For instance, their lab accuracies are 98.30%, 99%, 95.70%, and 98.26% respectively, and real-world results will be lower, such as in [5] where the lab accuracy is 99.68% and the real world one is 97.66% when tested on the VeRi dataset with an accuracy gap of 2.02%. Therefore, from this analysis, we consider that there is a need for a data-driven technique that is capable of minimizing the performance gap between the lab test and the real-world results to the lowest possible level. We provide an optimum vehicle classification solution that is independent of the adopted CNN. It is based on using an effective novel augmentation technique that outperforms the current

data augmentation. The technique is capable of raising the efficiency of the data in hand to make the model perform better and achieving higher accuracy when tested on real-world data. Besides, we integrated the proposed novel augmentation technique with transfer learning technology to maximize the acquired efficiency.

3. **Proposed Method.** We present the proposed method that is composed of 3 main operations: transfer learning, a recurring cycle of training-assessing, and testing. In this presentation, transfer learning, data augmentation, and data duplication technologies are described. Training-assessing cycle is explained where the preparation of the dataset is described and both basic and staged training cycles are highlighted and justified. Generalization through which we implement a novel technique of augmentation with duplication is described and justified. The flowchart of the proposed method is shown in Figure 1.

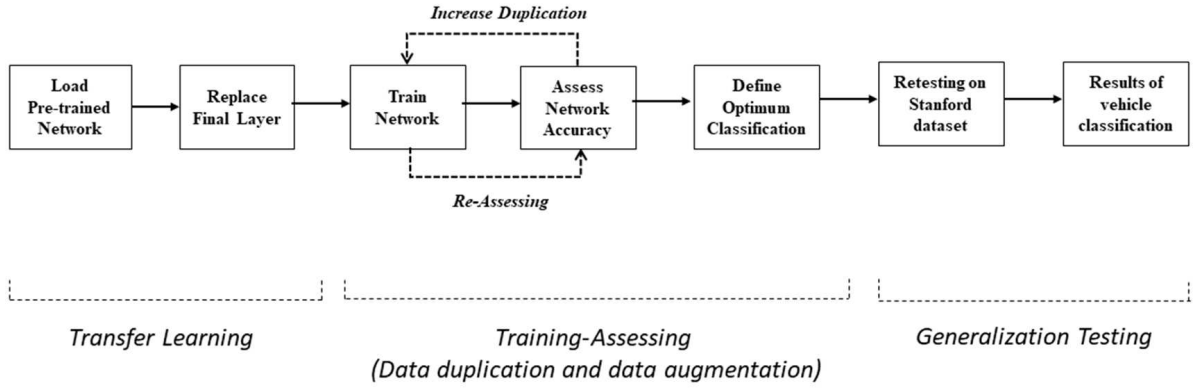

FIGURE 1. Flowchart of the proposed method

3.1. **Transfer learning.** There are a large number of variations of deep learning architectures. CNN is the most efficient method. The first part of a CNN is the actual convolutional part. It works as an image feature extractor. Successive filters, or convolution nuclei, create new images called convolutional maps. The output of this part is then forwarded into a second part, made up of fully connected layers. When building neural networks with more layers, the network tends to learn more functionality with better generalization. However, there is a limitation as the model deepens. Neural network training becomes difficult and the model's performance begins to saturate and even degrade. This is largely caused by the known problem of the leakage gradient. When the gradient is back-propagated to the previous layers, repeated multiplications of derivatives make the gradient infinitely small making it impossible to update the weights of the previous layers. Residual learning is one of the advances in solving this problem. Instead of making the network learn the characteristics of the input images, it learns the residue in residual learning. In this context, ResNet and similar networks appeared due to this problem of backpropagation of the gradients and the increase in learning error in CNNs. ResNet gets its name from its residual blocks that allow the information to jump over some layers and thus allow the model to be deeper. ResNet does this by introducing a so-called "shortcut connection" that skips one or more middle layers. In other words, the output of a first layer is provided to a deeper layer without any transformation. The idea behind the jump connection is that it is easier to learn the residue than any other direct mapping. It pushes the deeper layers to learn something different from what the entry has already learned. In the meantime, it allows very deep networks to be formed without worrying about the degradation problem.

In the proposed method, ResNet-50 is used in the context of transfer learning as a baseline pre-trained classification network. Transfer learning makes it possible to use a pre-trained model on a new problem. In this regard, a pre-trained model is re-trained on a small dataset consisting of hundreds of labeled vehicle images belonging to different classes and benefits from what allowed it to reach its optimal configuration. This re-trained model is then capable of making predictions on new data and defining the estimated accuracy. Figure 2 shows the transfer learning workflow where late layers of the pre-trained network ResNet-50 are removed to make the model specialized only on the new training dataset on which the model has been re-trained.

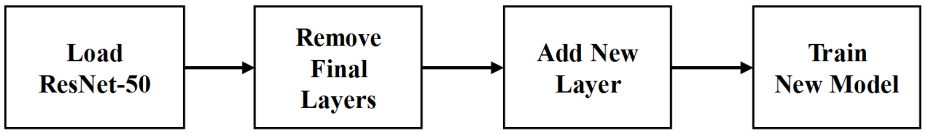

FIGURE 2. Flowchart of the transfer learning process in ResNet-50

In the implementation of transfer learning, the pre-trained network, ResNet-50, which is pre-trained by 1 million images of 1,000 classes, is loaded. Then, its fully connected layer and classification layer are replaced with new ones adapted to the new dataset. These replaced layers contain the required information needed to combine the extracted features into class probabilities that consist of a loss value and predicted labels. The new fully connected layers have several outputs equal to 6 classes. Figure 3 shows the configuration of ResNet-50 in transfer learning. The network performs the initial convolution and max-pooling using 7×7 followed by 3×3 kernels in the first two layers. There are 4 consecutive convolution blocks with 3, 4, 6, 3 layers stacked one over the other, respectively. For example, the configuration of the first layer in the first block is 1×1, and 64 means that the convolution operation is performed by a kernel size 64 with a 1×1 convolution filter.

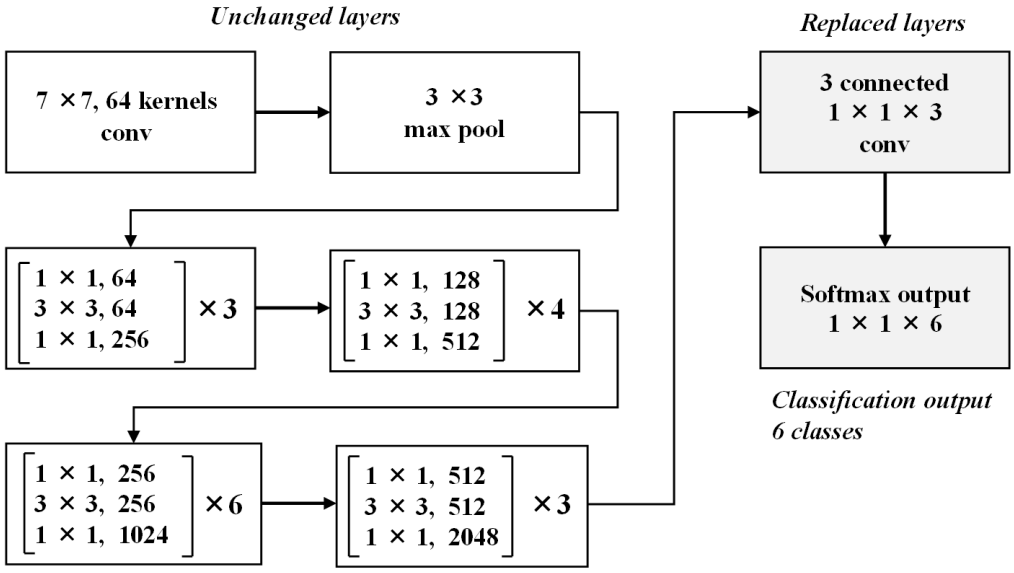

FIGURE 3. Architecture of ResNet-50 in transfer learning for obtaining 6 classes output

3.2. **Training-assessing.** We describe the dataset structure and how it was prepared. The recurring cycle of training-assessing is presented and both augmentation and duplication techniques are described.

3.2.1. *Prepare dataset.* The custom dataset is composed of six classes from the Stanford cars dataset [11]. Figure 4 shows an example of vehicle images from the dataset. The dataset includes 2,128 car images, grouped into six classes manually: Crossover, Sedan, Hatchback, Van, Pickup, and Minivan. In the procedure, the dataset is divided into the training set 70% and the test set 30% as shown in Figure 5. The validation set is formed by a random selection of 10% in the training dataset. The training dataset is used to learn the model vehicle types. The test dataset is used to verify that the model is capable of classifying the dataset. Table 2 shows the structure of the dataset after being classified into six categories. Each class has, approximately, the same number of images as the following, Crossover 362 images, Sedan 351 images, Hatchback 339 images, Van 352 images, Pickup 368 images, Minivan 356 images, respectively. The validation set is formed by a random selection of 10% of the images in the training dataset and contains (20) images in each class. In the testing set, each class has approximately the same number of images as the following, Crossover 108 images, Sedan 106 images, Hatchback 102 images, Van 106 images, Pickup 111 images, Minivan 107 images, respectively, with a total number of 640 images. The training dataset instances are of different sizes, while the network requires that input image size is $224 \times 224 \times 3$. Therefore, the input images are resized to this resolution. In addition, two data augmentation techniques are performed on those images before input to the training stages. In the first, training set instances are duplicated in a staged procedure. Then, images are randomly flipped along the vertical axis and translated horizontally and vertically up to 30 pixels in a random way.

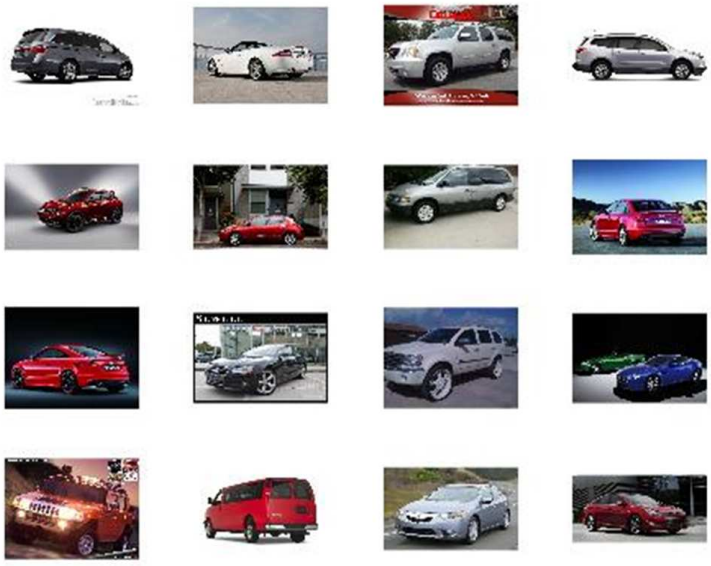

FIGURE 4. Example of vehicle images from the training dataset

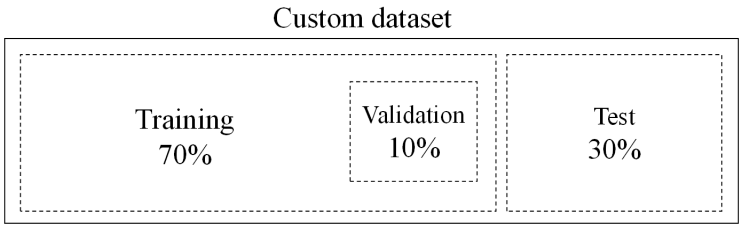

FIGURE 5. Structure of the custom dataset, training set 70% including validation set 10% and test set 30%

TABLE 2. The dataset after being classified into six classes

Class / Dataset	Crossover	Sedan	Hatchback	Van	Pickup	Minivan	Subtotal
Training	254	245	237	246	257	249	1,488
Validation	(20)	(20)	(20)	(20)	(20)	(20)	(20)
Testing	108	106	102	106	111	107	640
Total	**362**	**351**	**339**	**352**	**368**	**356**	**2,128**

3.2.2. *Recurring cycle of training assessing.* The model training is performed in a staged manner. In the first stage, the training dataset has only the baseline vehicle instances in the dataset without any duplication. In every subsequent stage, the number of duplications per image is increased by one and the training is repeated. The model is evaluated afterward on the test dataset where the overall accuracy is recorded. This cycle of multiple activities (i.e., duplication, training, and assessing) is repeated several times, with the number of data duplications increased by 1 each time, until the maximum classification accuracy is reached.

3.2.3. *Data augmentation.* In the proposed method, two consecutive augmentation techniques are implemented to increase the size of the training dataset as shown in Figure 6. The training dataset is duplicated first, and then the basic augmentation techniques are implemented where the training images are randomly flipped along the vertical axis and randomly translated up to 30 pixels horizontally and vertically. Therefore, the size of the dataset is artificially increased in the learning process as the data augmentation created more image variants in the training dataset. The created image variations improve the ability of the model undergoing training to be generalized and highly effective.

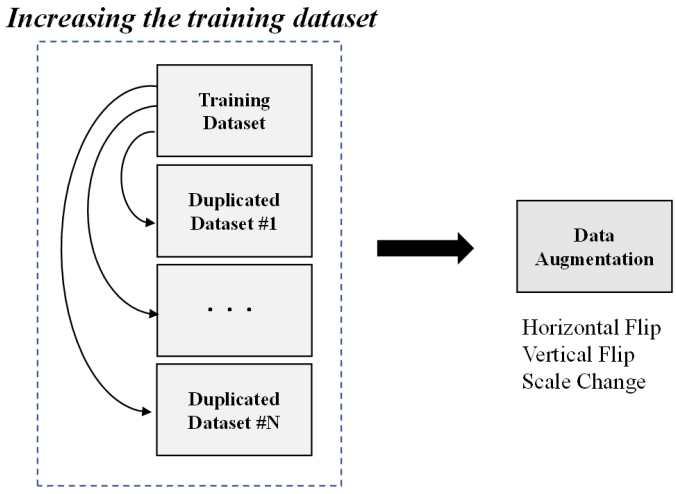

FIGURE 6. The proposed data augmentation process using data duplication for increasing the size of training dataset

We prove empirically that augmenting data by duplication enhances classifier performance. The training process is conducted in several stages. At each stage, the number of data duplications is increased, and also the model is re-assessed as shown in Figure 1. The loop of increasing data duplications and re-assessing is continued until the optimum performance is reached. The learning process continues in the context of transfer learning until the model reaches the desired stop criterion which is the maximum number of data duplications in the training dataset.

3.2.4. *Data duplication.* Data duplication in a dataset is generally avoided in deep learning because when dividing a dataset into testing and training sets, the goal then is to ensure that no data is shared between the two sets. This is because the purpose of the testing process is to simulate real-world data. Thus, when there are duplicates in the original dataset, the model is tested on data that has already been seen in the training dataset and results in unrealistically high accuracy.

The proposed method used data duplication as one of the data augmentation techniques as shown in Figure 6. The data duplication is performed only in the training dataset so that the testing dataset remains to have unique samples and thus represent the real-world data. Accordingly, the data duplication increases the size of the training dataset artificially and thus the learning process is improved since there are more image variants on which the DNN model is trained. The training is performed in consecutive stages. In each stage, the model is trained on the duplicated training dataset and evaluated on a testing dataset that has no duplicates. In this context, the baseline instances, as well as the duplicated ones, are exposed to the applied simple augmentation techniques which are flipping the training images randomly along the vertical axis and translate them up to 30 pixels horizontally and vertically in a random way. Thus, during the training process, each instance in the dataset appears with different variations as many times as its duplicate counts in each epoch. By the proposed data augmentation, it is ensured that the model will not be over-fitted due to the limited training data as, according to [10], the limited training data might make the model unable to generalize, i.e., it is a good model when classification is based on the features included in the small training dataset; however, it will not be able to do good classification for other objects with features that it has not been trained on.

3.3. **Generalization testing.** In DL, if the model can adapt to the training data, it does not mean that it will perform well on the test data that represents the real world. This disparity between performance on training and real-world data is called the generalization gap.

In the proposed method, to generalize and confirm the achieved results, generalization testing is implemented when the model reaches the desired stop criterion which is the maximum number of data duplication in the training dataset which leads to the highest accuracy level. The ResNet-50 based proposed model was then tested on the generalization test set which is a Stanford-based-dataset consisting of 8,000 images grouped into 6 categories. The objective is to prove that the network not only behaves properly with the testing dataset but also behaves similarly with a real-world dataset.

4. **Experiments.** We describe the testing environment, the model training, and the staged training. We also present the training parameters and indicate every type of training. From the experimental results, the optimum classification is identified. Furthermore, to ensure the model generalization, we tested the model on the Stanford image dataset. From this dataset, we collected, visually, 8,000 vehicle images and grouped them into six categories manually.

4.1. **Environment.** The proposed method was implemented on the MATLAB environment and verified using a GPU-enabled PC with the following specification: NVIDIA GeForce RTX 2080 Ti, Processor Intel Core i7-9700 CPU@300 GHz, 3000 MHz, 8 Cores, and 8 logical processors.

4.2. **Model training.** We performed a staged training where we started with a baseline training in which the basic data augmentation was implemented, and followed it with consecutive training stages. In the staged training, the proposed data augmentation technique was performed in which the training dataset instances are duplicated in the subsequent stages as 0, 2, 3, 4, and 5. The classification performance in each stage was recorded and the training progress in each one was also indicated.

4.2.1. *Baseline training.* Training parameters were set as the following; the number of epochs 6, learning rate 0.0001, and batch size 10. Figure 7 shows the progress of both training and validation of the baseline model training. It is noted that the over-fitting was not seen during the training progress. In the baseline training, we implemented the general data augmentation on the training dataset before commencing the training and tested the model afterward on the testing dataset. Figure 8 shows the confusion matrix of the testing dataset classification results after the baseline training was completed.

4.2.2. *Staged training.* The training parameters were unchanged while increasing the number of duplications one by one in every subsequent stage. When the training was

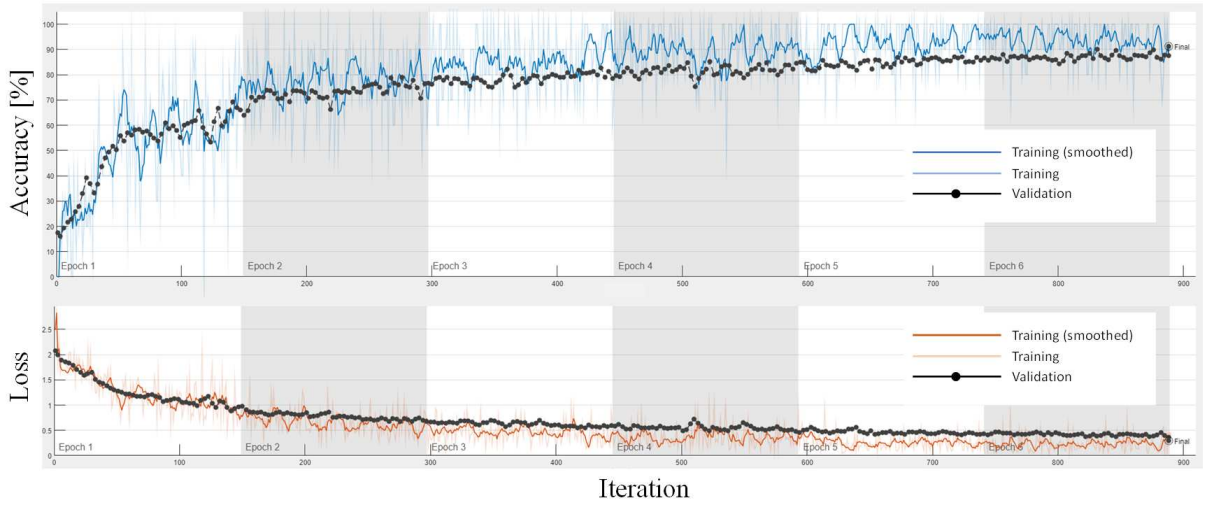

FIGURE 7. The progress of the baseline training at the stage 0 in the proposed method with 6 epochs, learning rate 0.0001, batch size 10

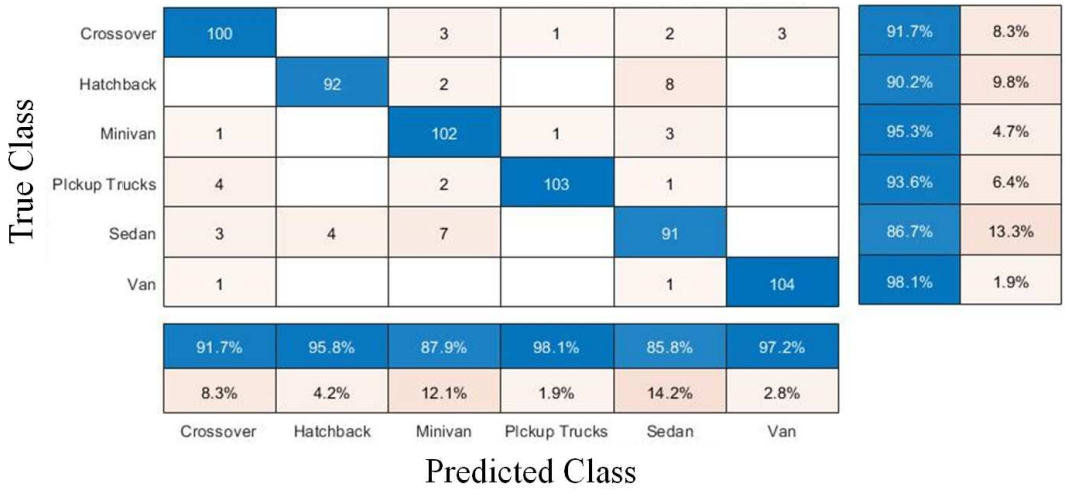

FIGURE 8. The confusion matrix of validation dataset in the baseline stage 0

finished, the model has evaluated afterward on the testing dataset and the overall classification accuracy was recorded. Table 3 shows the configuration of the training dataset used in the staged training. Stage numbers are 0, 2, 3, 4, and 5, which means how many times the training dataset instances are duplicated. Stage 0 is used for the original training dataset, stage 2 is a 2 times duplication of the original dataset, stage 3 is a 3 times duplication, and so on. We continued in this staged training until the optimum classification was obtained. Figure 9 shows the confusion matrix of the testing dataset classification results in stage 4, where the accuracy reached the maximum value.

TABLE 3. Configuration of training dataset with data duplication in each stage

Class / Stages	Crossover	Sedan	Hatchback	Van	Pickup	Minivan	**Total**
0 (original)	254	245	237	246	257	249	**1,488**
2	508	490	474	492	514	498	**2,976**
3	762	735	711	738	717	747	**4,437**
4	1,016	980	948	984	1,028	996	**5,952**
5	1,270	1,225	1,185	1,230	1,285	1,245	**7,440**

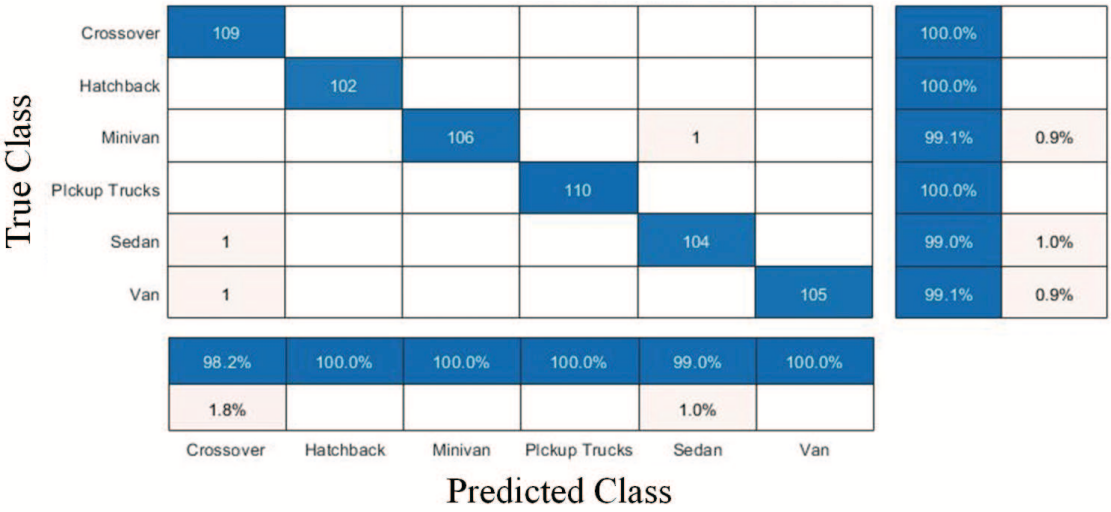

FIGURE 9. The confusion matrix of the validation dataset in stage 4 with maximum classification performance

4.3. **Experimental results.** In the experiments, we evaluated the ResNet-50 based model in several consecutive stages. The experimental results are shown in Table 4. Figure 10 summarizes the overall accuracy for all 5 stages. Overall accuracy in stage 0 is 92.68%

TABLE 4. Classification performance in 5 stages of duplication

Class / Stages	Crossover	Hatchback	Minivan	Pickup	Sedan	Van	**Average accuracy [%]**
0	91.3	95.8	87.9	98.1	85.8	97.2	92.68
2	97.3	98.1	98.1	100	99.0	100	98.75
3	97.3	98.1	98.2	100	100	100	98.93
4	**98.2**	**100**	**100**	**100**	**100**	**100**	**99.70**
5	98.1	99.8	99.7	99.9	100	100	99.58

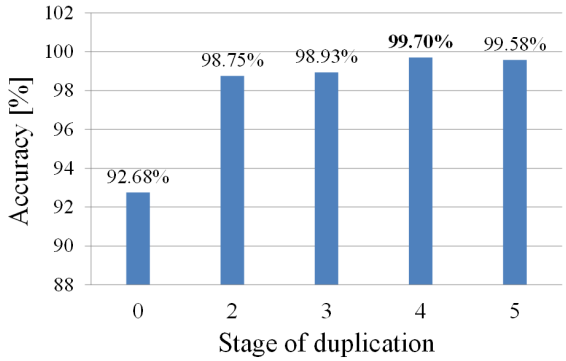

FIGURE 10. Comparison of classification accuracy in 5 stages. Stage 0 is the baseline data augmentation without any duplication, 2, 3, 4, and 5 are the applied proposed data duplication.

FIGURE 11. Experimental results of vehicle classification with the proposed method applied in different stages

without any duplication. The maximum accuracy in all stages is 99.70% and achieved in stage 4. It is noted that the accuracy increases by increasing the number of duplicates and then starts to decrease when the number of duplicates is more than stage 4 because the model then starts overfitting. Figure 11 shows the experimental results of vehicle classification when applying the proposed method. It shows samples of Minivan, Van, Sedan, Crossover, and Pickup vehicles that are successfully classified.

4.4. **Generalization.** The ResNet-50 based proposed model was tested on Stanford based dataset consisting of 8,000 images grouped into 6 categories. The objective is to prove that the network not only behaves properly with the testing dataset but also behaves similarly with a real-world dataset. We collected, visually, 8,000 vehicle images for the 6 categories of interest from the Stanford dataset and grouped them manually into those categories. We retested the model, whose training dataset instances were duplicated 4 times, on this constructed Stanford-based dataset and recorded the classification accuracy of each of the six categories and the overall accuracy which is the average of the six accuracies. The average was calculated by dividing the summation of the accuracies by 6. Table 5 shows that the overall average accuracy of the model is 99.5% when tested on the Stanford-based dataset, which is very close to the lab result 99.7%.

We used the Stanford dataset because it contains 16,185 images that are divided into 196 different categories, and each category has a visually distinctive view. Every vehicle image in every category constitutes a vehicle in the foreground as well as background and

TABLE 5. Experimental results of accuracy applied the proposed method with ResNet-50 on Stanford 8,000 image dataset

Class / Network	Crossover	Hatchback	Minivan	Pickup	Sedan	Van	**Average accuracy [%]**
ResNet-50	100	99.0	98.8	99.8	99.2	100	**99.5**

different angles' views. Besides, they are different from each other in terms of proximity or distance from the camera, the intensity of illumination, etc. Also, some of the images are of high quality, indicating that they were taken professionally, while others are of relatively low quality, indicating that they were collected from the Internet. Thus, it represents the real world in terms of images' characteristics we get from the street-mounted cameras.

4.5. **Discussion.** In general, vehicle classification needs large volumes of labeled images for training the DNN model. Training the model with very few images dataset is challenging. It means that it has access to only a limited number of data which is a cause of overfitting. The overlearning on the training data makes the model incapable of making relevant predictions on new images. In this case, the classification effectiveness would be poor when verifying the performance of the model on a testing dataset. This problem is often solved by artificially increasing the size of the training dataset through data augmentation techniques.

In the proposed method, we enhanced the classification performance by performing a novel augmentation technique, based on data duplication, before implementing the known basic data augmentation. The training dataset instances were duplicated in each stage. Therefore, the size of the dataset is artificially increased and the learning process is improved since the proposed data duplication technique obtains more image variations in the training dataset. This improves the ability of the model undergoing training to use what it has learned in predicting new images. In the experimental results, the accuracy continues to be enhanced as the number of duplicates increases until the saturation state is reached, i.e., duplication continues to result in more new image variations in the training dataset. The reason is that, in that case, each duplicated instance is transformed into many new variants different from the original ones. Then, when increasing duplicates to more than four, the produced variants have become redundant of old ones and there are no new image variants anymore. The model then, with duplicates more than 4, exhibits overfitting as it thus becomes adapted to the redundant variants specifically. In the custom dataset, vehicle categories are balanced in terms of the number of images in each category as shown in Figure 12; otherwise, the classification process will be biased or over-fitted towards categories of a high number of instances. This is a common problem in machine learning related to sample balancing. ResNet-50 was chosen as a baseline network because it is less demanding in terms of computation resources. Besides, according to [12], if our target is to reach the most accuracy larger than 75%, ResNet-50 achieves the maximum throughput, (i.e., the number of inferences per second) when compared with other DNNs. To ensure the generalization of the proposed method, the model was re-tested on a Stanford-based dataset of 8,000 images grouped into the six categories where we get, nearly, the same classification performance with ResNet-50 of 99.5% as shown in Table 5. Other vehicle classes would be classified as "others" according to the evidence reversal concept, where the inexistence of any vehicle in any of the 6 classes is used as a clue of its existence as one of the unauthorized classes. The performance of the proposed method is compared with some state-of-the-art vehicle classification methods as shown in Table 6. The comparison shows that the proposed method outperforms them. In the comparison

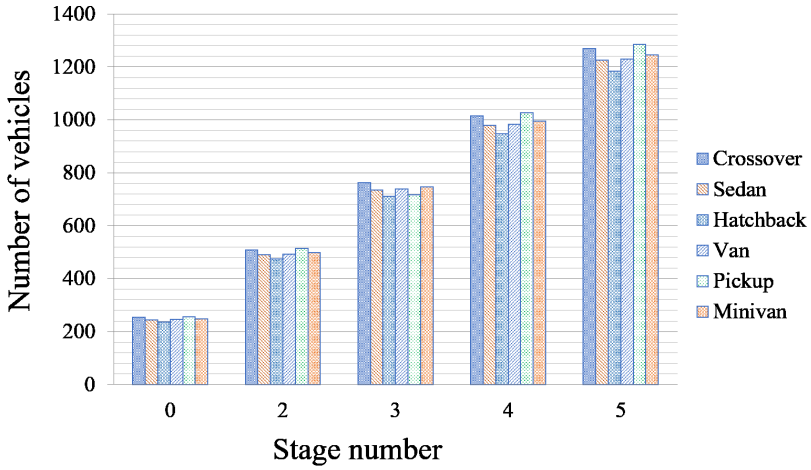

FIGURE 12. Balanced sampling in the 5 stages

TABLE 6. Comparison of accuracy of existing methods

Method	Dataset	Lab accuracy	Real-world accuracy
Jo et al. [2]	Limited scale vehicle dataset	98.30%	–
Sasongko and Fanany [3]	Dataset collected by a smartphone camera	99.00%	–
Dong et al. [4]	Custom dataset	95.70%	–
Butt et al. [5]	VeRi dataset	99.68%	97.66%
Zhuo et al. [6]	Custom vehicle dataset including 13,700 images of six classes	98.26%	–
Jung et al. [7]	MIO-TCD traffic dataset	–	97.90%
Our work	**8,000 images from Stanford dataset**	**99.70%**	**99.50%**

table, accuracies of methods verified on custom datasets are considered lab accuracy. In [5], the achieved lab accuracy is 99.68% which is very close to our results. However, when the model was tested on VeRi dataset, the accuracy was 97.66%, with an accuracy gap of 2.02%. In our method, lab accuracy is 99.7% and real-world accuracy is 99.5%. The gap is only 0.2%. Besides, in [5], the model is based on ResNet-152, which is much deeper, i.e., has a higher computational cost, and with higher inference time when compared with ResNet-50. This makes our model more proper for real-time implementations. Additionally, network training in [5] was performed on a machine equipped with the RTX 2080TI, 11 GB DDR5 GPU, Core i9 - 9900k CPU with 32 GBRAM. It took 8 hours to complete the training. In our model, the training was implemented on a GPU-enabled PC with the following specification: NVIDIA GeForce RTX 2080 Ti, Processor Intel Core i7-9700 CPU@300 GHz, 3000 MHz, 8 Cores, and 8 logical processors. It took 4 hours to complete the training.

Our model outperforms [7] as well in the real-world test accuracy. However, the model in [7] is based on ResNet-18 which is more proper in real-time implementations as it is less deep and with less inference time, when compared with ResNet-50 [12]. It is worth mentioning that the comparison does not include [8,9] as they used different benchmarks. They used precision and recall, rather than accuracy, to assess their method because of the imbalanced nature of the dataset. In [8], balanced sampling is used to augment, and thus alleviate the unbalanced data. They trained an ensemble CNNs based architecture

on this augmented training dataset and got an enhanced mean precision of all categories in case of high overall accuracy. In [9] they used a new image synthesis technique based on GANs. Through this technique, rare classes of training data are augmented with synthetic images.

4.6. **Contribution.** There are many contributions in this work as the following. 1) An empirical proof was provided to show that vehicle classification is enhanced in DNN by implementing a technique of augmenting the training dataset by duplicating the data instances. 2) The proposed method makes use of the duplicated data, which is usually avoided or reduced when conducting training of DNN based computer vision systems. In our method, the training data is only intentionally duplicated many times to enhance the data augmentation and thus transformed this phenomenon from being very negative to be a good technique in similar deep learning systems. 3) The evidence reversal phenomenon was highlighted throughout this work. According to this phenomenon, the inexistence of any object in a certain category is used as a clue of its existence in the opposite one. Specifically, six specific classes of interest are distinguished and classified. Thus, by definition, all vehicle classes, other than the 6 classes of interest, are considered as classes of "no interest", i.e., unknown. In practical implementations, where we are focusing on monitoring traffic rules in a certain area, these classes of no interest are temporarily considered violating rules until proven otherwise. That is to say, when you classify 6 specific classes, the unclassified classes are considered unknown and needed to be checked for a potential rules violation. 4) A good example of using transfer learning, through which a DNN model, focuses only on a limited number of objects was demonstrated. In this example, a modified ResNet-50 network was trained and tested on a small custom dataset and re-tested further on 8,000 images to ensure the capability of the generalization. 5) A comparison between the proposed vehicle classification method and the existing ones was conducted to highlight the effectiveness of the proposed method.

5. **Conclusion.** A new approach for enhancing vehicle classification was proposed through which we enhance training data augmentation by duplicating the training dataset instances. We proved, empirically, that this technique of augmenting training data by duplication enhances the classifier performance by a considerable value. We tested the model on a custom dataset of 640 images, grouped into 6 categories, and proved that the overall classification accuracy has improved from 92.68% without any duplications in the training dataset, to the maximum accuracy of 99.70% with 4 duplicates for every instance in the training dataset. To assure the method generalization, we re-tested this optimum model using the Stanford-based custom dataset of 8,000 images grouped into 6 categories and indicated that the model classification performance is nearly the same 99.5% with ResNet-50. We compared the proposed method with the existing methods and it was shown that it outperforms them. In future work, it will be intended to use the proposed method to enhance the fine-grained classification of vehicles.

REFERENCES

[1] T. Chompookham and O. Surinta, Ensemble methods with deep convolutional neural networks for plant leaf recognition, *ICIC Express Letters*, vol.15, no.6, pp.553-565, doi: 10.24507/icicel.15.06.553, 2021.

[2] S. Y. Jo, N. Ahn, Y. Lee and S. Kang, Transfer learning-based vehicle classification, *2018 International SoC Design Conference (ISOCC)*, Daegu, Korea, pp.127-128, doi: 10.1109/ISOCC.2018.8649802, 2018.

[3] A. T. Sasongko and M. I. Fanany, Indonesia toll road vehicle classification using transfer learning with pre-trained ResNet models, *2019 International Seminar on Research of Information Technology and Intelligent Systems (ISRITI)*, pp.373-378, doi: 10.1109/ISRITI48646.2019.9034590, 2019.

[4] Z. Dong, Y. Wu, M. Pei and Y. Jia, Vehicle type classification using a semi-supervised convolutional neural network, *IEEE Transactions on Intelligent Transportation Systems*, vol.16, no.4, pp.2247-2256, doi: 10.1109/TITS.2015.2402438, 2015.
[5] M. A. Butt, A. M. Khattak, S. Shafique, B. Hayat, S. Abid, K. Kim, M. Waqas, A. A. Sajid and A. Adnan, Convolutional neural network based vehicle classification in adverse illuminous conditions for intelligent transportation systems, *Complexity*, vol.2021, pp.1-11, https://doi.org/10.1155/2021/6644861, 2021.
[6] L. Zhuo, L. Jiang, Z. Zhu et al., Vehicle classification for large-scale traffic surveillance videos using convolutional neural networks, *Machine Vision and Applications*, vol.28, pp.793-802, https://doi.org/10.1007/s00138-017-0846-2, 2017.
[7] H. Jung, M. K. Choi, J. Jung, J. H. Lee, S. Kwon and W. Y. Jung, ResNet-based vehicle classification and localization in traffic surveillance systems, *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp.61-67, 2017.
[8] W. Liu, M. Zhang, Z. Luo and Y. Cai, An ensemble deep learning method for vehicle type classification on visual traffic surveillance sensors, *IEEE Access*, vol.5, pp.24417-24425, doi: 10.1109/ACCESS.2017.2766203, 2017.
[9] W. Liu, Z. Luo and S. Li, Improving deep ensemble vehicle classification by using selected adversarial samples, *Knowledge-Based Systems*, vol.160, pp.167-175, 2018.
[10] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever and R. Salakhutdinov, A simple way to prevent neural networks from overfitting, *J. Mach. Learn. Res.*, vol.15, pp.1929-1958, 2014.
[11] J. Krause, M. Stark, J. Deng and F. Li, 3D object representations for fine-grained categorization, *2013 IEEE International Conference on Computer Vision Workshops*, 2013.
[12] S. Bianco, R. Cadene, L. Celona and P. Napoletano, Benchmark analysis of representative deep neural network architectures, *IEEE Access*, vol.6, pp.64270-64277, 2018.

Author Biography

Abdelrahman Harras received his BBA degree in Management Information Systems from Arab Academy for Science, Technology and Maritime Transport in Egypt and a Master's degree in System Innovation Engineering from Tokushima University in Japan in 2018. His current research interests include computer vision and deep learning.

Akinori Tsuji received his B.E. degree from the University of Tottori in 1998. He received his M.E. and Ph.D. degrees from the University of Tokushima in 2002 and 2013, respectively. He has been an engineer in the Department of Information Science and Intelligent Systems at Tokushima University since 1998. His research interests include image processing and embedded systems. He is a member of the IEICE and IEEE.

Stephen Karungaru received his B.S. degree from the Department of Electrical/Electronics, MOI University. He received a master's and a doctoral degree from the Department of Information Science and Intelligent systems, Faculty of Engineering, Tokushima University, in 2001 and 2004. He became an Associate Professor at, Institute of Advanced Science and Technology, Tokushima University in 2004. His research interests are in face detection, recognition, neural networks, and genetic algorithms.

Kenji Terada received a doctoral degree from Keio University in 1995. In 2009, he became a Professor in the Department of Information Science and Intelligent Systems, University of Tokushima department. His research interests are in computer vision and image processing. He is a member of the IEICE, SICE, SCIE, and JSPE.