International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 16 (2018) pp. 12976-12982

Scratch Detection in Cars Using a Convolutional Neural Network by Means of Transfer Learning

César Giovany Pachón-Suescún[1], Paula C. Useche Murillo[2], Robinson Jimenez-Moreno[3].

1WDQW5HVHDUFKHUDUFKHSDUWPHQWRIHFKDWURQLFV(QJLQHHULQJ0LOLWDU1XHYD*UDQDGD8QLYHUVLWBRJRWiRORPELD

3URIHVVRUHSDUWPHQWRIHFKDWURQLFV(QJLQHHULQJ0LOLWDU1XHYD*UDQDGD8QLYHUVLWBRJRWiRORPELD

Abstract

This paper presents the development of a system of recognition of scratches in cars, implemented through convolutional neural networks (CNN). For this case the AlexNet architecture is employed making use of the transfer learning technique. As input images, sections of the vehicles are entered for training and, as an exit from CNN, the last Fully-Connected is altered so that it only has two exit categories (sections with scratches and without scratches). After the training, a validation accuracy of 88.29% is obtained and with the test images an average be-tween the categories of 86.99% is reached, presenting disadvantages in the classification because of the analysis of areas that did not correspond to the cars or even dirt in parts of the car that can be confused with scratches.

Keywords. Convolutional Neural Network, Transfer Learning, Scratch Detection, AlexNet.

INTRODUCTION

The development of artificial intelligence techniques for the classification and recognition of patterns has come to address broad fields of application as they improve the results obtained with these methods, allowing the development of applications such as the recognition of faces for access systems, as the one presented in [1]. Other fields of application range from complex food systems, where these techniques are used to analyze foods and ensure their quality [2], to leak detection tests in aseptic package seams, using image processing and classification techniques, from images collected experimentally, obtaining results between 96.92 - 98.46% accuracy in the classification, as presented in [3].

Some of the techniques of artificial intelligence that allow the recognition of pat-terns and classification of images are the Convolutional Neural Netowrks (CNN). These have been designed to classify a large number of categories, using filter pack-ages whose parameters are trained from large databases of thousands of images for each category such as Alexnet [4]. However, depending on the categories to be trained and their similarity, CNN may present problems for classifying them, as in the case of medical images, given that having databases with a large number of images of this type in some cases is not possible. For this reason, there are alternative methods of training such as training a CNN from another already trained with a different database, as mentioned in [5], a process known as transfer learning.

A clear example of transfer learning is the use of CNN developed by [6], whose small filters and large number of convolution layers have allowed the recognition of a large number of categories that can be generalized to different data sets with cutting-edge results. On the other hand, the use of the transfer learning technique for the development of multitasking applications has been exploited, in such a way that they allow an autonomous agent to learn to behave in a certain way in front of a group of tasks and to have the ability to generalize what it has learned for different spaces or work situations, testing the methods proposed in different Atari games [7].

CNNs have been worked on in different fields of application, there are even studies focused on explaining their operation applying deconvolution techniques as mentioned in [8]. In [9], an application of transfer learning is presented focused on the detection of pedestrians in the street and in [10], deep CNNs are used for the recognition of faces from multiple views.

Additionally, works focused on the classification of vehicles has been developed, such as the one proposed by [11], where images of the front of the mobile agent are captured and a semi-supervised CNN is used, trained with few labels and Laplacian filters to set the kernel of the network, in order to obtain the probability of what type of vehicle belongs to in the image entered. In [12], vehicles that have been captured by low resolution traffic cameras are detected and classified, reaching an accuracy of 94.7%.

In the present work, the development of a system of detection of scratches in vehicles is exposed, where transfer learning was used for the training of the CNN employed, and divisions of the global image for exhaustive searches of the scratch. It started with the Alexnet network architecture for the training of a new CNN focused on the recognition of scratches, where a global image of a vehicle is captured and divided into sections to search in each one the existence of some type of scratch, and once they are identified, the sections of the vehicle where they are located are high-lighted.

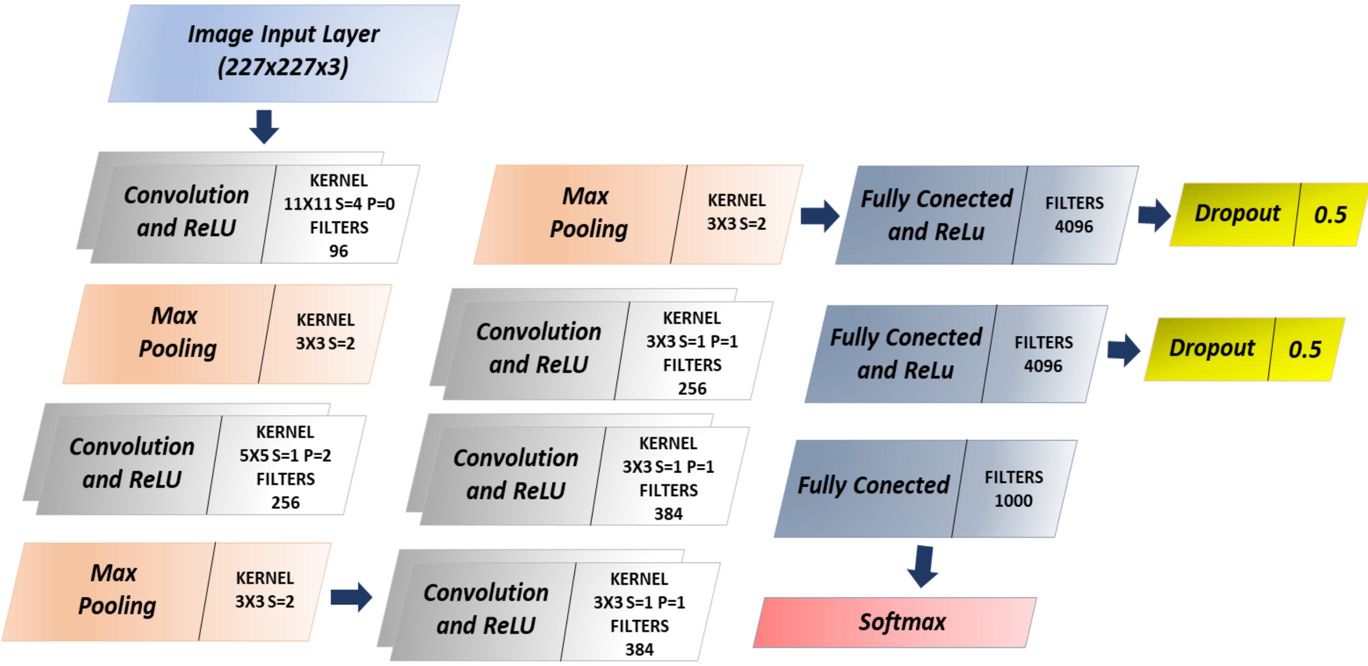

Figure 1. AlexNet architecture.

The paper is divided into three sections: The first one corresponds to the materials and methods used, where the architecture implemented by transfer learning, the database used, and the set training parameters are described. The second section focus-es on the results and discussions of network training, the generated confusion matrix, cases in which a correct and incorrect prediction is presented, and processing times. The last section is focused on exposing the conclusions derived from the work.

MATERIALS AND METHODS

When designing systems based on convolutional neural networks, three things are necessary: network architecture, database, and training parameters. Next, each of these parts is described.

Architecture by transfer learning

The AlexNet architecture is implemented, which consists of 5 layers convolutions, 3 of MaxPooling and 3 of Fully-Connected. It should be noted that for the normalization of the output data of the network, this CNN implements a Softmax type function (see Figure 1).

AlexNet architecture receives as input parameters images of 227x227 pixels of 3 channels (RGB), these images go through each of the layers to finally be classified in one of the 1000 categories for which the network was trained. In the case of this work, there are only 2 categories, sections with scratches and sections without scratches, for this reason, it is necessary to change the number of classes in the fully-connected output to 2 and retrain the network. In Figure 2 the modified section of the AlexNet architecture is shown.

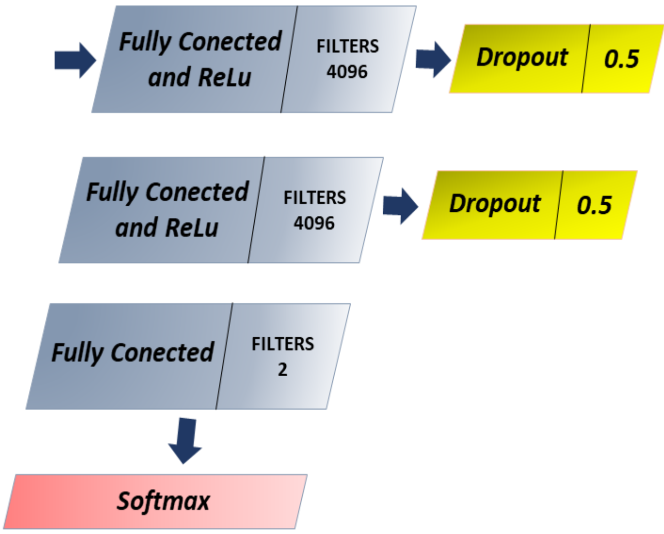

Figure 2. Last Fully-Connected modified for only 2 output categories.

Data Base

Taking into account the architecture of AlexNet, one of the initial problems that arise when generating a database of scratches in vehicles is to define the resolution with which the captures of the cars must be taken, since when resizing the image to the input size of 227x227 pixels, the details of the scratches disappear (see Figure 3).

Figure 3. Resized car of 2340x4160 pixels at 227x227 pixels

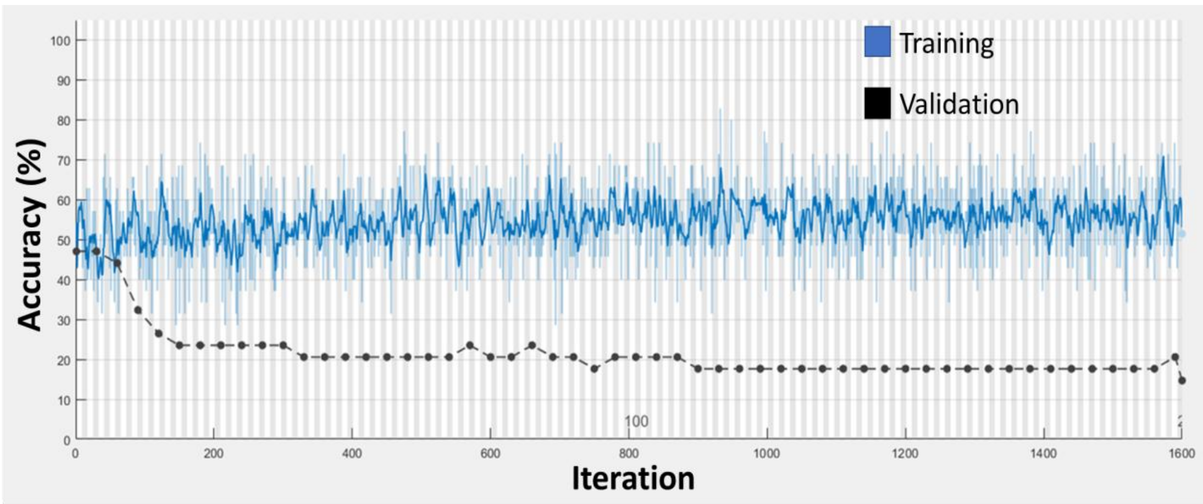

Figure 4. CNN training with resized vehicles.

Based on the above, if CNN is trained with a database of this type, it is not possible for the network to learn the characteristics of the scratch and perform a correct classification between cars that have scratches or not, as evidenced in Figure 4, where the graph of network training is observed. In the X axis are the training epochs and iterations, and in Y axis, the accuracy of the network when classifying each car in the correct category.

The previous tests were performed with databases at different resolutions, but the results were similar, concluding that taking the entire vehicle regardless of the resolution of the same, resizing it will lose the details of the scratch. Therefore, the strategy of dividing the images of the database into multiple sections and generating a data-base with sections with and without scratches is proposed (see Figure 5).

Figure 5. Database sections with scratches and without scratches.

With the sections of the vehicle images, a database of 862 sections without scratches and 424 with scratches is generated. In order to generalize the network, an augmentation in the database is perform by making changes in the lighting, by means of the tool developed in [13], obtaining in this way a total of 2736 sections without scratch-es and 1272 with scratches. Of each category of the final database, 80% of the images are for training, 10% for tests and 10% for validations during training, the latter being necessary to show that the network is not in overfitting.

Training parameters

To retrain the AlexNet network and adapt it to the current problem, it is necessary to establish learning parameters of the network. The batch size is set to 40, in this way, from the total of the training dataset, random packets of 40 images are made to update the weights of the network, having a total of 80 packages per epoch, thus avoiding overloading the GPU. 200 training epochs are set, this value is selected based on iterative tests, being sufficient in order to avoid overfitting of the network (see Table 1).

Table 1. Training parameters.

Batch size	40
Training Epochs	200
Learning Rate	0.00001

RESULTS AND DISCUSSIN

Training results

Once the database, the CNN network and the training parameters are set, it can be proceed to train the network, showing the behavior presented in Figure 6. In the upper graph the accuracy is presented in the Y axis and in the X each of the iterations, the blue lines represent the accuracy that each batch had in the training in each iteration and the black dots, the accuracy of the validation images. It is evident that the training images tend to have values of precision in their classification higher than 90%, while the images of validation in the last epoch reached an accuracy of 88.28%. The lower graph shows the losses in each training set and in the X axis each of the iterations; the orange lines represent the losses in each iteration of the training images and the black points correspond to the losses of the validation images, being 0.2233 in their last check.

Based on the foregoing, it can be shown that the behavior of the validation images tends to move away from the checks of the training images. This can be caused by the fact that, between more training epochs, the CNN tends to memorize the images and loses the generalization of the characteristics learned from a section of a car with or without scratch.

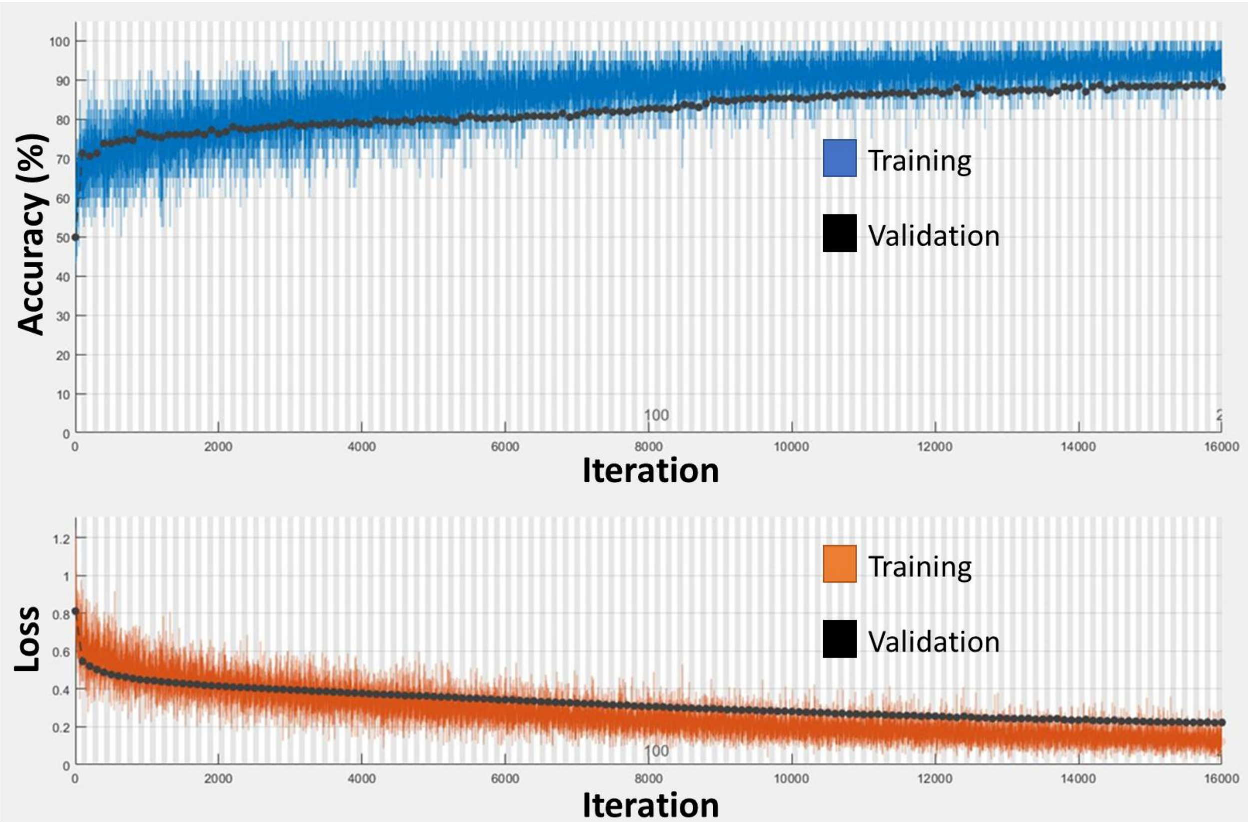

Figure 1. Training using transfer learning.

Confusion matrix

In order to check the accuracy on the trained CNN, test images corresponding to 127 sections with scratches and 273 without them are classified. Figure 7 shows the confusion matrix, where on its diagonal are correctly classified images and outside it, the images that were confused with another category.

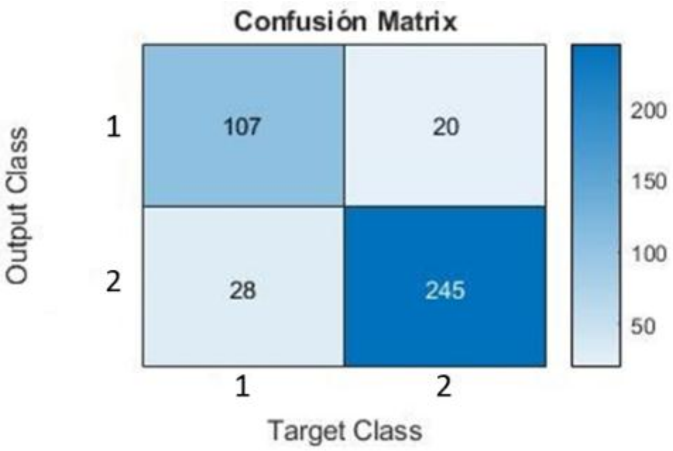

Figure 7. Confusion matrix of the test dataset, where 1 = Section with scratch and 2 = Section with-out scratch.

From the confusion matrix, it can be shown that of the 127 images of sections with scratches, 20 were classified erroneously by the network obtaining an 84.25% accuracy in the classification of images with scratches. On the other hand, from 273 sections without scratches, 28 were classified in another category, obtaining 89.74% accuracy in the classification of sections without scratches. Therefore, the average accuracy of CNN with the test images is 86.99%.

Right and wrong classifications

In order to establish a point of improvement for the system in future developments, it is necessary to identify the cases in which the CNN correctly and incorrectly classifies the sections of the vehicles. In Figure 8-A, sections with scratches are observed, which for visualization reasons were marked with a red arrow, and sections without scratch-es that were classified in the correct category. In Figure 8-B, there are some cases in which the CNN classified the sections in the incorrect category, where it is identified that in the case of the sections with scratches, it is possible that these were confused with part of the car or because the lighting. In Figure 8-B, a section corresponding to the asphalt is observed, although it belongs to the category without scratches, the CNN classified it in the category with scratch, just like the blue vehicle section, which presents dirt that is mistaken for scratches.

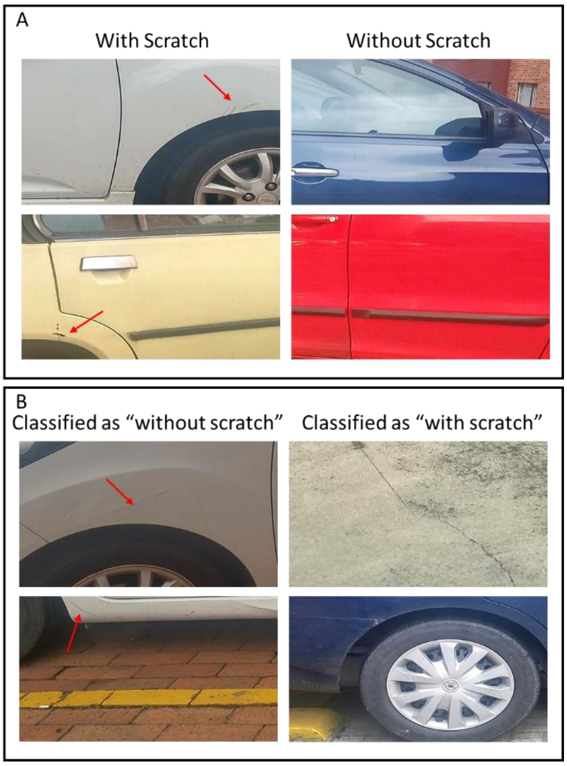

Figure 8. Correct and incorrect classifications, where A = correctly classified cases and B = cases with incorrect classification.

In **Error! Reference source not found.**-A, one of the test vehicles is shown, in which a scratch is located (marked with the red arrow), the car is sectioned and each of these sections is evaluated by CNN, in case it detects a scratch in the final assembly, the affected section is highlighted as shown in **Error! Reference source not found.**-B.

Figure 9. Process of identifying sections with scratches.

Processing times

In the case of implementation in an application in real time, it is important to know the average times it could take the system to analyze a vehicle to identify if it has any scratch or not. **Error! Reference source not found.** shows the time it takes CNN if a section without scratch, one with scratch or all the sections that make up the car are entered.

Table 2. Processing times.

	Section without scratch	Section with scratch	25 sections of the car
Time (s)	0.0866	0.0914	0.5706

CONCLUSIONS

The detection of scratches in cars tends to be a complex task if adequate strategies for analyzing vehicles are not looked for. By dividing the cars into multiple sections and training the CNN with these, it allows the network to learn the characteristics of the scratches, reaching an accuracy of up to 86.99% with the test images.

The implementation of a CNN architecture with transfer learning reduces the complexity in the development of quality systems focused on the detection of physical damage in vehicles, but AlexNet, when designed for the classification of 1000 categories, may not have the most optimal result for the detection of scratches. For example, in its first convolutional layer it has an 11x11 filter, and the scratches that occur in many of the vehicles are small. It is possible that with architectures with less complexity and variation in the size of the convolution filters, similar results can be obtained.

One of the problems that arise when analyzing the sections of the cars, are the analysis of sections that do not correspond to the vehicle, these increase the processing time and can cause erroneous classifications. A possible solution is the implementation of an R-CNN to extract from the image only the region where the car is located and later on to move on to the process of detecting scratches in the sections of the car with CNN.

Based on the processing times, it can be seen that the system has relatively small times, averaging 0.5706 seconds. In this way, it is possible to implement them in quality inspection systems in automotive paint and it can even be think of their implementation in parking lots to guarantee the integrity of the vehicles.

ACKNOWLEDGMENTS.

The research for this paper was supported by Davinci research Group of Nueva Granada Military University.

REFERENCES

[1] Useche, M.P.C., Arenas, J.O.P. and Moreno, R.J., 2018, "Face Recognition Access Control System using Convolutional Neural Networks," Research Journal of Applied Sciences, 13(1), pp.47-53.

[2] Jeon, I.J., 1991, "Pattern recognition techniques for food research and quality assurance," Instrumental Methods for Quality Assurance in Foods, pp.271-298.

[3] Adem, K., Orhan, U. and Hekim, M., 2015. Image processing based quality control of the impermeable seams in multilayered aseptic packages. Expert Systems with Applications, 42(7), pp.3785-3789. doi: 10.1016/j.eswa.2014.12.045.

[4] Krizhevsky, A., Sutskever, I. and Hinton, G.E., 2012, "Imagenet classification with deep convolutional neural networks," In Advances in neural information processing systems, pp. 1097-1105.

[5] Hoo-Chang, S., Roth, H.R., Gao, M., Lu, L., Xu, Z., Nogues, I., Yao, J., Mollura, D. and Summers, R.M., 2016, "Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning," IEEE transactions on medical imaging, 35(5), p.1285. doi: 10.1109/TMI.2016.2528162.

[6] Simonyan, K. and Zisserman, A., 2014, "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv:1409.1556.

[7] Parisotto, E., Ba, J.L. and Salakhutdinov, R., 2015. Actor-mimic: Deep multitask and transfer reinforcement learning. arXiv preprint arXiv:1511.06342.

[8] Zeiler, M.D. and Fergus, R., 2014, "Visualizing and understanding convolutional networks," In European conference on computer vision, Springer, Cham, pp. 818-833. doi: 10.1007/978-3-319-10590-1_53.

[9] Tian, Y., Luo, P., Wang, X. and Tang, X., 2015, "Deep learning strong parts for pedestrian detection," In Proceedings of the IEEE international conference on computer vision, pp. 1904-1912. doi: 10.1109/ICCV.2015.221.

[10] Farfade, S.S., Saberian, M.J. and Li, L.J., 2015, "Multi-view face detection using deep convolutional neural networks," In Proceedings of the 5th ACM on International Conference on Multimedia Retrieval, pp. 643-650.

[11] Dong, Z., Wu, Y., Pei, M. and Jia, Y., 2015, "Vehicle type classification using a semisupervised convolutional neural network," IEEE transactions on intelligent transportation systems, 16(4), pp.2247-2256. doi: 10.1109/TITS.2015.2402438.

[12] Bautista, C.M., Dy, C.A., Mañalac, M.I., Orbe, R.A. and Cordel, M., 2016, "Convolutional neural network for vehicle detection in low resolution traffic videos," In Region 10 Symposium (TENSYMP), 2016 IEEE, pp. 277-281. doi: 10.1109/TENCONSpring.2016.7519418.

[13] Useche, M.P.C., Arenas, J.P. and Moreno, R.J., 2017, “Implementation of a data augmentation algorithm validated by means of the accuracy of a convolutional neural network,” Journal of Engineering and Applied Sciences, pp.5323-5331. doi: 10.3923/jeasci.2017.5323.5331.