International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Traitement du Signal
Vol. 40, No. 2, April, 2023, pp. 619-627
Journal homepage: http://iieta.org/journals/ts

Master-Slave Convolutional Deep Architecture for Vehicle Identification and Type Classification

Bencheriet Chemesse Ennehar[1*], Bencheriet Samra[2]

[1] LAIG Laboratory, Department of Computer Sciences, 8 Mai 1945- Guelma University, PB 401, Guelma 24000, Algeria
[2] Faculty of Letters and Languages, University of Mentouri Brothers (Constantine 1), Constantine 25000, Algeria

Corresponding Author Email: bencheriet.chemesseennehar@univ-guelma.dz

https://doi.org/10.18280/ts.400220

Received: 20 October 2022
Accepted: 12 March 2023

Keywords:
intelligent transportation system, vehicle detection, vehicle classification, convolutional neural network, deep learning, master-slave architecture

ABSTRACT

Video surveillance of road traffic plays an important role in highway safety and is an important application of intelligent transportation systems. One of the basic applications of intelligent transport systems is the detection and classification of vehicle types. The major problems encountered by these systems are the significant similarity between the vehicles, frequent occlusions on the highway, and low resolution of the surveillance cameras. This paper proposes a novel convolutional neural network architecture called master-slave convolutional deep architecture for vehicle detection and type classification. The basic concept of this architecture is twofold: a. The sequential operation of the two networks where the slave network only works if the master network detects vehicles in the road scene will allow a considerable reduction in the search area for vehicles. It will induce a significant reduction in processing time. b. A combination of deep-shallow neural networks allows the system to share the knowledge gained from the vehicles on two networks. The first (master) shallow learns the shape of vehicles while the second (slave) is responsible for learning all the details of vehicles to distinguish the different classes. The experimental results, performed on 3200 images, have shown that the favorable performance of the proposed CNN architecture allowed us to achieve successful detection with TPR of 92% and TNR of 95% and vehicle type classification with a considerable mean average precision of 93.38% where cars classification gives the highest rate (98.63%).

1. INTRODUCTION

Road and motorway safety has become a problem of high priority for public authorities because of the massive increase in road accidents per year, mainly caused by the violation of the traffic code, driving under the influence of drugs, fatigue, drowsiness, speeding, congestion, etc. An effective; solution to this problem adopted in several developed countries; is installing an intelligent transport system (ITS) on the road network [1].

Intelligent Transportation Systems (ITS) owe their establishment and development to new technologies applied in the analysis and control of transport to improve safety, mobility, and efficiency [2].

ITS cover a wide range of applications enabling the processing and sharing of information to minimize congestion, improve traffic management, reduce pollution. from vehicles and increase the reliability and efficiency of public transport. The intelligent transportation system (ITS) significantly affects transport applications such as electronic toll collection, ramp counters, traffic light cameras, traffic light coordination, transit signal priority, and traffic systems passenger information.

The adoption of ITS is expected to increase in applications such as fleet monitoring, toll management, ticket management, transport pricing, telematics, and traffic monitoring. The main beneficiaries of the improvements in ITS safety are travelers, businesses, and transport agencies. ITS data also has homeland security applications [3, 4].

The research proposed for improving the efficiency of such systems is divided into two parts. The first concerns deep research on the improvement of sensor technologies representing the source of information for ITS [5-7], and the second mainly concerns the scientific community that works on improving intelligence algorithms and artificial vision, keys elements of ITS decision-making [8-11].

IoT research [12-14], which is a combination of technology and algorithms, has also participated in important solutions in conjunction with network research and its applications such as Blockchain, 5G network, … etc. [15-17].

Deep neural networks (deep learning) have been implemented in various fields and have given remarkable results, thus opening the door for researchers to expose them to unresolved problems [18-20]. Problems of which ITS are part, given the constraints and specifications of these systems mentioned above. Therefore, recently, deep neural networks have been exploited in ITS; to improve their performance, which before that was not very favorable; especially when it comes to real-time response for the monitoring and prevention of road accidents [21-23] or in recognition of the license plate or logo of vehicles inviolate of the highway code [24, 25]. Another application where deep learning has proven its effectiveness both in the context of road safety and energy through the control of traffic lights [26, 27].

Most vehicle-type classification systems proposed in the scientific literature are based on searching for each frame, forcing the system to search for vehicle classes even in scenes that do not contain any. This induces a considerable loss in research time, thus slowing down these systems, which are generally designed to operate in real time. To remedy the problem of searching for vehicle types in every frame, we have proposed an architecture based on two networks combined as master-slave. The search for vehicle types by the slave network only takes place if the master detects the presence of the vehicles on the scene. this will significantly improve the reliability and the performance of these systems.

The contributions of this paper are summarized as follows:

- We propose a system of identification and classification of vehicle types based on deep learning.
- We propose an architecture of a sequential CNN based on two deep learning networks: the first has the role of detection (identification) of vehicles among all moving objects in the sequence. In contrast, the second network is responsible for classifying the types of vehicles detected by the previous network.
- We propose a master-slave CNN where the vehicle detector is the master because his positive decision (detection of vehicles) will start the operation of the classifier CNN considered a slave.
- We propose a combination of shallow and deep networks where the choice of network depth depends on the context characteristics of the object sought in the scene, this is why the identification of the vehicles, which is an operation that requires only coarse details of the general appearance of vehicles to be able to distinguish them from other types of objects in the scene, we have chosen a shallow network, and for the classification of vehicle types, which is an operation that requires learning lesser details about each vehicle, we have adopted a deep network.

The remainder of this paper is organized as follows:

Section II presents related works about intelligent transportation systems and the most recent studies of vehicle detection and classification.

Section III presents the dataset used in the training and test of our system. Section IV presents the overall architecture of the proposed vehicle identification and type classification system. Experiment results are shown in Section V. The final section provides conclusions and directions for future research.

2. RELATED WORK

The detection and classification of vehicles remain one of the ITS problems not yet resolved because of several problems: including mainly the acquisition conditions related to sensors (cameras, electromagnetic loop, radar, optical fiber…, etc.) and the variable external environment in atmospheric conditions and lighting [28]

Researchers in the field of automatic road traffic monitoring have conducted preliminary research on detecting and classifying vehicles from a video stream issued by cameras placed on red highway lights or onboard vehicles in traffic [29]. Most recent research has adopted deep networks, thus allowing systems to learn from the external environment, which is often complex and variable according to climatic conditions. For example, Zohra et al. [30] proposed a framework for the detection and recognition of vehicles from a video stream. The proposed model used a deep learning approach based on the convolutional neural network (CNN). This model works in two stages: a data preparation stage, which applies to process on the images composing the data set to extract the characteristics. The second stage used the concept of convolutional neural networks to classify vehicles.

The dataset used contains two files (vehicle files and non-vehicle files). It is taken from video sequences (obtained by a front camera mounted on a car). To ensure proper learning of the data, images are captured under different road conditions (far, near, left, right). The vehicle file includes 8,798 images, and the non-vehicle file includes 8,971 images. Each image is of dimension: (64x64) pixels despite obtaining high accuracy. Tsai et al. [31] provided an optimized method of vehicle detection and classification based on deep learning technology for intelligent transport applications. The authors optimized the CNN (Convolutional Neural Network) architecture by refining the existing CNN architecture for intelligent transport applications. The proposed design; achieved 90% accuracy on three categories of target vehicles, including small vehicles (sedan, SUV, van), large vehicles (bus), and trucks, and achieved the performance of 720x480 video in different weather conditions (day, night, rain) at 25 fps. The vehicle datasets used to form the proposed models are IVS-1 and IVS-2 (collected by them), consisting respectively of 316,733 and 599,277 vehicles.

The combination of deep networks with other methods has been considered by researchers in order to set up more efficient vehicle detection and classification systems. For example, Chen et al. [32] presented a system for detecting and classifying vehicles using traffic surveillance cameras. First, the scales and proportions of the vehicles are grouped into the vehicle datasets using the k-means algorithm. Then a convolutional neural network is used to detect a vehicle then high-level, and low-level features are concatenated using feature merge techniques. Detected vehicles are therefore classified into four categories (bus, minibus, car, truck). To improve speed, a fully convolutional architecture is adopted instead of fully connected "FC" layers. The performance of the algorithm is evaluated on the "JiangSuHighway Dataset" (JSHD), composed of 5000 images collected from 25 videos of the JiangSu highway. The proposed algorithm achieved significant improvement over Faster R-CNN and SSD. The network speed is 15 FPS, three times faster than the Faster R-CNN. Seo [33] proposed a method of detection and classification of vehicles according to AUSTROADS's plan using Deep Learning and UAV (Unmanned Aerial Vehicle) in 4K UHD (ultra-high definition). Darknet-53 and Kalman filter are used to detect and classify vehicle types in UHD images. The aerial images used in this article are images of vehicles on the road, recorded at an altitude of fewer than 120 m using a drone (Phantom3 Professional by DJI). The size of the experimental images is 3940 × 2160, 30 FPS in UHD resolution. Vehicles are classified into three categories according to their size: short vehicles (motorcycles, cars), medium vehicles (buses, heavy trucks), and long vehicles (long trailers). Authors also suggested the variable classification method based on parked and stopped cars for traffic flow monitoring. He, therefore, considered the three conditions of driving, stopping, and parking. The results of the experiment show that the proposed approach gives low errors than conventional methods, which use a fixed search area. Asvadi et al. [34] discussed the problem of vehicle detection using Deep Convolutional Neural Network (ConvNet) and

3D-LIDAR (Object Motion Detector) data. They introduced: a vehicle detection system based on HG-HV (the hypothesis generation and verification) paradigm using a Deep ConvNet and range data from a 3D-LIDAR mounted on board an instrumented vehicle. The proposed solution is based on removing the points on the ground, followed by segmentation of the point clouds. Then bounding boxes are fitted to the segmented objects as vehicle hypotheses (step HG). Finally, the bounding boxes are used as inputs in a ConvNet to classify/verify the assumptions belonging to the category "vehicle" (step HV). The performance of the proposed system is evaluated on the KITTI Benchmark [35], where vehicle recognition accuracy with applying data augmentation is 96.02% for the training data set and 91.93% for the validation data set.

Although the existing studies have led to fruitful and encouraging results, the existing systems, on the one hand, do not meet the specific needs of poor road traffic, and, on the other hand, the majority of the proposed systems are based on a single search and classification module based on a single deep network embedded on sophisticated hardware, thus compensating for the shortcomings of these systems. Our study is based on the proposal of a new architecture using two CNN networks combined as master-slave whose main objective is to adapt to our road system (which is more or less mediocre) and having considerable performance thanks to the separation of detectors and vehicle classifiers, even if the equipment used is not high performance.

3. DATASET

The training and test of the proposed system are performed using a publicly available dataset called MIO-TCD dataset. Vehicle classification [36] with 50,000 images where vehicles are presented in different lighting conditions, different resolutions, and different viewing angles.

This dataset contains 11 categories described in Table 1.

Figure 1 contains some sample images of the MIO-TCD dataset.

Figure 1. Sample images of the MIO-TCD dataset

Table 1. Categories of MIO-TCD dataset

Categories	Number
Car	6238
Bus	5072
Bicycle	2254
motorcycle	1952
pedestrian	6232
Pickup truck	6260
Non motorize vehicle	1721
Work van	5425
Articulated truck	4961
Non-motorized vehicle	1721
Single unit truck	5090

4. PROPOSED METHOD

The brief architecture of the proposed framework is shown in Figure 2. We proposed a vehicle identification and classification algorithm based on sequential convolutional deep architecture. The proposed is composed of two main parts working successively. (1) Vehicle identification module whose role is the classification of vehicles / non-vehicles in a traffic scene (2) vehicle type classification module, for separation of the different types of vehicles according to their weight and functions where we have opted for four classes (the most present in our highways): car, truck, motorcycle, bus.

4.1 Vehicle identification

Most existing vehicle detection systems under static cameras are based on two steps: foreground segmentation and vehicle classification. Generally, the task of foreground segmentation is to extract from the video stream the regions of interest represented by all moving objects then the detected blobs will subsequently be classified into vehicles or non-vehicles. But such systems cannot detect vehicles parked on a road or a motorway for a reason of breakdown or accident, for example. In this paper, we proposed to merge the two tasks of vehicles detection into one that we called vehicle identification (Figure 2 (1)), where we used a convolutional neural network (CNN) which will scan the whole scene and classify all the regions of the image into two classes: vehicle / non-vehicle.

The images from the video stream of the road traffic scene are firstly scaled to a size proportional to the sliding window and to the number of network inputs fixed at 100 x 100 pixels. Then the CNN will scan the scene with a window of 100 x 100 pixels to separate the foreground (vehicles) from the background, i.e., any object or region present in the traffic scene which is not a vehicle like a highway, trees, sidewalk, pedestrians, road signs … etc.

Since the CNN of vehicle identification is based on the shape and texture of objects, the color of vehicles is therefore optional in our application, so it operates on the grayscale image. This will also make it possible to speed up the processing (vehicle search) and therefore reduce the vehicle identification time to a third (1/3).

Table 2. Model architecture of vehicle identifier CNN

Layer		Feature Map	Size	Kernel size	Stride	activation
Input	Image	1	100x100x1	-	-	-
1	Conv1	96	90x90x96	11x11	1	Relu
	Max pool	96	45x45x96	2x2	1	Relu
2	Conv2	256	35x35x256	11x11	1	Relu
	Max pool	256	17x17x256	2x2	1	Relu
3	Conv3	384	15x15x384	3x3	1	Relu
4	Conv4	384	13x13x384	3x3	1	Relu
5	Conv5	256	11x11x256	3x3	1	Relu
	Max pool	256	5x5x256	2x2	1	Relu
6	FC	-	4096	-	-	Relu
7	FC	-	4096	-	-	Relu
8	FC	-	1000	-	-	Relu
	Output	-	1	-	-	Sigmoid

The model architecture is given in Figure 2 and Table 2. Our CNN model consists of 5 convolutional, three max-pooling, 1

Flatten, one drop out, and three dense layers. Binary cross-entropy for loss function and Adam optimizer were used.

The optimizer used in both networks of our system is adaptive moment estimation (Adam) [37] based on adaptive estimation of the low-order moment, which is widely applied in the machine learning field.

Binary cross-entropy is intended to use with binary classification where the target value is 0 or 1. It compares each of the predicted probabilities to the actual class output, which can be either 0 or 1. It then calculates the score that penalizes the probabilities based on the distance from the expected.

The loss function as shown in (1), $\hat{y}_i$ is the predicted value, y_i is the target, and N is the output size. Output size is the number of scalar values in the model output.

$$Loss = -\frac{1}{N}\sum_{i=1}^{N} y_i \,.\, log\ \hat{y}_i + (1 - y_i).\log(1-\hat{y}_i) \quad (1)$$

The output layer needs to configure with a single node and a “sigmoid” activation, given by "2"; in order to predict the probability for class 1.

$$f(x) = \frac{1}{1+e^{-x}} \quad (2)$$

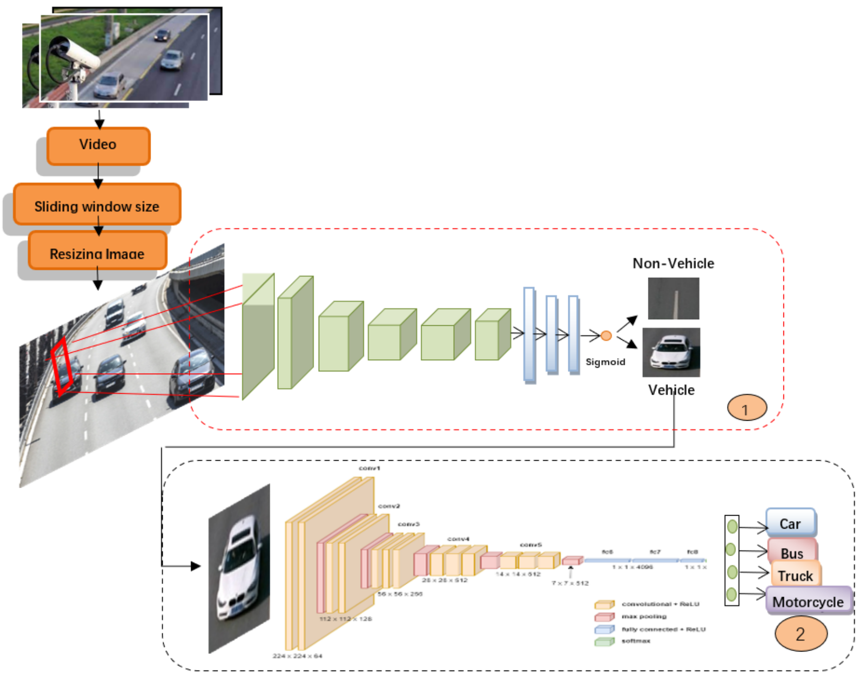

Figure 2. The architecture of proposed framework

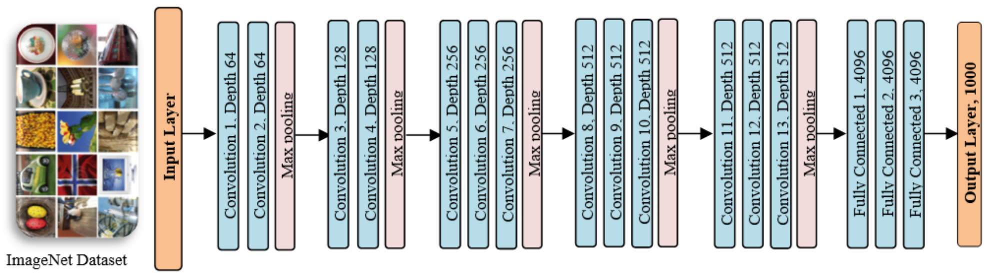

Figure 3. The architecture of the VGG-16 model trained on the ImagetNet dataset

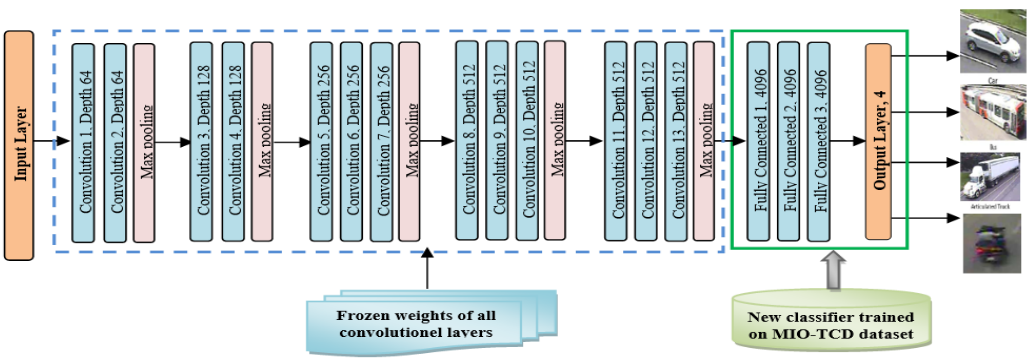

Figure 4. The architecture of VGG-16 with transfer learning

4.2 Vehicle classification type

The vehicle identifier network supplies the regions (blobs) classified as vehicles to the vehicle type classifier network (Figure 2 (2)) to classify these vehicles as cars, motorcycles, trucks, bus.

We used VGG-16 as a classifier vehicle type network, which is pre-trained with ImageNet [38] dataset. Figure 3 and Table 3 show a model architecture that has thirteen convolutional layers and five maximum 'pooling' layers, then three 'fully connected' layers, and finally a 'softmax' classifier with 1000 classes where the last layer is replaced by another with four classes instead of 1000 classes.

In order to increase the performance of the Vgg-16 pre-trained, we have enhanced its training in the four classes mentioned above, using transfer learning as shown in Figure 4. To extract feature vectors from the VGG-16 model, weights of all convolutional layers are frozen, and resulted output is given to a new classifier.

Two steps are necessary for this learning mode:

a. Download the weights from the pre-trained 'Vgg16' model.
b. Train network on the last layers only.

Table 3. Model architecture of vehicle classifier CNN

Layer		Feature Map	Size	Kernel size	Stride	activation
Input	Image	1	224x224x3	-	-	-
1	Conv1	64	224x224x64	3x3	1x1	Relu
2	Conv2	64	224x224x64	3x3	1x1	Relu
	Max pool	128	112x112x128	2x2	2x2	Relu
3	Conv3	128	112x112x128	3x3	1x1	Relu
4	Conv4	128	112x112x128	3x3	1x1	Relu
	Max pool	256	56x56x256	2x2	2x2	Relu
5	Conv5	256	56x56x256	3x3	1x1	Relu
6	Conv6	256	56x56x256	3x3	1x1	Relu
7	Conv7	256	56x56x256	3x3	1x1	Relu
	Max pool	512	28x28x512	2x2	2x2	Relu
8	Conv8	512	28x28x512	3x3	1x1	Relu
9	Conv9	512	28x28x512	3x3	1x1	Relu
10	Conv10	512	28x28x512	3x3	1x1	Relu
	Max pool	512	14x14x512	2x2	2x2	Relu
11	Conv11	512	14x14x512	3x3	1x1	Relu
12	Conv12	512	14x14x512	3x3	1x1	Relu
13	Conv13	512	14x14x512	3x3	1x1	Relu
	Max pool	512	7x7x512	2x2	2x2	Relu
	FC	-	4096	-	-	Relu
	FC	-	4096	-	-	Relu
	FC	-	4096	-	-	Relu
	Output	-	4	-	-	Softmax

5. EXPERIMENTAL RESULTS

In this section, we introduce the details of network training and testing where we used Google Colab [39] as the training environment along with Tensorflow for training our model. We have used the GPU of Google Colab, which is 60 times faster than the CPU. The specifications of CPU runtime offered by Google Colab are Intel Xeon Processor with two cores @ 2.30 GHz and 13 GB RAM. Python and PyCharm were used as the programming language.

After network training and validation, the trained model is imported to perform all the tests on a laptop with an Intel Core i7 and 16 Go RAM.

5.1 Dataset

The training of networks was carried out using the MIO-TCD dataset used differently for each network where in the vehicle identifier network, all vehicles (bus, car, truck, … etc.) are considered as positive examples, while for the class of negative examples, we have considered backgrounds and the pedestrians as represented in Table 4.

In the case of the vehicle classifier network, interest is given to vehicle subclasses such as car, bus, car, truck, and motorcycle, as represented by Table 5.

It should be noted that for each network, the dataset is separated into 80% for the training phase and the remaining 20% for the test phase.

Table 4. Dispatching dataset for vehicle identifier CNN

Classes	Training	Test
Vehicles ['bus', 'car', 'truck', 'motorcycle']	4000	1000
Non-Vehicles ['background', pedestrian']	4000	1000
Total	8000	2000

Table 5. Dispatching dataset for vehicle classifier CNN

Classes	Training	Test
Bus	3550	800
Car	4366	800
Motorcycle	1366	800
Truck	3563	800
Total	16045	3200

5.2 Evaluation metrics

We have implemented CNNs of our proposed system to compare the overall accuracy and loss. Additionally, to evaluate the effectiveness of our proposed networks, we have calculated the confusion matrix.

The following conventional evaluation metrics are used for the confusion matrix:

True Positives (**TP**), which are examples correctly labeled as positive;

True Negatives (**TN**) refer to negative examples correctly labeled as negatives

False Positives (**FP**) refer to negative examples incorrectly labeled as positives;

False Negatives (**FN**), which are positive examples mislabeled as negatives.

Accuracy given by "3": is a proportion of observations correctly predicted to the total observations.

Loss in CNN: is the difference between the predicted output and the actual output. It measures the mistakes made by the network in predicting the output.

$$f(x) = \frac{1}{1 + e^{-x}} \tag{3}$$

In the detection task, the model's prediction is evaluated through the Bounding-Box measure, in which the overlap ratio between the predicted Bounding-Box **Bp** and the ground truth box **Bgt** is calculated. A correct detection is obtained when the overlap ratio Intersection over Union (**IoU**) surpasses 0.5 using Eq. (4)

$$IoU = \frac{area(B_p \cap B_{gt})}{area(B_p \cup B_{gt})} \quad (4)$$

5.3 Training and validation

During the training phase of the vehicles identifier model, the following parameters have been fixed:

Optimizer=Adam, loss=binary_crossentropy, epochs=40, batch_size=32.

A dataset of 10000 samples is used, including 8000 for training and 2000 for validation of the vehicle identifier network. Figure 5 illustrates the training and validation represented by the graph of accuracy and loss, where we notice that the model has reached an accuracy of (98%) during the training and (95%) during the test, and a loss of (0.05%) during the training and (0.28%) for the test.

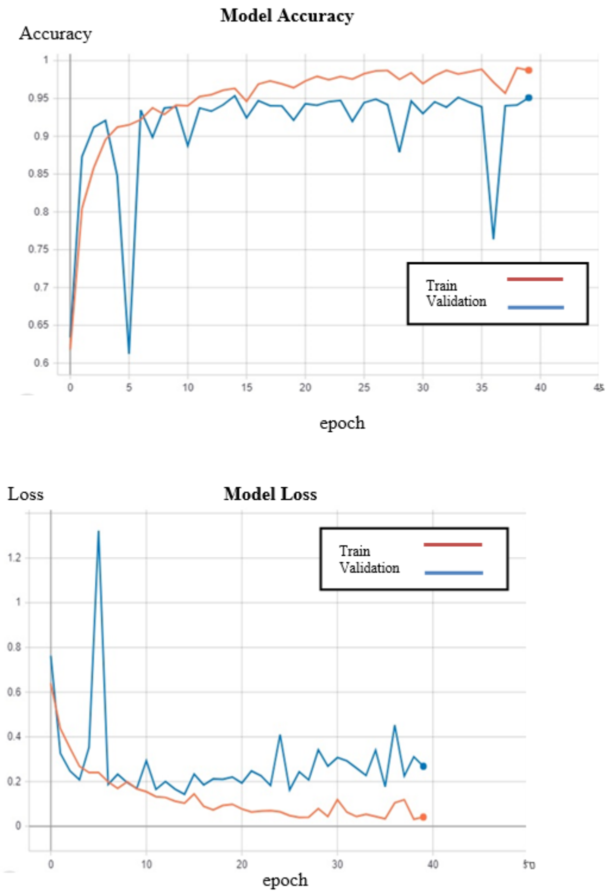

Figure 5. Accuracy and loss for training and validation of vehicle identifier CNN

Table 6. Tests of vehicle identifier network

Rates	Percentage
TP	92%
TN	95%
FP	05%
FN	08%

Tests were carried out on 2000 images (20% of the dataset) that are not used in training. A recap of the rates obtained during the test phase is illustrated in Table 6, where the positive detection rate of vehicle classes reached 92%, the positive detection rate of the non-vehicle class reached 95%, and the false alarms in both classes are markedly weak.

During the training phase of the vehicles classifier model, the following parameters have been fixed:

Optimizer=Adam, loss=categorical_crossentropy, epochs=30, batch_size=32.

Figure 6 illustrates the training and validation represented by the graph of accuracy and loss, where we notice that the model has reached an accuracy of (98%) during the training and (92%) during the test, and a loss of (0.05%) during the training and (0.4%) for the test.

Tests were performed on 3200 images (20% of the dataset) from the four classes not used in training. A recap of the rates obtained during the test phase is illustrated in Table 7 represents the confusion matrix of the rates obtained for each class, where the highest detection rate (98.63%) was obtained for the car class and the lowest for the truck class 87.75% while the false alarms in the four classes are markedly weak.

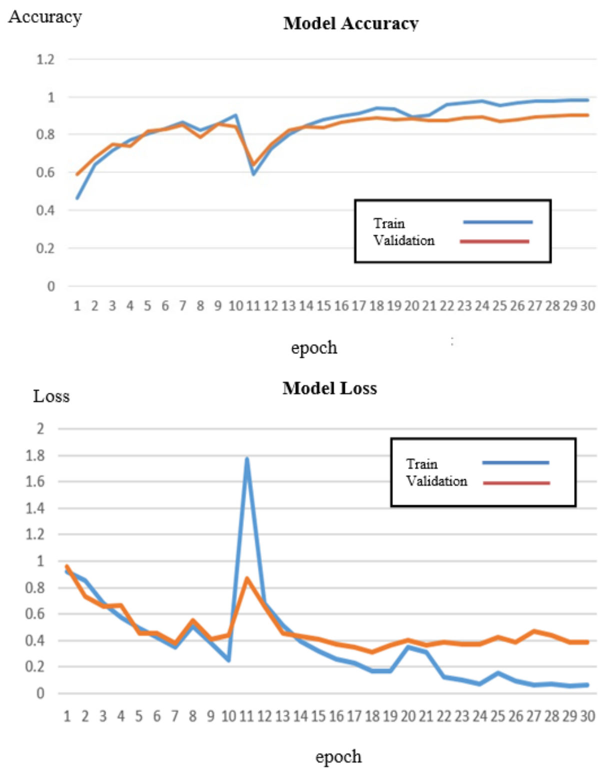

Figure 6. Accuracy and loss for training and validation of vehicle classifier CNN

Table 7. Confusion matrix for vehicle classification

Target Class / *Output Class*	*Bus*	*Car*	*Motor cycle*	*Truck*
Bus	**94.62%**	01.00%	00.13%	04.25%
Car	00.25%	**98.63%**	00.25%	00.87%
Motorcycle	00.00%	03.63%	**92.50%**	03.87%
Truck	06.25%	03.38%	02.62%	**87.75%**

5.4 Tests and analysis

We verify the proposed system for different real highway videos. The tests were carried out on videos filmed from a road intersection, as shown in Figure 7, where (a) original image, (b) Vehicle detection (identification), and (c) vehicle type classification.

From the obtained results, we find the following:

1. The detector (vehicle presence identifier) manages to detect the vehicles which present from different angles of view and different positions.
2. The vehicle type classifier manages to classify the majority of the vehicles present in the scene, whatever their positions, by discarding, of course, the tiny vehicles that we have set at 30 pixels. The obtained overall mean average precision (mAP) is 93.38%.
3. A considerable reduction of processing time (average ten fps) thanks to the master-slave combination, which allows the classifying network to locate its search on a very small area (area of detected vehicle) instead of searching in the entire image.

(a) (b) (c)

Figure 7. Illustration of vehicle detection and classification (a) original image (b) Vehicle detection (c) vehicle classification

It should be noted that:

1. Subject classes are the four most available (present) classes in the Algerian road network.
2. A class can integrate sub-classes of vehicles having the same gauge; for example, the motorcycle class includes motorcycles and bicycles, and the Bus class includes: buses, minibuses…etc.

To evaluate the effectiveness of our proposed network, we compare the proposed network to the state-of-the-art detectors (DenseNet 121, ResNet 90, ResNet 50, Inception V3) on MIO TCD. Figure 8 and Table 8 show the results of our experiment. Where we notice that our network outperforms the other algorithms with an overall mean average precision (mAP) of 93.38%, thus exceeding state-of-the-art by almost 3%.

It is also important to note that our network has a high ability to detect and classify MCs and trucks.

Table 8. Results on the MIO TCD

Class / *Algorithms*	*Bus (%)*	*Car (%)*	*Motor Cycle (MC)*	*Truck (%)*	*Overall mAP (%)*
DenseNet 121	**96.32**	**98.88**	92.32	74.14	90.42
ResNet 90	95.77	98.47	91.52	74.84	90.15
ResNet 50	85.73	97.36	87.79	66.42	84.33
Inception V3	92.25	98.23	88.74	71.35	87.64
Ours	94.62	98.63	**92.50**	**87.75**	**93.38**

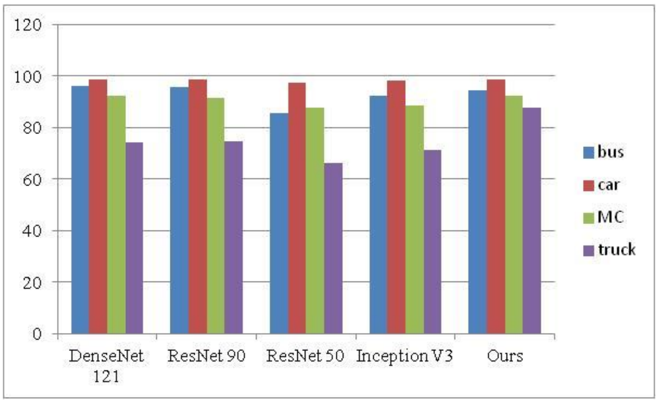

Figure 8. Graphical comparison of our proposed network and the state of the art

6. CONCLUSION

In this paper, we present a new CNN architecture based on a master-slave convolutional neural network for vehicle detection and type classification.

The two networks operate sequentially as master and slave. This separation of the detector and the classifier allowed us to boost the performance of the system because instead of the classifier proceeding in each frame of the video to search for vehicles even when the scene is empty (contains no vehicle) it will therefore classify only vehicles previously detected and located by the master (detector). The latter is responsible for triggering the classifier only in scenes (frames) containing vehicles if not the detector goes on to search in the next frame without triggering the slave (the classifier) this will reduce the tasks and the occupation of the system and thus increase its performance. Vehicles are classified into four categories (bus, car, truck, motorcycle). Tests were performed on 3200 images of the four classes where the best accuracy is obtained for cars (98.63%) and the overall mean average precision is (Map) 93.38% thus exceeding state-of-the-art by almost 3%. It should be noted that these results considered very promising in the field of ITS are due not only to the proposed master-slave architecture but also to the type of network used where we used a shallow network for the detection because its main task is to look for the general shape of the vehicles whereas for the classification, we used a deep network to look for details allowing to discriminate the vehicles previously detected.

In future studies, we intend to carry out a series of tests of our system on the roads and highways of our country to put it into real service for the monitoring of the road network. But preliminary tests on the processing time of the system are essential before putting it into operation in real time.

ACKNOWLEDGMENT

This project is supported by the LAIG laboratory and the 8 Mai 1945 University of Guelma. www.univ-guelma.dz.

We want to thank ATRST for the research grant in Algeria. https://atrst.dz/.

REFERENCES

[1] An, S.H., Lee, B.H., Shin, D.R. (2011). A survey of intelligent transportation systems. In 2011 Third International Conference on Computational Intelligence, Communication Systems and Networks, Bali, Indonesia, pp. 332-337. https://doi.org/10.1109/CICSyN.2011.76

[2] Selvathi, D., Pavithra, P., Preethi, T. (2017). Intelligent transportation system for accident prevention and detection. In 2017 International Conference on Intelligent Computing and Control Systems (ICICCS) Madurai, India, pp. 442-446. https://doi.org/10.1109/ICCONS.2017.8250761

[3] Lin, Y., Wang, P., Ma, M. (2017). Intelligent transportation system (ITS): Concept, challenge and opportunity. In 2017 IEEE 3rd International Conference on big data security on cloud (bigdatasecurity), IEEE international conference on high performance and smart computing (HPSC), and IEEE International Conference on Intelligent Data and Security (ids) Beijing, China, pp. 167-172. https://doi.org/10.1109/BigDataSecurity.2017.50

[4] Zhu, L., Yu, F.R., Wang, Y., Ning, B., Tang, T. (2018). Big data analytics in intelligent transportation systems: A survey. IEEE Transactions on Intelligent Transportation Systems, 20(1): 383-398. https://doi.org/10.1109/TITS.2018.2815678

[5] Guerrero-Ibáñez, J., Zeadally, S., Contreras-Castillo, J. (2018). Sensor technologies for intelligent transportation systems. Sensors, 18(4): 1212. https://doi.org/10.3390/s18041212

[6] Menouar, H., Guvenc, I., Akkaya, K., Uluagac, A. S., Kadri, A., Tuncer, A. (2017). UAV-enabled intelligent transportation systems for the smart city: Applications and challenges. IEEE Communications Magazine, 55(3): 22-28. https://doi.org/10.1109/MCOM.2017.1600238CM

[7] Jin, L., Zhang, B., Zhang, L., Yang, W. (2019). Nanogenerator as new energy technology for the self-powered intelligent transportation system. Nano Energy, 66: 104086. https://doi.org/10.1016/j.nanoen.2019.104086

[8] Rhoades, B.B., Conrad, J.M. (2017). A survey of alternate methods and implementations of an intelligent transportation system. In SoutheastCon 2017 oncord, NC, USA, pp. 1-8. https://doi.org/10.1109/SECON.2017.7925303

[9] Kaffash, S., Nguyen, A.T., Zhu, J. (2021). Big data algorithms and applications in intelligent transportation system: A review and bibliometric analysis. International Journal of Production Economics, 231: 107868. https://doi.org/10.1016/j.ijpe.2020.107868

[10] Loce, R.P., Bala, R., Trivedi, M. (Eds.). (2017). Computer Vision and Imaging in Intelligent Transportation Systems. John Wiley & Sons.

[11] Muthuramalingam, S., Bharathi, A., Gayathri, N., Sathiyaraj, R., Balamurugan, B. (2019). IoT-based intelligent transportation system (IoT-ITS) for global perspective: A case study. In the Internet of Things and Big Data Analytics for Smart Generation 154, pp. 279-300. https://doi.org/10.1007/978-3-030-04203-5_13

[12] Patel, P., Narmawala, Z., Thakkar, A. (2019). A survey on intelligent transportation systems using the internet of things. Emerging Research in Computing, Information, Communication, and Applications, 231-240.

[13] Chand, H.V., Karthikeyan, J. (2018). Survey on the role of IoT in the intelligent transportation system. Indonesian Journal of Electrical Engineering and Computer Science, 11(3): 936-941. https://doi.org/10.11591/ijeecs.v11.i3.pp936-941

[14] Guerrero-Ibanez, J. A., Zeadally, S., Contreras-Castillo, J. (2015). Integration challenges of intelligent transportation systems with the connected vehicle, cloud computing, and internet of things technologies. IEEE Wireless Communications, 22(6): 122-128. https://doi.org/110.1109/MWC.2015.7368833

[15] Chaudhary, R., Jindal, A., Aujla, G.S., Aggarwal, S., Kumar, N., Choo, K.K.R. (2019). BEST: Blockchain-based secure energy trading in an SDN-enabled intelligent transportation system. Computers & Security, 85: 288-299. https://doi.org/10.1016/j.cose.2019.05.006

[16] Gao, J., Agyekum, K.O.B.O., Sifah, E.B., et al. (2019). A blockchain-SDN-enabled Internet of vehicles environment for fog computing and 5G networks. IEEE Internet of Things Journal, 7(5): 4278-4291. https://doi.org/110.1109/JIOT.2019.2956241

[17] Mollah, M.B., Zhao, J., Niyato, D., et al. (2020). Blockchain for the internet of vehicles towards intelligent transportation systems: A survey. IEEE Internet of Things Journal, 8(6): 4157-4185. https://doi.org/10.1109/JIOT.2020.3028368

[18] Ferdowsi, A., Challita, U., Saad, W. (2019). Deep learning for reliable mobile edge analytics in intelligent transportation systems: An overview. IEEE Vehicular Technology Magazine, 14(1): 62-70. https://doi.org/10.1109/MVT.2018.2883777

[19] Guerrero-Ibañez, J., Contreras-Castillo, J., Zeadally, S. (2021). Deep learning support for intelligent transportation systems. Transactions on Emerging Telecommunications Technologies, 32(3): e4169. https://doi.org/10.1002/ett.4169

[20] Sirohi, D., Kumar, N., Rana, P.S. (2020). Convolutional neural networks for 5G-enabled intelligent transportation system: A systematic review. Computer Communications, 153: 459-498. https://doi.org/10.1016/j.comcom.2020.01.058

[21] Lv, Z., Zhang, S., Xiu, W. (2020). Solving the security problem of intelligent transportation system with deep learning. IEEE Transactions on Intelligent Transportation Systems, 22(7): 4281-4290. https://doi.org/10.1109/TITS.2020.2980864

[22] Wang, X., Liu, J., Qiu, T., Mu, C., Chen, C., Zhou, P. (2020). A real-time collision prediction mechanism with deep learning for the intelligent transportation system. IEEE Transactions on Vehicular Technology, 69(9): 9497-9508. https://doi.org/10.1109/TVT.2020.3003933

[23] Yu, K., Lin, L., Alazab, M., Tan, L., Gu, B. (2020). Deep learning-based traffic safety solution for a mixture of autonomous and manual vehicles in a 5G-enabled intelligent transportation system. IEEE Transactions on

Intelligent Transportation Systems, 22(7): 4337-4347. https://doi.org/10.1109/TITS.2020.3042504
[24] Pustokhina, I.V., Pustokhin, D.A., Rodrigues, J.J., et al. (2020). Automatic vehicle license plate recognition using optimal K-means with convolutional neural network for intelligent transportation systems. Ieee Access, 8: 92907-92917. https://doi.org/10.1109/ACCESS.2020.2993008
[25] Xia, Y., Feng, J., Zhang, B. (2016). Vehicle Logo Recognition and attributes prediction by multi-task learning with CNN. In 2016 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD), Changsha, China, pp. 668-672. https://doi.org/10.1109/FSKD.2016.7603254
[26] Kumar, N., Rahman, S.S., Dhakad, N. (2020). Fuzzy inference enabled deep reinforcement learning-based traffic light control for intelligent transportation system. IEEE Transactions on Intelligent Transportation Systems, 22(8): 4919-4928. https://doi.org/10.1109/TITS.2020.2984033
[27] Kumar, N., Rahman, S. S. (2019). Deep reinforcement learning with vehicle heterogeneity based traffic light control for intelligent transportation system. In 2019 IEEE International Conference on Industrial Internet (ICII) Orlando, FL, USA, pp. 28-33. https://doi.org/10.1109/ICII.2019.00016
[28] Yang, Z., Pun-Cheng, L.S. (2018). Vehicle detection in intelligent transportation systems and its applications under varying environments: A review. Image and Vision Computing, 69: 143-154. https://doi.org/10.1016/j.imavis.2017.09.008
[29] Kiran, V.K., Parida, P., Dash, S. (2020). Vehicle detection and classification: A review. In International Conference on Innovations in Bio-Inspired Computing and Applications, pp. 45-56. https://doi.org/10.1007/978-3-030-49339-4_6
[30] Zohra, A.F., Kamilia, S., Fayçal, A., Souad, S. (2013). Detection and classification of vehicles using deep learning. International Journal of Computer Science Trends and Technology (IJCST), 6(3): 23-29.
[31] Tsai, C.C., Tseng, C.K., Tang, H.C. Guo, J.I. (2018). Vehicle detection and classification based on deep neural network for intelligent transportation applications. In 2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), Honolulu, HI, USA, pp. 1605-1608. https://doi.org/10.23919/APSIPA.2018.8659542
[32] Chen, L., Ye, F., Ruan, Y., Fan, H., Chen, Q. (2018). An algorithm for highway vehicle detection based on a convolutional neural network. Eurasip Journal on Image and Video Processing, 2018: 1-7. https://doi.org/10.1186/s13640-018-0350-2
[33] Seo, C.J. (2019). Vehicle detection and car type identification system using deep learning and unmanned aerial vehicle. International Journal of Innovative Technology and Exploring Engineering, 8: 2278-3075.
[34] Asvadi, A., Garrote, L., Premebida, C., Peixoto, P., Nunes, U.J. (2017). DepthCN: Vehicle detection using 3D-LIDAR and ConvNet. In 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC) Yokohama, Japan, pp. 1-6. https://doi.org/10.1109/ITSC.2017.8317880
[35] Geiger, A., Lenz, P., Urtasun, R. (2012). Are we ready for autonomous driving? the kitti vision benchmark suite. In 2012 IEEE conference on computer vision and pattern recognition, Providence, RI, USA, pp. 3354-3361. https://doi.org/10.1109/CVPR.2012.6248074
[36] Luo, Z., Branchaud-Charron, F., Lemaire, C., et al. (2018). MIO-TCD: A new benchmark dataset for vehicle classification and localization. IEEE Transactions on Image Processing, 27(10): 5129-5141. https://doi.org/10.1109/TIP.2018.2848705
[37] Kingma, D.P., Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.
[38] Krizhevsky, A., Sutskever, I., Hinton, G.E. (2017). Imagenet classification with deep convolutional neural networks. Communications of the ACM, 60(6): 84-90. https://doi.org/10.1145/3065386
[39] Carneiro, T., Da Nóbrega, R.V.M., Nepomuceno, T., Bian, G.B., De Albuquerque, V.H.C., Reboucas Filho, P.P. (2018). Performance analysis of google colaboratory as a tool for accelerating deep learning applications. IEEE Access, 6: 61677-61685. https://doi.org/10.1109/ACCESS.2018.2874767