

Detection And Classification Of Vehicles Using Deep Learning

Adami Fatima Zohra ^[1], Salmi Kamilia ^[2], Abbas Fayçal ^[3], Saadi Souad ^[4]

Department Of Computer Science

University of Abbes Laghrour khenchela

Algeria

ABSTRACT

In this paper, we focus on detection and recognition of vehicles from a video stream. Compared to traditional methods of object detection and classification, Deep learning methods are a new concept in the field of computer vision. Our model operates in two stages: a data preparation step, it consists of applying Treatments on the images composing the dataset in order to extract the characteristics, the second step is to apply the concept of convolutional neural networks to classify vehicles. Our method gave better results in terms of precision, detection and classification where we obtained an accuracy of 99.2%.

Keywords :- Vehicle detection and recognition, deep learning, classification, convolutional neural networks (CNN).

I. INTRODUCTION

Vehicle detection allows the use of various applications of artificial intelligence system for several purposes, especially: intelligent transportation, automatic monitoring, autonomous driving, and driver safety guarantee. The purpose of this article is to allow us to detect vehicles moving in front of us via a camera placed under the rearview mirror and draw the trajectory lines of our vehicle. In this work, we focus on the detection and recognition of vehicles in a video stream. For this reason, we have used the convolutional neural network technique (CNN) and a dataset that contains images to enable recognition and classification of vehicles. The main purpose of our work is therefore to reduce human effort and offers research perspectives to make driving more enjoyable and almost autonomous. Our paper is organized as follows: the next part will discuss previous work that uses learning methods for vehicle detection, then we will explain our proposed architecture (model), then we will calculate the estimate of the deviation of the vehicle, in the following part we will present the results obtained from this research. Finally, we will finish our work with a conclusion and we will propose perspectives.

II. PREVIOUS WORK

Convolutional neural networks are inspired from the visual brain cortex; they are used in recognition systems, robotics and self-driving cars as well as in other areas. They provide good results and a higher accuracy rate compared to traditional methods. A vehicle detection method presented by Chabot.F and al whose goal is to recognize the brand and model of vehicles. This approach is based sure setting

correspondence between the vehicle in the picture and real 3D model. This method is based on a detector based on convolutional neural networks (CNN); its principle is based on the extraction of the points of interest corresponding to predefined parts on the image of the vehicle. These points will then be filtered and matched with the points of the 3D model [1]. A method of detecting vehicles from high resolution aerial imagery called HEM whose learning process uses convolutional neural networks. This method is proposed by Koga.Y and al [2]. Yang.M and al have adopted the use of a double-focal loss function (DFL-CNN) for the detection of vehicles in aerial images. This method is used to improve network capacity in order to distinguish vehicles in a crowded scene [3]. Zhou.Y and al are working on the problems of detection and classification of vehicles using deep neural network (DNN) approaches. They answer the three questions specific to their application, the first question is asked about the use of DNN for vehicle detection, the second question about useful functions for the classification of vehicles and one last question is how to extend a model on a set of limited size of data in the case of extreme lighting [4]. Mengxi.W used the algorithm You Only Look Once (YOLO) to detect vehicles from a video stream dashboard camera [5].The YOLO algorithm is developed by J.Redmon and al in 2015. This algorithm is used for the detection of objects, it is based on two stages; the first is to detect objects by convolutional neural networks, the second step makes the grid of the image and does the prediction of the detected object class if it exists. The advantage of this prediction is that it can be done independently since the first detection but that she does not target objects in particular. The disadvantage of this method is the difficulty of detecting the smallest objects as well as objects that are

overlapping [6].The Faster R-CNN method is created by S.Ren and al in 2015 it is based on convolutional neural networks. She is considered as the first detector tested for the detection of apples [6].

III. OUR MODEL

Convolutional neuron network efficiency depends to a large extent on the quality of the training data set; the network will produce good results only if the training data used contain sufficient important characteristics so that they can produce new predictions.

A. The dataset used

In order to evaluate our method, the dataset used includes two files (vehicle file and non-vehicle file). Our database has images extracted by video sequences (obtained by a front camera mounted on a car). To ensure good data learning, images are captured in different road conditions (far, near, left, right). The vehicle file includes 8798 images and the non-vehicle file includes 8971 images. Each image is of dimension: 64 * 64 pixels.

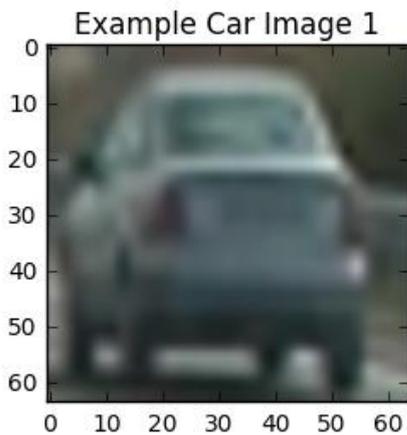


Fig. 1 random image from the class vehicle of our data set.

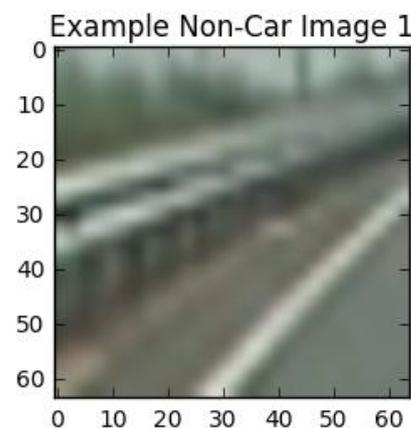


Fig.2 random image from the class Non-Vehicle of our data set.

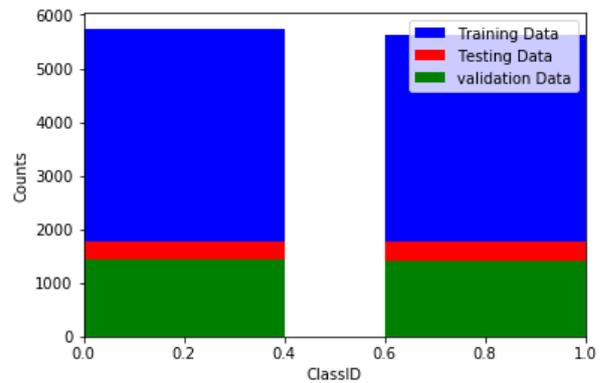


Fig. 3 Histogram of the database

The resemblance between the left part and the right part of the histogram happened because the number of non-vehicle images that is more close to the vehicle image number.

B. Model architecture

Our model is composed of six layers of convolution, The first five convolutional layers use a ReLU activation function, knowing that there are several activation functions such as (sigmoid, TanhLeaky, ReLU, ... etc.) at the output of each of these layers, the Dropout layer is used to regularize the CNN. We add to the fifth convolutional layer the MaxPooling2D layer that will be explained later. The convolutional sixth layer uses a sigmoid activation function.

TABLE I
DESCRIPTION OF THE CNN USED.

layer	Description	Output Shape
1	Conv2D (filters (8), kernel_size(3,3), activation(ReLU), name=C1, padding (same)) dropout (0.5)	(None, 64, 64, 8)
2	Conv2D (filters (16), kernel_size(3,3), activation(ReLU), name=C2, padding (same)) dropout (0.5)	(None, 64, 64, 16)
3	Conv2D (filters (32), kernel_size(3,3), activation(ReLU), name=C3, padding (same)) dropout (0.5)	(None, 64, 64, 32)
4	Conv2D (filters (64), kernel_size(3,3), activation(ReLU), name=C4, padding (same)) dropout (0.5)	(None, 64, 64, 64)
5	Conv2D (filters (128), kernel_size (3,3), activation(ReLU), name=C5, padding(same), MaxPooling2D(8, 8)) Dropout (0.5)	(None,64,64,128)
6	Conv2D(filters(1),kernel_size(8,8), activation(sigmoid), name=Cf)	(None, 1, 1, 1)

c. The convolutional neural network of our model

We present in this part the different modules used in the CNN, these modules are: convolution layers and activation functions.

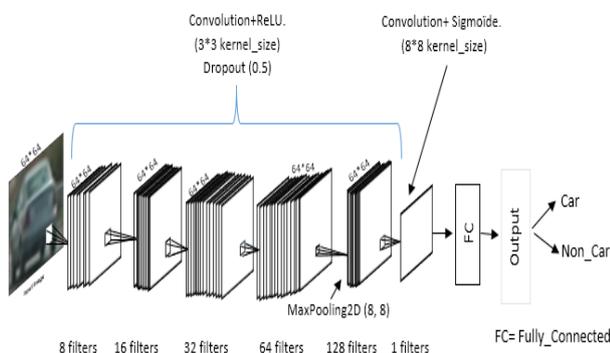


Fig. 4 illustrating the CNN architecture of our model.

1) Convolution layer:

The convolution layer is the main layer of CNN, which applies a specified number of convolution filters to the input image. This layer allows performing a set of mathematical operations to produce a unique value at the output, and then it applies an activation function.

Equation (1) explains how to calculate the output of a given neuron in a convolutional layer [7].

$$g_{i,j,k} = b_k + \sum_{u=1}^{f_h} \sum_{v=1}^{f_w} \sum_{k'=1}^{f_{nr}} x_{i',j',k'} \cdot w_{u,v,k',k} \quad \text{avec} \quad \begin{cases} i' = u * s_h + f_h - 1 \\ j' = v * s_w + f_w - 1 \end{cases} \quad (1)$$

- $g_{i,j,k}$ Is the output of the neuron located in row i , column j in feature map k of the convolutional layer (layer l).
- s_h and s_w are the vertical and horizontal strides, f_h and f_w are the height and width of the receptive field, and f_{nr} is the number of feature maps in the previous layer (layer $l - 1$).
- $x_{i',j',k'}$ Is the output of the neuron located in layer $l - 1$, row i' , column j' , feature map k' (or channel k' if the previous layer is the input layer).
- b_k Is the bias term for feature map k (in layer l). You can think of it as a knob that tweaks the overall brightness of the feature map k .
- $w_{u,v,k',k}$ Is the connection weight between any neuron in feature map k of the layer l and its input located at row u , column v (relative to the neuron's receptive field), and feature map k' .

It is possible to improve the effectiveness of treatment by intercalating in each treatment layer a mathematical function called activation function. In our work, we used two types of activation functions: The ReLU function [8], which will introduce nonlinearities in the model equation (2), and the sigmoid function [9] that will arrange the classes equation (3). There are two types of class: A class vehicle and a class non-vehicle.

$$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases} \quad (2)$$

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

f: The activation function and x: the value of the neuron.

2) Dropout:

This is a regularization approach in neural networks, which helps to reduce interdependent learning between neurons [10]. It allows to be disabled randomly neurons during the different learning iterations to improve the compatibility of neural networks [11].

3) **The Maxpooling Layer:**

The pooling layer is typically used after the convolutional layer. It is used to reduce the dimensions of an image in order to reduce the calculation time and minimize the occupied memory space. The formula (4) below illustrates the calculation of the pooling size [12].

$$W_2 = \frac{W_1 - F}{S} + 1$$

$$H_2 = \frac{H_1 - F}{S} + 1$$
(4)

W1, H1: the size of the input volume, F: The spatial size of the output volume, S: the pitch, and W2, H2: The size of the output volume.

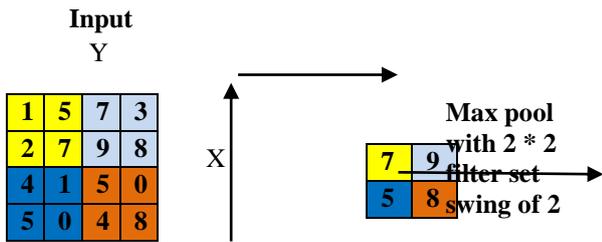


Fig. 5 Max-pooling Layer.

In the last step, we used fully connected layer (FC) to classify the data from the previous layer.

A. **Heatmap operation**

The heatmap function is usually used to extract the hot areas in an image. We used this function to detect vehicles running in front of the camera on the road.



Fig. 6 Heat Map illustrate the true prediction of our model .

C. **Calculate estimated vehicle Offset:**

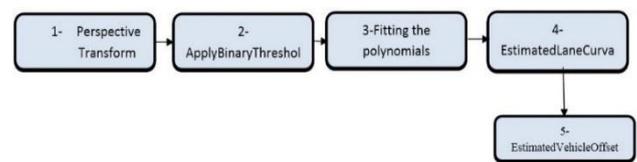


Fig. 7 The stages to calculate estimated vehicle offset:

In the first step, we transform the image of the undistorted road into an image in the form of "bird's-eye" which focuses only on the ends of the lane, and then displays the ends in such a way that they seem to be relatively parallel to each other. In the second step, we converted the image obtained in the previous step into other color spaces and then we converted the images to a shade of gray that highlight only the lines of the path and ignore everything else. To reach this objective we applied color channels (HLS, lab, LUV) and we specified the thresholds, which allowed identify the lines of the way in the images. The objective of the third step is to identify the peaks in a histogram of the image to determine the location of each line of the road, and then it identifies all the nonzero pixels around the peaks. In the fourth step, we calculated the radius of curvature, Based on the work of the authors [13],[14], the formula of the radius of curvature at any point x of the curve $y = f(x)$:

$$f(x) = \frac{\left[1 + \left(\frac{dy}{dx}\right)^2\right]^{3/2}}{\left|\frac{d^2y}{dx^2}\right|}$$
(5)

In the last step, we assume that the camera is mounted in the center of the car below the mirror. The offset is calculated so that the center of the road is the midpoint of the two lines that we have detected. The offset is therefore the difference

between the center of the car and the center of the lane see Fig.8.

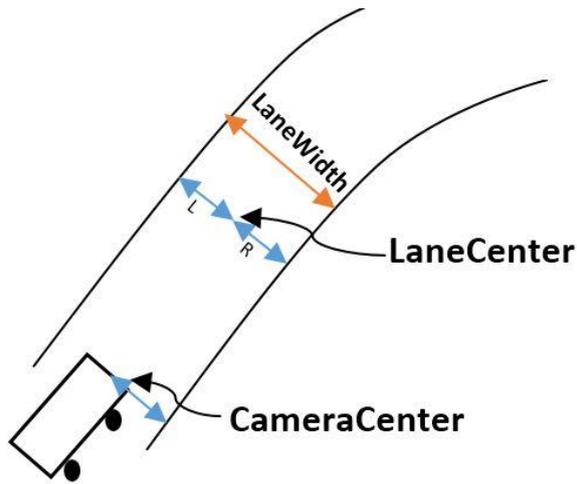
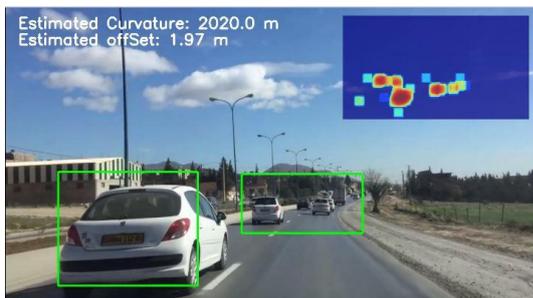


Fig. 8 The offset computing of the vehicle in circulation.

IV. RESULTS

In this part, we evaluate our model in terms of precision and performance to show the effectiveness of our approach.

The figure below shows us some results obtained by our model.



(a)



(b)

Fig. 9 (a),(b) Vehicle detection by our approach.

The previous figures shows that vehicle detection was well done by our model.



(a)



(b)

Fig. 10 (a), (b) yellow surface show the detection of the road line.

In this figure, we find that our system has specified the road on which the vehicle must circulate.

In the following figure 11, we show a graph that represents the evolution of precision curve on the learning set and the validation set.

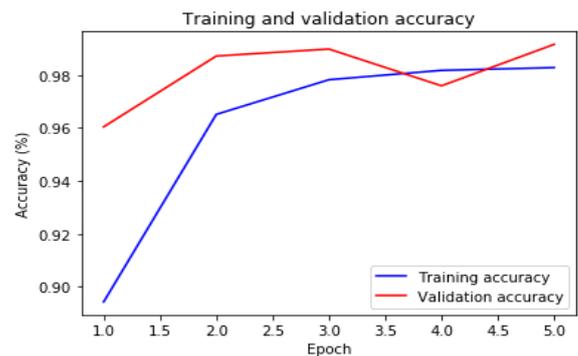


Fig. 11 Accuracy of training and accuracy of validation.

We find that accuracy increases progressively according to a number of epoch, and we observe that there is a gap between the two curves; this means that the model is become generalized in a better way especially from the fourth epoch.

In the following, we show a graph that represents the curve evolution of the error function (Loss) on the learning set and the validation set.

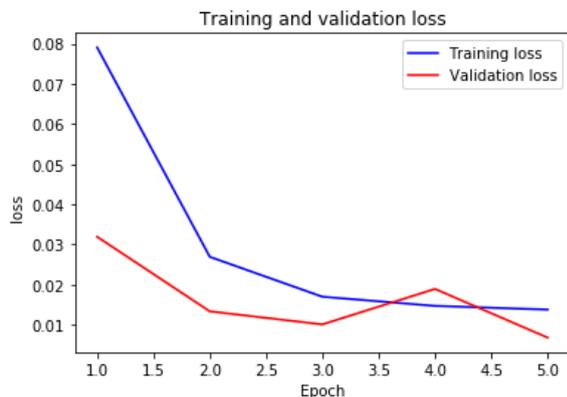


Fig. 12 The training error and the validation error.

Fig. 12 shows that the error function decreases as the number of epoch increases.

This means that the model gives a good prediction on the learning set as well as on the validation set.

We find that the error function on the validation set begins to increase gradually from the fourth epoch; this means that the model is over-adjusted.

We conclude that our model is better by reaching a satisfactory accuracy rate that reaches the 99.2% and very low error rate that equals 0.8%. Our model gives good results in terms of detection and recognition of vehicles in a video stream.

V. CONCLUSION

In this article, a method of recognition and detection of vehicles was present. The proposed algorithm uses a deep learning approach based on the convolutional neural network (CNN). We have arrived at our goal through two phases. The first phase consists of performing pretreatments on the learning game. The second phase consist of a convolutional network in order to extract the characteristics and a fully connected neuron network for recognition and detection of vehicles. We have shown that our method of work greatly improves the accuracy rate and decreases the error rate, However despite the use of regularization, standardization and optimization techniques, the training time of our model remains a problem to raise.

In future work, we want to improve the method by using a pre-trained neural network this will allow us to increase the performance of our model, Thus we plan to take advantage of the immense computing power of the GPU graphics processors by distributing the calculations on several GPUs.

REFERENCES

- [1] F. Chabot, M. Chaouch, J. Rabarisoa, T. Chateau, and C. Trulière, "Détection de pose de véhicule pour la reconnaissance de marque et modèle," in *Journées francophones des jeunes chercheurs en vision par ordinateur*. 2015.
- [2] Y. Koga, H. Miyazaki, and R. Shibasaki, "A CNN-Based Method of Vehicle Detection from Aerial Images Using Hard Example Mining," *Remote Sensing*, vol. 10, pp. 124.2018.
- [3] M. Y. Yang, W. Liao, X. Li, and B. Rosenhahn, (2018, Feb). "Vehicle Detection in Aerial Images," Available: <https://arxiv.org/abs/1801.07339>.
- [4] Y. Zhou, H. Nejati, T. T. Do, N. M. Cheung, and L. Cheah, "Image-based vehicle analysis using deep neural network: A systematic study," in *Digital Signal Processing (DSP), 2016 IEEE International Conference on*. IEEE p. 276-280.
- [5] W. Mengxi, (2017). Real time vehicle detection using YOLO. *The Medium website*.www.medium.org.
- [6] A. Dore, M. Devy, A. Herbulot, (2017, Aug). "Détection d'objets en milieu naturel : application à l'arboriculture," Available : <https://hal.archives-ouvertes.fr/hal-01579391>.
- [7] A. Géron, *Hands-On Machine Learning with Scikit-Learn & TensorFlow, 1st ed, O'Reilly Media*, March 2017.
- [8] W. Xiang, H. D. Tran, and T. T. Johnson, (2017, Dec). "Reachable Set Computation and Safety Verification for Neural Networks with ReLU Activations," Available: <https://arxiv.org/abs/1712.08163>.
- [9] P. Ø. Kanestrøm, "Traffic flow forecasting with deep learning," *MS thesis, Norwegian University of Science and Technology*, NTNU, 2017.
- [10] A. Budhiraja, (2016, Dec). Dropout in (Deep) Machine learning. *The website Medium*, www.medium.org.
- [11] F.Chabot. "Analyse fine 2D/3D de véhicules par réseaux de neurones profonds," *Thèse de doctorat. Clermont Auvergne*, 2017
- [12] Y. LeCun, L. Bottou, Y. Bengio, and P .Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE* vol. 86, pp. 2278-2324. 1998.
- [13] S. Zamfir, R. Drosescu, and R. Gaiginschi, "Practical method for estimating road curvatures using onboard GPS and IMU equipment," in *IOP Conference Series: Materials Science and Engineering*, IOP Publishing, 2016, Vol. 147, No. 1, p. 012114.

- [14] C.Jin, X.Wang, Z.Miao, and S.Ma, "Road curvature estimation using a new lane detection method," *In Chinese Automation Congress (CAC)*, pp. 3597-3601, Oct.2017.