International Journal of Latest Engineering Research and Applications (IJLERA) ISSN: 2455-7137

Volume – 07, Issue – 05, May 2022, PP – 79-84

Iraqi Multi-style License Plate Recognition System using Efficient Dets Model

Rasool Hasan Finjan[1], Ali Salim Rasheed[2]

[1,2]Department of Media Technology and Communications Engineering, College of Engineering, University of Information Technology and Communications, Baghdad, Iraq

Abstract: License plate recognition is one of the great challenges that come with the applications of computer vision, due to ambient conditions such as lack of lighting, motion blur, and deviation of the shooting angle. In this paper, we deal with vehicle plates in the State of Iraq, due to the variety of forms of plates in terms of design, size, and color. Which was designed to contain numbers, letters, and names of the provinces were written in Arabic. Where there is a new and old design, and there is also a special design in the northern region of the same country. This requires us to huge data for all these categories to obtain a high performance in the process of discrimination. A system has been proposed consisting of two parts, the first part is responsible for the process of detecting the plates, in which the SSD model with ResNet-50-FPN was used, and the second part is responsible for the process of discrimination, in which we used three models of efficientdet_d0 and the reason for this is that we have three types of plates and each Some of them are of different sizes, so we assigned each category a model to complete the process. Experimental results show that the proposed method is promising for license plate recognition.

Keywords: Computer Vision, Deep Learning, Image Classification, License plate recognition.

I. INTRODUCTION

With the growing population increasing dramatically, the number of vehicles is increasing due to their appliance in daily life. Therefore, there was a need to create automatic systems to identify vehicles through license plate number, which is unique for each one. That can be used in parking systems, traffic control systems, detection of stolen vehicles, etc. In every country, there is a special design on the number plate that may contain numbers only or letters and numbers together. In Iraq country, each type of car (car – taxi - truck) contains a specific color and size, in Fig.1, we see three main categories of number plates in Iraq, (a) the new style, which contains three types (car – taxi - truck) respectively, each one of them distinguished by a different vertical color font, (b) the old style, also contains three types and distinguished by different color font, (c) the north style, it contains one type only. As we note from the figure, the three types vary in length and width. Iraqi number plates of all types can be considered a major challenge in this field, the first reason is that there are so many different styles, where they are dimensionally different, and the second as we note in Figure1, there are several discriminatory identifiers in the plate, for example, in (a), the first line includes Arabic and word numbers, the second line includes English numbers, and the third line includes the name of the state and the type of vehicle also in Arabic. This means that it requires us to have a huge dataset and many indicators for each image in it. Therefore, many techniques have emerged to deal with this need that comes within the field of digital image processing and computer vision, and the latest of these technologies are deep learning techniques [1], Which has proven very effective in the field of image recognition [2], object detection [3], image segmentation [4], and that have been able to overcome earlier problems such as ambient conditions in the image such as lack of lighting [5], noise and weather conditions [6] that reduce the quality of the captured images and thus the accuracy of the algorithm will decrease in the processing of vehicle number plates. The automatic system process is generally divided into two parts: the detection process, which is deals with localizing the position of the vehicle plate in the captured image using a specific algorithm, and then the output will be the presence of the plate with the coordinates of its location in the image. The second part is the recognition process, Responsible for identifying the content of the plate from numbers and words and their segmentation and then identifying each segment individually, and this is also done using a specific algorithm.

Figure 1 Examples of Iraqi License Plates

II. Related Work:

Many previous works applied to deal with the problem of detecting number plates and distinguishing their contents. various types of techniques were used to solve these problems, but in this scientific paper, we will review the works that used deep learning methods to deal with these problems, because their work is related to the method proposed in this paper.

In [7], a modified version of YOLO has been proposed called YOLO-LPD to be convenient to work in the detection of number plates where was divided the input image using an 11x11 grid to ensure that the small license plates be detected. In addition, to avoid false positives, each grid only predicts one bounding box in the image. In [8], a paper introduced an automatic system for license plate detection and a recognition-based learning method, which divided the pipeline into three parts: detection, segmentation, and recognition. In the detection part, many preprocessing is applied to the image to improve the quality that becomes an input to the CNN classifier model to distinguish between LP and non-LP. In the segmentation part, a few steps are applied to the image to extract the numbers to get ready for the last step of recognizing it by using the CNN model with 37 classes this system is tested on two datasets and gets the result mentioned in the paper.

In [9], the authors suggested that the study focuses only on the detection process, where it is suggested to use two CNNs, a shallow CNN and a deep CNN. A shallow CNN examines the entire image to eliminate unwanted regions quickly, and a deep CNN is used to classify the remaining regions to detect license plates. This method is tested by the BIT-Vehicle dataset, which gave a fast, and accurate result. The study by [10] focused on the use of the a deep neural network for the detection step Where the Faster-RCNN was used, the region proposal network is the core of this type for license plate detection task. One of the advantages of this type is the speed of detection and acceptable accuracy, while for the recognition step, Images detected have been processed first by using conventional image processing methods, and the final step is to recognize the filtered images by using LSTM Tesseract OCR engine.

In [11], the author have developed a lightweight convolutional network to work in the detection and recognition system for the license plate that theoretically decreases the computational complexity. On the other side, this model can be run on different platforms such as GPU, CPU, and FPGA, it can work perfectly in real-time on dedicated devices with low power. It obtained a high accuracy result of up to 96%. In [12], proposed a novel automatic number plate recognition system under the name VSNet, it is composed of two CNNs, VertexNet for license plate detection and SCR-Net for license plate recognition. VertexNet accepts the input image with a small size of 256x256to address the problem of time consumption during the detection process. SCR-net is a classifier that predicts the characters by a forward pass. The architecture for this work used a horizontal encoding method for the left to right feature extraction and a weight-sharing classifier for character recognition. Promising results have been achieved in both the detection and recognition processes compared to the previous works that were presented in their research.

III. Proposed Method

The proposed method for automatic number plate detection and recognition system is composed of two main sections, which is mentioned as follows:

License Plate Detection (LPD):

We have been choosing SSD with resnet-50-FPN as a backbone for the detection step [13]. This model was used from Tensor flow-Model-Zoo Pre-trained on the MS-COCO dataset [14].

This model generates bounding boxes and confidence scores for each image. After the model is fed with the input image, the license plate will be detected and categorized in the style of the detected plate (New-Old-North), as shown in Fig.2. And then we crop the part of the detected image that contains the license plate and fed it to the second stage (recognition of the numbers and words), which consists of three models, each model trained to read the contents of one of the three types of the license plates. And through the type of detected plate, we choose the model to which we fed the image of the license plate.

From this motivation, it becomes obvious that the image will pass in each case two trained models, the first representing the first stage (detection of the license plate) and the second representing the second stage (recognition of the numbers and words).This will take additional time, and this is a serious challenge. To address this challenge, we resorted to using model SSD with resnet-50-FPN. This model is suitable in terms of speed and accuracy, especially if we know that we did not face the problem of the small size of the detected plates.

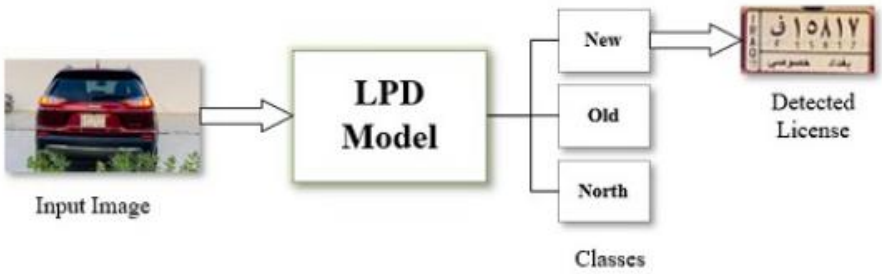

Figure 2 LP Detection System

License Plate Recognition (LPR):

After the license plate is detected, it is cropped, and make some modifications to it, we increase its size to qualify it as an input to the second stage (reading numbers).

Our task is to recognize the detected plate and which category belongs to it (New-Old-North), we propose using a specific model for each category as shown in Fig.3.

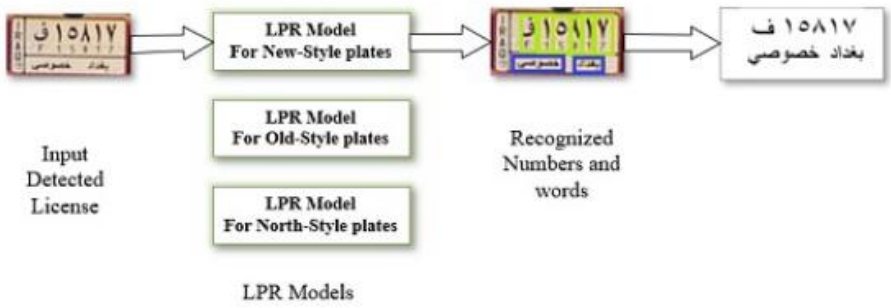

Figure 3 LP Recognition System

The reason for this is the discrepancy in the dimensions of the plate detected for each of the three types. So, in sum, we will have three trained models.

We used three pre-trained models on coco-dataset data of type efficientdet_d0_coco17_tpu-32 [15].This model is located within a group of models called (Efficient Dets) and it consists of eight models starting from (d0) to (d7), and it is used in objects detection tasks and achieves a mean average precision of 55.1mAP according to COCO test. Our model (d0) has a speed of 39 ms and an accuracy of 33.6 coco mAP, as mentioned in the table in the file (tf2_detection_zoo.md) of the Google platform called (Tensorflow2_object_detection_api).

This model contains three basic components as it uses in the classification stage a deep network called (EfficientNet) that extracts the features or characteristics of the input image, and this network is the backbone of this type of model, it also uses a featured network, which it takes multiple levels of features from the backbone as input and has as an output a set of fused features that represent the salient characteristics of the image. It also uses a final stage (class/box network), this stage takes the fused features as its entry data and performs two separate tasks: detecting objects and determining their location.

IV. EXPERIMENTS AND RESULTS

Because there are no specific sources, the internet was our only source as we collected images from different locations (car galleries, parks, gas stations, as well as images we take from a video collection showing scenes from the streets in Iraq). More than 5000 images were collected, where we allocated 1500 images for the data of the first stage, considering that there are three types of plates (Classes) to be detected, so we allocated 500 images for each category. The images for the second stage data are as follows: (a) 3500 images of training and checking the model that performs the task of recognition numbers in New Style plates. This number of images is considered normal, as the total of the items is 43 items. These categories are 10 categories representing the numbers from 0 to 9, as well as 15 categories representing the letters used, and 15 categories representing the Iraqi governorates except for the northern states, and finally 3 categories representing the type of vehicle (car – taxi - truck). (b) 1282 An image of training and checking the model that performs the task of recognizing the numbers in the Old-Style plates. This number is proportional to the number of categories for these plates, which is 25 categories, as 10 of them represent numbers from 0 to 9, and 15 items represent the governorates except for the northern states.

(c) 600 images for training and examining the model that performs the task of recognizing the numbers in the plates of the region, as the number of items is 13 types, 10 for the numbers and 3 for the northern states.

Given that the images that we obtained are from various sources and in different sizes, as well as the aspect ratio is different, it was necessary to take some appropriate measures to make the images of one context in terms of sizes as well as in terms of aspect ratio to be The training process is normal. As for the images for training the models in the second stage, we deduct the areas of the plates from the images, and then we run on them the same methods that we performed previously, i.e. in terms of size and proportion.

We chose a width of 720 and a height of 470 for the images of the first stage, as for the images for training and testing in the second stage, they are different in terms of height because they belong to three types of plates with different heights, and the width is also chosen to be 720.

We used the (label Img) software to make annotations on the images to identify objects using Ground Truth. In the second stage, we took into account the small width of some numbers and letters, especially the number (1) and the letter (A),Because they are similar in the form in Arabic, as we increased the width of the ground truth for each one for fear that their data will fade through the different layers of the model.

The models have been prepared for training, where the first stage will be the initialization of the model (ssd_resnet50_v1_fpn_640x640_coco17_tpu-8), which we used in this stage for the purpose of training it in order to enable it to perform the task of detecting plates of various types (New-Old-North) through the file (pipeline.config).) and pass the following data to it: num_classes: 3, batch_size: 8, We did an image augmentation technique data_augmentation_options including random_horizontal_flip.

Choosing the training volume in terms of the number of epochs through (total_steps: 25000).

In the second stage we created the three models that we will use as follows: In the model for reading numbers in new plates style, as we mentioned earlier, we used the model (efficientdet_d0_coco17_tpu-32), where we configured it through its config file as follows: num_classes: 43, batch_size: 4, We did not do the data_augmentation_options technique, because numbers, letters, and phrases cannot bear any change in their shape.Choosing the training volume in terms of the number of epochs through (total_steps: 50000).

And in the model for reading numbers in the Od plates style: As we mentioned earlier, we used the (efficientdet_d0_coco17_tpu-32) model, where we configured it through its config file as follows: num_classes: 25, batch_size: 4, We did not do the data_augmentation_options technique, because numbers, letters, and phrases cannot bear any change in their shape. Choosing the training volume in terms of the number of epochs through (total_steps: 50000).

And in the model for reading numbers in the North plates style: As we mentioned earlier, we used the (efficientdet_d0_coco17_tpu-32) model, where we configured it through its config file as follows: num_classes: 13, batch_size: 4, We did not do the data_augmentation_options technique, because numbers, letters, and phrases cannot bear any change in their shape. Choosing the training volume in terms of the number of epochs through (total_steps: 50000).

In general, the mean average precision (mAP) is used to evaluate the accuracy. However, as long as we have two stages, the first is responsible for detecting the license plate, and the second consists of three models.So, at each stage we evaluated the models, the results are shown in Table 1. In Fig.4, an example of the process of detecting and recognizing car vehicles is shown, where we note the excellent discrimination accuracy in the process.

All the experiments have been executed using Ubuntu OS with Intel® Core™ i7-8750H CPU @ 2.20GHz and with NVIDIA GeForce GTX 1060.

Table 1 Evaluation Table

Stage	Model	Training epochs	mAP
First	LPD	25000	99.7%
Second	New Style LPR	10000	99.7%
Second	Old Style LPR	10000	100%
Second	North Style LPR	10000	99.7%

Figure 4 Detection example of Iraqi License Plates

V. CONCLUSION AND FUTURE WORK

Many studies have been conducted on the task of detecting and recognizing vehicle plates, and each country has its style in terms of size, color, and pattern of information contained in vehicle plates. This work is focused on the Iraqi license plate. Due to the great diversity in the style of its vehicle plates in terms of size and color, which made our task was a great challenge in terms of collecting data and obtaining reliable accuracy for the system to work effectively. The proposed system consists of two parts: License Plate Detection (LPD) and License Plate Recognition (LPR), which is used the SSD model with ResNet-50-FPN and three models of efficientdet-d0 respectively. The proposed system achieved promising results obtained when applied to various images and in all environmental conditions. The future work will be to implement the system in real-time and try to adapt the system to work at an acceptable speed. Our suggestion will be to use the work environment Yolo version 5 to be suitable for this purpose.

REFERENCES

[1] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[2] R. H. Finjan, A. S. Rasheed, A. A. Hashim, and M. Murtdha, “Arabic handwritten digits recognition based on convolutional neural networks with resnet-34 model,” *Indones. J. Electr. Eng. Comput. Sci.*, vol. 21, no. 1, pp. 174–178, 2021, doi: 10.11591/ijeecs.v21.i1.pp174-178.

[3] Z. Q. Zhao, P. Zheng, S. T. Xu, and X. Wu, “Object Detection with Deep Learning: A Review,” *IEEE Trans. Neural Networks Learn. Syst.*, vol. 30, no. 11, pp. 3212–3232, 2019, doi: 10.1109/TNNLS.2018.2876865.

[4] S. Minaee, Y. Y. Boykov, F. Porikli, A. J. Plaza, N. Kehtarnavaz, and D. Terzopoulos, “Image Segmentation Using Deep Learning: A Survey,” *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1–22, 2021, doi: 10.1109/TPAMI.2021.3059968.

[5] P. R. Gomes, H. H. Sabuj, M. A. Uddin, M. T. Reza, R. I. Faiz, and M. A. Alam, “A Deep Learning Approach for Reconstruction of Color Images in Different Lighting Conditions Based on Autoencoder Technique,” in *2021 International Conference on Electronics, Information, and Communication (ICEIC)*, 2021, pp. 1–4.

[6] A. Osipov *et al.*, “Deep Learning Method for Recognition and Classification of Images from Video Recorders in Difficult Weather Conditions,” *Sustainability*, vol. 14, no. 4, p. 2420, 2022.

[7] G. S. Hsu, A. Ambikapathi, S. L. Chung, and C. P. Su, “Robust license plate detection in the wild,” *2017 14th IEEE Int. Conf. Adv. Video Signal Based Surveillance, AVSS 2017*, no. August, 2017, doi:

10.1109/AVSS.2017.8078493.

[8] Z. Selmi, M. Ben Halima, and A. M. Alimi, "Deep learning system for automatic license plate detection and recognition," in *2017 14th IAPR international conference on document analysis and recognition (ICDAR)*, 2017, vol. 1, pp. 1132–1138.

[9] L. Zou, M. Zhao, Z. Gao, M. Cao, H. Jia, and M. Pei, "License plate detection with shallow and deep CNNs in complex environments," *Complexity*, vol. 2018, 2018.

[10] J. Singh and B. Bhushan, "Real Time Indian License Plate Detection using Deep Neural Networks and Optical Character Recognition using LSTM Tesseract," *Proc. - 2019 Int. Conf. Comput. Commun. Intell. Syst. ICCCIS 2019*, vol. 2019-Janua, pp. 347–352, 2019, doi: 10.1109/ICCCIS48478.2019.8974469.

[11] V. Pandey, J. P. Patra, S. Choubey, and A. Choubey, "Light weighted Convolutional Neural Network for License Plate Recognition Light weighted Convolutional Neural Network for License Plate Recognition," vol. 12, no. 12, pp. 4119–4125, 2021.

[12] Y. Wang, Z. P. Bian, Y. Zhou, and L. P. Chau, "Rethinking and Designing a High-Performing Automatic License Plate Recognition Approach," *IEEE Trans. Intell. Transp. Syst.*, pp. 1–13, 2021, doi: 10.1109/TITS.2021.3087158.

[13] T. Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal Loss for Dense Object Detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 318–327, 2020, doi: 10.1109/TPAMI.2018.2858826.

[14] T. Y. Lin *et al.*, "Microsoft COCO: Common objects in context," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 8693 LNCS, no. PART 5, pp. 740–755, 2014, doi: 10.1007/978-3-319-10602-1_48.

[15] M. Tan, R. Pang, and Q. V Le, "Efficientdet: Scalable and efficient object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 10781–10790.