

## Fused faster RCNNs for efficient detection of the license plates

Naaman Omar<sup>1</sup>, Adnan Mohsin Abdulazeez<sup>2</sup>, Abdulkadir Sengur<sup>3</sup>, Salim Ganim Saeed Al-Ali<sup>4</sup>

<sup>1</sup>Department of Information Technology, Duhok Polytechnic University, Iraq

<sup>2</sup>Presidency of Duhok Polytechnic University, Duhok Polytechnic University, Iraq

<sup>3</sup>Department of Electrical-Electronics Engineering, Firat University, Turkey

<sup>4</sup>Department of Information Technology Management, Duhok Polytechnic University, Iraq

---

### Article Info

#### Article history:

Received Jan 2, 2020

Revised Feb 29, 2020

Accepted Mar 13, 2020

---

#### Keywords:

Deep learning

Faster-RCNN

License plate detection

Vehicle images

---

### ABSTRACT

Automatic License Plate Detection and Recognition (ALPD-R) is an important and challenging application for traffic surveillance, traffic safety, security, services purposes and parking management. Generally, traditional image processing routines have been used in ALPD-R. Although the general approaches perform well on ALPD-R, new and efficient approaches are needed to improve the detection accuracies. Thus, in this paper, a new approach, which is based on fusing of multiple Faster Regions with Convolutional Neural Network (Faster- RCNN) architectures, is proposed. More specially, the Deep Learning (DL) is used to detect license plates in given images. The proposed license plate detection method uses three Faster-RCNN modules where each faster RCNN module uses a pre-trained CNN model namely AlexNet, VGG16 and VGG19. Each Faster-RCNN module is trained independently and their results are fused in fusing layer. Fusing layer use average operator on the X and Y coordinates of the outputs of the Faster-RCNN modules and maximum operator is employed on the width and height outputs of the Faster-RCNN modules. A publicly available dataset is used in experiments. The accuracy is used as a performance indicator of the proposed method. For 100 testing images, the proposed method detects the exact location of license plates for 97 images. The accuracy of the proposed method is 97%.

Copyright © 2020 Institute of Advanced Engineering and Science.  
All rights reserved.

---

### Corresponding Author:

Naaman Omar,  
Department of Information Technology,  
Duhok Polytechnic University,  
61 Zakho Road, Mazi Qr '1006, Duhok 42001, Iraq.  
Email: ijeecs.iaes@gmail.com

---

## 1. INTRODUCTION

Deep Learning (DL) concerns with processing information utilizing deep networks. It is a part of machine learning approaches. In its earlier appearance in 1943, DL was termed by McCulloch and Pitts as "cybernetics". DL is superior in drawing out-comes from complex problems. Convolutional neural network (CNN) is an instance of deep learning strategy is mimicking brain function in processing in-formation. Recurrent Neural Network (RNN), Deep Neural Network (DNN), and Convolutional Neural Network (CNN) are architecture types of DL[1,2].

Automatic License Plate Recognition (ALPR) is an important topic in intelligent transportation applications. ALPR can be used to identify the vehicles from its license plate. ALPR has potential to be used in variety of application areas such as traffic surveillance, traffic safety, security, services purposes and parking management [3-6]. Lighting and weather conditions effect ALPR system significantly [7,8]. An ALPR system generally composed of four main blocks such as image capturing block, license plate detection block, character segmentation block and character recognition block, respectively [9-12]. The license plate detection, which is in the second block of the ALPD, can be seen as the most important

part. Because, it effects the recognition performance of subsequent parts of the ALPD [13-16]. ALP systems are in the demand and in the last two decades, so many works have been carried out on it. In this work, as we concentrated on the license plate detection, the literature that are related with the license plate detection is reviewed.

Because of increasing the vehicles number, the traffic violation and traffic control become a serious and challenging problem in all over the world. Moreover, the vehicle detection and recognition become difficult from vehicle LP. Various dataset used for training evaluation algorithms [17,18]. In last decade, different algorithms and methods have been proposed for detection and recognition vehicle LP to date, such as VGG16, VGG19, CNN [19].

Ho et al. [20] proposed a license plate detection method for online applications. The proposed method was composed of two stages. Authors used Adaboost method, support vector machine (SVM) classifier and scale invariant feature transform (SIFT) descriptors for ALPD. In the first stage the Adaboost method was used to localize the license plate region. The SIFT features were extracted from the characters of the license plate region and SVM classifier was used to recognize the license plate. Authors used a dataset involves 800 images in their experiments. The reported accuracy was 88%. Katata et al. proposed a method to detect license plate of a given image based on Gabor filters and neural networks (NNs) [21]. The proposed method was composed of three steps. The first step covers the generation of the feature vectors for both training and testing images. Authors used contrast limited adaptive histogram equalization (CLAHE) before Feature Extraction (FE) for improving the quality of the input images. Gabor filters were used feature extraction and NNs was used for classification. 58 Tunisian vehicles images were used in experiments and acceptable results were reported by the authors.

Kim et al. presented a method for license plate detection through two stages [16]. In the first stage, the region of vehicle was located in whole image by using CNN algorithm to specify the region of interest (ROI) easily. In the second stage, the detection of the license plate candidates from vehicle region was accomplished by using the hierarchical sampling method. The false positive license plate regions were eliminated by using deep CNN. The performance of the method was evaluated on Caltech dataset. The obtained precision and recall scores were 98.39% and 96.83%, respectively. Yuan et al. proposed a method to detect vehicle license plate [4]. In the proposed approach line density filter was used to find the candidate regions. Then, the positive (true) license plate was identified based on linear SVMs. For performance evaluation, the Caltech license plate dataset was used. Authors also used another dataset that contains 3828 images. The authors reported 96.62% average accuracy score.

Zhao et al. proposed a method which was composed of the Haar-like cascade classifier and Adaboost for vehicle license plate detection [13]. The dataset, which was used in experiments, was collected from different environments in China. The accuracy of the proposed method was 89.5%. Masood et al. proposed an algorithm for LPDR based on deep Convolutional Neural Networks (CNNs) [22]. The algorithm was applied under various weather conditions and license plate shapes. The proposed algorithm performed detection, character segmentation and recognition. Two different datasets have been used to evaluate the performance, 328 images from USA and 550 images from European. The performance was 99.09% and 99.64% for USA and European datasets, respectively.

Azam et al. proposed an algorithm to detected vehicle license plate region in different hazardous image conditions [10]. Various pre-processing steps were applied on input image for noise removal and contrast enhancement. Authors [10] used Radon transform and tilt correction for detection of the license plate. A dataset that contains 850 vehicle images for different conditions were used in experiments and the obtained accuracy score was 94%. Naimi et al. proposed an algorithm for license plate detection for various nation and multi criterions plate in different condition [11]. The proposed work was based on deep learning where self-taught features were used for detection of the license plate. The authors combined region proposal network with CNN to improve the performance. Firstly, features were extracted by CNN and presented the feature map to the RPN. Secondly, soft-max classifier was used to detect the LP region. 5000 color images were used in experiments and 99% accuracy score was obtained.

Li et al. proposed an algorithm for vehicle license plate detection and recognition using deep CNN and LSTMs [7]. In the proposed approach, two CNN classifiers were used, one was used for detection of the characters from images, and the second one was used to remove false positive. Caltech cars dataset and AOLP dataset were used for detection performance. The achieved performances were 97.56% precision and 95.24% recall. Lalimi et al. presented an automatic algorithm for vehicle license plate detection [23]. The input images were enhanced by using edge density and intensity variance. Vertical edge detection, morphological filtering and geometrical features were used to obtain the correct location of license plate. The dataset used in this paper consists of 425 collected images. The access rate achieved for VLPD in this algorithm is 94.12%.

In this paper, a novel approach, which is based on DL, is proposed for detection of license plates in given images. The proposed method does not use any pre-processing for improving the quality of the input images. More specifically, the proposed approach uses Faster RCNN for detection of the license plates [15]. Three Faster RCNN modules are used where each of them uses a pre-trained CNN model such as AlexNet, VGG16 and VGG19, respectively. Each faster RCNN produces a rectangle that is defined by four parameters such as X and Y coordinates of the upper corner of the rectangle and width (W) and height (H) of the rectangle. Thus, three rectangles are produced by three faster RCNN models. A fusion layer is used after faster RCNN models. The fusion is handled by using average operator on column wise for X and Y coordinates from all faster RCNNs and maximum operators is used on column wise for W and H values that obtained from all faster RCNNs. A dataset, that contains 502 images, is used in our experiments. The images were collected from various environments and various conditions for making the dataset more realistic. These all images are used for training of the faster RCNN models and the flipped and rotated versions of the images are used in testing of the proposed method. The obtained results show that the proposed method is quite successful in detection of the license plates.

a) The original contribution of this work is as following.

The pre-trained CNN models based faster RCNN models are generally used in various object detection applications. However, a fusion operation over multiple faster RCNN models has not been applied on license plate detection until so far. The fusion operation is justified both mathematically and experimentally. While single faster RCNN models produce low detection rates, the fused model highly improves the experimental results.

The rest of the paper is organized as follow. Next section briefly introduces the theory of the faster RCNN. Section 3 describes the proposed method in detail. Experimental works and results are given in Section 4. The paper is concluded in Section 5.

## 2. PROPOSED METHOD

The proposed license plate detection system uses multiple faster regional convolutional neural networks (Faster RCNN) for efficient detection of the license plate region on a given image. Figure 1 shows the illustration of the proposed method.

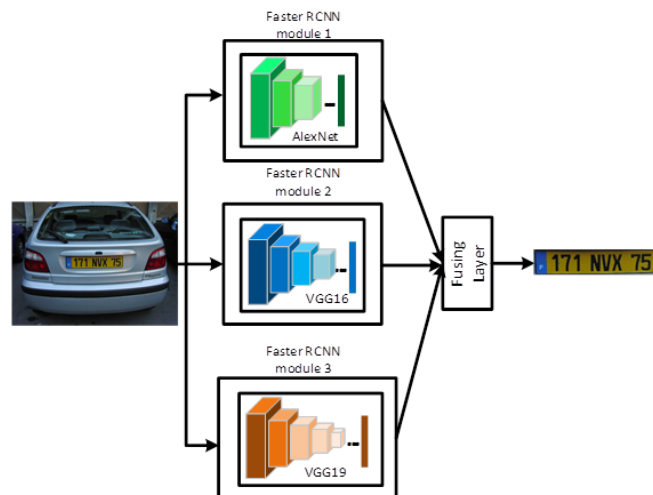


Figure 1. The illustration of the proposed method

The proposed method does not use any pre-processing stage for improving the quality of the input image. Because, the faster RCNN models are quite robust against the noise and other effects. This also saves the computation time of the proposed method. Three Faster RCNN modules are used where each of them uses a pre-trained CNN model. As seen in Figure 1, the fusion layer is in the heart of the proposed method. It fuses the results that are fed from all the faster RCNN modules. Each Faster RCNN module produces a rectangular region (X, Y coordinates and the width and height of the rectangle region) as the location of the license plate. This region might not be located on the exact location of the license plate so fusion layer is used to fix this problem.

Figure 2 illustrates the VGG19, VGG16 and AlexNet architecture. The first Faster RCNN model uses the well-known AlexNet model. AlexNet model architecture contains 8 layers. 5 layers of the AlexNet model is named as convolutional layers and 3 layers are called fully connected layer. Add to that, it contains three pooling layers and five ReLU layers come after convolutional layers.

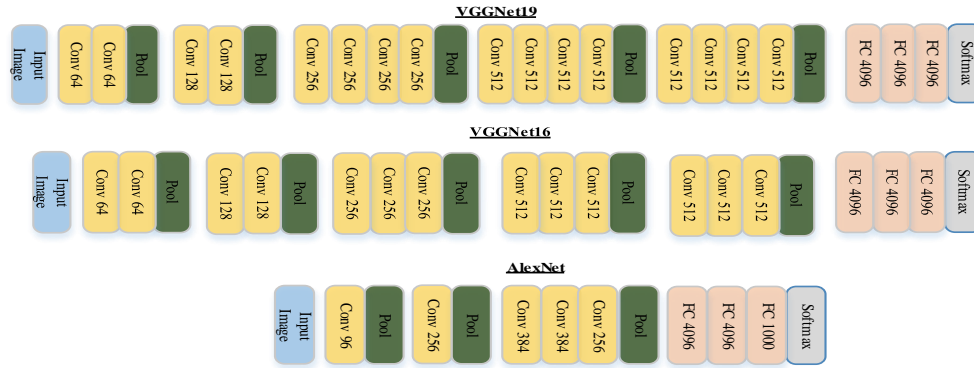


Figure 2. The structures of the AlexNet, VGG16 and VGG19 architectures

VGG16 is another pre-trained deep model which has 13 convolutional layers and 3 fully connected layers. It works very well and has good performance with large datasets, because it uses 3×3 small convolution filters in all layers. The VGG19 model is constructed as a deeper version of the VGG16 model for more performance and better output. It contains 16 convolutional layers and 3 fully connected layers. An illustrative example of the proposed method is given in Figure 3.



Figure 3. The effect of the fusing layer, (a) The result of Faster RCNN module 1, (b) The result of Faster RCNN module 2, (c) The result of Faster RCNN module 3, (d) The result of fusing layer

Figure 3 (a) shows the output of the first faster RCNN module. The license plate region is indicated with a yellow bounding box. As seen, the first faster RCNN module could not obtain the exact location of the license plate region. One character is not covered by the yellow bounding box. Moreover, as given in Figure 3 (b), the second faster RCNN module can detect only a small region of the license plate region as shown by the red bounding box. The most successful detection is handled by the third faster RCNN module as shown in the Figure 3 (c). Almost all license plate region is covered as indicated by the green bounding box. Finally, the Figure 3 (d) shows the fused results where the final region is indicated by a magenta bounding box. The fused result is announced as the output of the proposed method. The fusing operation is given as following;

$$Bounding\_box_{AlexNet} = [X_{AlexNet} \ Y_{AlexNet} \ W_{AlexNet} \ H_{AlexNet}] \tag{1}$$

$$Bounding\_box_{VGG16} = [X_{VGG16} \ Y_{VGG16} \ W_{VGG16} \ H_{VGG16}] \tag{2}$$

$$Bounding\_box_{VGG19} = [X_{VGG19} \ Y_{VGG19} \ W_{VGG19} \ H_{VGG19}] \tag{3}$$

$$Fused\_Bounding\_box = \left[ average \begin{bmatrix} X_{AlexNet} & Y_{AlexNet} \\ X_{VGG16} & Y_{VGG16} \\ X_{VGG19} & Y_{VGG19} \end{bmatrix} \max \begin{bmatrix} W_{AlexNet} & H_{AlexNet} \\ W_{VGG16} & H_{VGG16} \\ W_{VGG19} & H_{VGG19} \end{bmatrix} \right] \tag{4}$$

where  $X$  and  $Y$  show the upper corner coordinate of the bounding box and  $W$  and  $H$  show the width and height of the bounding box, respectively. As given in (1)-(3), the bounding boxes from all Faster RCNN are saved and the fused bounding box is defined in (4). The average operator is used on column wise for  $X$  and  $Y$  coordinates and the maximum operator is used on column wise for  $W$  and  $H$  values. This fusing procedure ensures the optimum location of the license plate.

It is worth of mentioning that the proposed method has training and testing phases. Initially, a training image set is used to train the all faster RCNN modules. After training of the all faster RCNN modules, the testing phase can be applied on the testing images.

### 3. RESEARCH METHOD

#### 3.1. Faster region with convolutional neural network

Explaining The faster RCNN architecture was proposed to reduce the running time of the object detection in a given image. The Regional Proposal Network (RPN) is in the heart of the Faster RCNN structure [24, 25]. RPN is composed of four layers such as input layer, region proposal layer, feature extraction layer and classification layer, respectively. The CNN architecture is used in feature extraction layer. In Faster RCNN, the trained is carried out with multiple of the proposed regions instead of a single region, which makes the training procedure faster than the previous architecture. In faster RCNN, the RPN's and fast RCNN's convolutional layers are combined in one network. Fast RCNN has several advantages, such as high detection quality, using multi task loss in training stage, all layers in network can be update by training, features extraction no need storage disk and faster speed [26, 27]. Thus, the computational time of proposed network is reduced.

#### 3.2. Region proposal network (RPN)

Region proposal network (RPN) aims to generate potential regions and it employs a network to determine if the potential regions contain any objects [24, 25]. The region proposals are generated by the selective search algorithm. The produced regions are ranked by the RPN and the ones most likely containing objects are selected. The region proposal boxes are called as anchors. Anchors are important in Faster R-CNN architecture. Generally, there are 9 anchors in Faster RCNN architecture at a position of an image. The anchors are then examined by a classifier to check the probability of objects. In other words, RPN predicts the possibility of an anchor being background or foreground, and refine the anchor.

The training of the Faster RCNN is carried out with a training dataset and the corresponding ground-truth boxes. In other words, the ground-truth boxes are used to label the anchors. The basic idea in Faster RCNN is that the anchors having the higher overlaps with ground-truth boxes are labelled as foreground and the ones with lower overlaps are labelled as background. We trained three networks AlexNet, VGG16 and VGG19 separately, used baza\_slika, which is a publicly available dataset [28, 29].

### 4. EXPERIMENTAL WORKS AND RESULTS

The experiments were conducted on MATLAB with a computer having an Intel Xeon E5-1650 v4 CPU and 64 GB memory. A publicly available dataset was used in experiments that contains 502 images [29]. The vehicle images were collected under various environmental conditions such as cloudy day, rainy day and night lighting. The images cover different types of vehicle such as cars, trucks, buses and mini buses. All 502 images were used in training of the proposed approach and for testing procedure, randomly selected 100 images were flipped and rotated in 5 and 10 degrees. While Figure 4 shows some training sample images, Figure 5 shows some test sample images.



Figure 4. Some sample images that were used in training of the proposed method



Figure 5. Some sample images that were used in test of the proposed method

As it was mentioned earlier, three pre-trained CNN models were used in Faster RCNN modules. The training parameters of the faster RCNN modules set as following. The stochastic gradient descent with momentum optimizer was utilized in the training process. Maximum epoch, mini-batch size, and initial learning rate were set to 10, 1, and 0.001, respectively. In addition, the positive and negative overlap ranges were scaled to the [0 - 0.3] and [0.6 - 1] ranges, respectively. The number of region proposals to randomly sample from each training image was selected as [256 128]. Box pyramid scale, which was named as anchor box pyramid scale factor, is also 1.2.

Table 1. The training procedure of the faster RCNN with VGG16 model

Epoch	Iteration	Time Elapsed	Mini-batch Loss	Mini-batch Accuracy	Mini-batch RMSE	Base Learning Rate
10	4500	00:22:53	0.0131	100.00%	0.37	0.0010
10	4550	00:23:08	0.0219	100.00%	0.54	0.0010
10	4600	00:23:22	0.0468	99.22%	0.76	0.0010
10	4650	00:23:36	0.0138	100.00%	0.36	0.0010
10	4700	00:23:50	0.0233	100.00%	0.39	0.0010
10	4750	00:24:04	0.0200	100.00%	0.36	0.0010
10	4800	00:24:18	0.0172	100.00%	0.39	0.0010
10	4850	00:24:32	0.0305	99.61%	0.47	0.0010
10	4900	00:24:46	0.0270	100.00%	0.44	0.0010
10	4950	00:25:00	0.0153	100.00%	0.44	0.0010

Table 1 shows the training iterations of the Faster RCNN. The VGG16 model was used in Faster RCNN structure. The columns of the Table 1 show epoch, iteration, time elapsed, Mini-batch loss, Mini-batch accuracy, Mini-batch RMSE and base learning rate, respectively. As seen in Table 1, the training procedure reached the maximum accuracy (100%) at 10 epoch and 4950 iterations. The mini-batch RMSE value was 0.44. The learning rate was not changed during the iterations. The obtained test results are given in Figures. 6, 7, 8 and 9, respectively. While Figure 6 shows the AlexNet based faster-RCNN’s results, Figures 7, 8 and 9 show the VGG16 based faster RCNN’s results, VGG19 based faster RCNN’s results, and the fused results. The detected license plate regions were indicated with colored rectangles for each method. Figure 6 shows the test results for pre-trained AlexNet model based faster-RCNN. As seen in given sample images, the “yellow” rectangles were used to locate the license plate regions. While all car’s license plates were detected correctly, the license plates for trucks were not detected.



Figure 6. Shows the testing results of the AlexNet model

Figure 7 shows the detection results with “red” rectangles that were obtained with VGG16 based faster RCNN. As seen in Figure 7, all vehicle’s license plate regions were detected correctly by the VGG16 based faster RCNN. Figure 8 also shows the detection results with “green” rectangles for VGG19 based faster RCNN. As seen in Figure 8, except one truck, all other vehicle’s license plate regions were detected correctly by the VGG19 based faster RCNN. Finally, Figure 9 shows the fused results from previous faster RCNN models. The “magenta” color was used for determining the exact location of the license plates for all vehicles. As seen in Figure 9, all vehicles license plate locations were detected correctly.



Figure 7. Shows the testing results of the VGG16 model



Figure 8. Shows the testing results of the VGG19 model



Figure 9. Shows the testing results of the fused model

Table 2 shows the obtained result with individual Faster-RCNN modules and fused result. As seen in Table 2, the AlexNet model produces 74% accuracy score which is the worst among all results. The second worst result 87% is obtained by the VGG19 model. VGG16 model produces 93% accuracy score which is better result than AlexNet and VGG19 models. The Fusing Layer produces the best result which is 97%.

Table 2. Performance comparison of individual Faster-RCNN modules and fusing layer

Pre-trained CNN models	Accuracy (%)
AlexNet	74
VGG16	93
VGG19	87
Fusing Layer	97

## 5. CONCLUSION

This paper proposes a novel method for detection of the license plates in a given vehicle images. The proposed method is based on deep faster RCNN models. Recently, deep learning has become a hot topic in computer vision and image processing communities. We opted to use multiple faster RCNN models where pre-trained deep CNN models are used. Three pre-trained CNN models namely AlexNet, VGG16 and VGG19 were used in faster RCNN modules. The results from all deep models were then fused for obtaining a final license plate region. The experiments on the publicly available vehicle license plate dataset showed that proposed fused results outperformed individual faster RCNN modules. The obtained total accuracy was 97%. In our future works, we are planning to apply other deep models in license plate detection. In addition, license plate recognition will be in our future works.

## REFERENCES

- [1] Zeebaree, D. Q., Haron, H., and Abdulzeez, A. M., "Gene Selection and Classification of Microarray Data Using Convolutional Neural Network," presented at the International Conference on Advanced Science and Engineering (ICOASE) (pp. 145-150). IEEE., 2018.
- [2] O. Ahmed and A. Brifceni, "Gene Expression Classification Based on Deep Learning," presented at the 2019 4th Scientific International Conference Najaf (SICN), 2019, pp. 145–149.
- [3] M. S. Al-Shemarry, Y. Li, and S. Abdulla, "An efficient texture descriptor for the detection of license plates from vehicle images in difficult conditions," *IEEE Trans. Intell. Transp. Syst.*, 2019.
- [4] Y. Yuan, W. Zou, Y. Zhao, X. Wang, X. Hu, and N. Komodakis, "A robust and efficient approach to license plate detection," *IEEE Trans. Image Process.*, vol. 26, no. 3, pp. 1102–1114, 2016.
- [5] B. V. Kakani, D. Gandhi, and S. Jani, "Improved OCR based automatic vehicle number plate recognition using features trained neural network," presented at the 2017 8th International Conference on Computing, Communication and Networking Technologies (ICCCNT), 2017, pp. 1–6.
- [6] B. Singh, M. Kaur, D. Singh, and G. Singh, "Automatic number plate recognition system by character position method," *Int J Comput Vis. Robot.*, vol. 6, no. 1/2, pp. 94–112, 2016.
- [7] H. Li and C. Shen, "Reading car license plates using deep convolutional neural networks and lstms," *ArXiv Prepr. ArXiv160105610*, 2016.



- [8] N. Sulaiman, S. N. H. M. Jalani, M. Mustafa, and K. Hawari, "Development of automatic vehicle plate detection system," presented at the 2013 IEEE 3rd International Conference on System Engineering and Technology, 2013, pp. 130–135.
- [9] W. Wiharto, H. Kusnanto, and H. Herianto, "System Diagnosis of Coronary Heart Disease Using a Combination of Dimensional Reduction and Data Mining Techniques: A Review," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 7, no. 2, pp. 514–523, 2017.
- [10] S. Azam and M. M. Islam, "Automatic license plate detection in hazardous condition," *J. Vis. Commun. Image Represent.*, vol. 36, pp. 172–186, 2016.
- [11] A. Naimi, Y. Kessentini, and M. Hammami, "Multi-nation and multi-norm license plates detection in real traffic surveillance environment using deep learning," presented at the International Conference on Neural Information Processing, 2016, pp. 462–469.
- [12] T. Ajanthan, P. Kamalaruban, and R. Rodrigo, "Automatic number plate recognition in low quality videos," presented at the 2013 IEEE 8th International Conference on Industrial and Information Systems, 2013, pp. 566–571.
- [13] Y. Zhao, J. Gu, C. Liu, S. Han, Y. Gao, and Q. Hu, "License plate location based on haar-like cascade classifiers and edges," presented at the 2010 Second WRI Global Congress on Intelligent Systems, 2010, vol. 3, pp. 102–105.
- [14] D. Zang, Z. Chai, J. Zhang, D. Zhang, and J. Cheng, "Vehicle license plate recognition using visual attention model and deep learning," *J. Electron. Imaging*, vol. 24, no. 3, p. 033001, 2015.
- [15] P. Prabhakar, P. Anupama, and S. Resmi, "Automatic vehicle number plate detection and recognition," presented at the 2014 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT), 2014, pp. 185–190.
- [16] S. Kim, H. Jeon, and H. Koo, "Deep-learning-based license plate detection method using vehicle region extraction," *Electron. Lett.*, vol. 53, no. 15, pp. 1034–1036, 2017.
- [17] N. O. Yaseen, S. G. S. Al-Ali, and A. Sengur, "An Efficient Model for Automatic Number Plate Detection using HOG Feature from New North Iraq Vehicle Images Dataset," presented at the 2019 1st International Informatics and Software Engineering Conference (UBMYK), 2019, pp. 1–6.
- [18] N. O. Yaseen, S. G. S. Al-Ali, and A. Sengur, "Development of New Anpr Dataset for Automatic Number Plate Detection and Recognition in North of Iraq," presented at the 2019 1st International Informatics and Software Engineering Conference (UBMYK), 2019, pp. 1–6.
- [19] N. Omar, A. Sengur, and S. G. S. Al-Ali, "Cascaded Deep Learning-Based Efficient Approach for License Plate Detection and Recognition," *Expert Syst. Appl.*, p. 113280, 2020.
- [20] W. T. Ho, H. W. Lim, and Y. H. Tay, "Two-stage license plate detection using gentle Adaboost and SIFT-SVM," presented at the 2009 First Asian Conference on Intelligent Information and Database Systems, 2009, pp. 109–114.
- [21] S. Ktata, F. Benzarti, and H. Amiri, "License plate localization using Gabor filters and neural networks," *J. Comput. Sci.*, vol. 9, no. 10, p. 1341, 2013.
- [22] S. Z. Masood, G. Shu, A. Dehghan, and E. G. Ortiz, "License plate detection and recognition using deeply learned convolutional neural networks," *ArXiv Prepr. ArXiv170307330*, 2017.
- [23] M. A. Lalimi, S. Ghofrani, and D. McLernon, "A vehicle license plate detection method using region and edge based methods," *Comput. Electr. Eng.*, vol. 39, no. 3, pp. 834–845, 2013.
- [24] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," presented at the Advances in neural information processing systems, 2015, pp. 91–99.
- [25] J. E. Espinosa, S. A. Velastin, and J. W. Branch, "Vehicle detection using alex net and faster R-CNN deep learning models: a comparative study," presented at the International Visual Informatics Conference, 2017, pp. 3–15.
- [26] R. Girshick, "Fast r-cnn," presented at the Proceedings of the IEEE international conference on computer vision, 2015, pp. 1440–1448.
- [27] H. Qassim, A. Verma, and D. Feinzimer, "Compressed residual-VGG16 CNN model for big data places image recognition," presented at the 2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC), 2018, pp. 169–175.
- [28] D. Q. Zeebaree, H. Haron, A. M. Abdulazeez, and D. A. Zebari, "Machine learning and Region Growing for Breast Cancer Segmentation," presented at the 2019 International Conference on Advanced Science and Engineering (ICOASE), 2019, pp. 88–93.
- [29] LPDatabase. Accessed: Jul. 2016. [Online]. Available: [http://www.zemris.fer.hr/projects/LicensePlates/english/baza\\_slika.zip](http://www.zemris.fer.hr/projects/LicensePlates/english/baza_slika.zip).