Soft Computing Journal, vol. 12, no. 1, pp. 41–48, 2023
https://doi.org/10.22052/scj.2024.253175.1163

University of Kashan

Contents lists available at SCJ

Soft Computing Journal

Journal homepage: https://scj.kashanu.ac.ir/

Automatic license plate recognition using improved deep learning ✧

Sara Motamed [1,*], Assistant Professor

[1] *Department of Computer Engineering, Fouman and Shaft Branch, Islamic Azad University, Fouman, Iran.*

ARTICLE INFO.

Article history:
Received June 28, 2023
Accepted September 21, 2023

Keywords:
License plate recognition
Deep learning
Neural networks
Generative adversarial network
Encoding-decoding structure

ABSTRACT

According to today's statistics, more than half a billion vehicles move around the world, making inspection and monitoring one of the basic needs of any traffic control system. All vehicles have an identification number exhibited as the license plate, their primary ID, a vital element. Deep Learning methods are adopted to detect vehicle license plates. This proposed method consists of two steps: highlighting the license plate and reading the ID stages. In this context, the combination of deep neural networks (DNN) and the competitive generative adversarial network (GAN) is applied in the encoding-coder network/structure for this highlighting. The proposed models are assessed based on the FZU Cars and Stanford Cars datasets, to which the results of this study are compared and discussed. The findings here indicate that the accuracy of this proposed model is almost 98%, subject to the two datasets.

1 Introduction

Due to the increase in vehicle count, the control and management of traffic in motion or still anywhere is an essential issue. The indiscriminate increase in vehicle count has led to the installation of traffic control systems, highway tolls, adoption of proper parking management, etc. Like many other contemporary issues, it is obvious that the control of this high volume of vehicles is beyond the sole modern human capabilities without applying computer systems [1]. All vehicles have an identification number as their primary ID. An identification number or license plate is mandatory for every vehicle by which it is identified elements. The vehicle license plate identification system is a mechanized and computerized system where image processing is applied to identify the license plates by reading the license plate IDs' characters through the images taken by surveillance cameras [2, 3]. By applying image processing methods effectively, the accuracy of identification can increase. Considering that the images are taken by outdoor cameras, the lighting conditions at different hours of the day, weather conditions, and pollution on the plate can contribute to having unclear final images that will not serve the purpose. The camera angle to horizon ratio may make the license plate be seen angularly, thus a perspective error [4].

Among the available studies on this issue applying the perceptron neural networks is the most common. The drawback of these networks is their inability to learn complex structures in images with high accuracy. In contrast to conventional neural networks (CNN), referred to as shallow networks, more complex structures are introduced as deep learning. CNNs are applied in different applications like letter identification, objects, etc. Because each image taken by the cameras is not flawless, applying an upgraded method will not be an effective overall method for all images [5]. Another issue is that image enhancement methods often have certain advantages and are time-effective [3–7]. To overcome these drawbacks, it is suggested to apply the high capacity of deep neural networks

✧ Article type: original research

* Corresponding author.

Email address: motamed.sarah@gmail.com (S. Motamed)

How to cite this article: S. Motamed, "Automatic license plate recognition using improved deep learning", *Soft Comput. J.*, vol. 12, no. 1, pp. 41–48, 2023, doi: 10.22052/scj.2024.253175.1163.

in training the license plate identification systems. By training a deep neural network with license plate images, it is possible to directly highlight the vehicle IDs in a short time without adopting complex upgrading methods. The objective here is to apply the Deep Learning Networks for vehicle license plate identification.

This proposed model consists of the two highlighting the license plate and reading the ID stages. To highlight the combination of different deep neural networks and the GAN model in this proposed model, the: encoding, decoding, and feature conversion layers are introduced. The encoder layer takes the binary image of the license plate and then highlights the license plate IDs in the new images. This highlighting is to generate an image of the license plate where the black license plate IDs and other components of the license plate are closer to white as the background from the image. The input here is the license plate image and the target images consist of the binary images of the license plates already tagged by the users.

This paper is structured as follows: the literature is reviewed in Section 2; the model is proposed in Section 3; the experiments are run in Section 4, and the paper is concluded in Section 5.

2 Related work

The chronology of intelligent transportation systems is rooted in studies run on image processing [8–10]. A license plate recognition system is typically implemented in three stages:

(1) Finding distances of the image received from the imaging system, which includes the license plate. At this stage applying features like the density of edges on the place of the plate, and the location where the plate is installed are identified. Following this, the angle of the license plate can be corrected.
(2) The image lighting problems are usually corrected first and the extracted plate image is upgraded by applying different methods like the morphological operators. The enhanced image can be converted into two areas of identifiers and backgrounds through binary methods.
(3) The digits and letters are identified separately. In the process of finding the location of the plate, one of the methods applied in most studies is to find areas with high edge density. To find the location of the license plate, the moving window, where the color space information and extraction of tissue information through wavelet transform is applied must be of concern [11–13].

There exist some studies on learning methods. Training a cascading band by applying characteristics similar to that of [14] and CNN to find the location of plates in recent research [9, 15]. In the recognition process of feed plate, the following two main issues are of concern.

2.1 Upgrading and zoning the IDs of the license plates

Because the plate is separated from the locating stage due to problems such as variable lighting conditions, the presence of shadows on the plate, contamination on the body of the plate, and deviation of the plate from the horizon, does not have the proper quality to read the identifiers and it is necessary to be upgraded. To convert the plate image to a binary image is applied instead of the global threshold [16]. After thresholding, the connected regions are extracted. At this point, some areas with large areas are removed. By aligning the remaining areas in the horizontal direction, the plate deviation is eliminated. The thresholding is done by applying histogram information on lighting intensity and neural network [17]. The endowment thresholding is applied in [18], where a local threshold is calculated for each pixel by applying the average and variance in the pixel neighborhood. After thresholding, and running the analysis of the connected components, the area on which the identifiers are located is extracted as a mask. Because of the considerable difference between the color of the identifiers and the plate background, the plate outlay profile may contain important information about the license plate identifiers. In [19], the two rows on the plate are separated by applying the horizontal outsoaring index. The distance between the two rows on the outline profile appears as a valley, and by finding the location of this valley, the separator line of the two rows will be specified. Identities are separated through the vertical output profile [20]. By applying this vertical profile information, the border of each identifier becomes extracted. A hybrid method to separate identifiers from the background is applied in [21], where, first, the plate image is smoked by applying a comparative threshold method, and next, the remaining excess areas on the image are removed through the thinning algorithm. A method for coding the license plate is presented in [22], where a neural network with an encoder-decoder structure is applied.

2.2 Reading vehicle license plate IDs

After zoning the plate identifiers, they can be detected through the conventional methods of recognizing optical letters. Typically, due to the change in distance between the imaging system and the license plate, the license plate image will have a deviation, which introduces a perspective in the image and makes the IDs separated from the plate that may be observed in different sizes. This issue is dealt with by considering different patterns of identifiers and changing their rotation angles [23]. The skeleton of identifiers obtained through a morphology operator is applied to extract the feature. In this process, the window containing the plate ID skeleton is divided into nine segments where the angle of each is extracted as a feature. An artificial neural network detects the plate IDs through these features. The contour curve of identifiers is applied as a feature free of the shape and size of identifiers [24]. Gabor filter is applied as one of the feature extraction methods for categorizing identifiers.

The brightness intensity vectorization of normalized license plate identifiers is applied as a feature in two categories the nearest neighbor strap and the support vector machine [25]. IDs are identified through the geometric features of identifiers and the distribution of brightness intensity [26]. The gray image of identifiers in the neural network input is applied in training the network [27]. Reading license plate features is a confirmation method for finding the license plate location, where the count of identifiers detected by a neural network serves as feedback for the plate locating method. In all studies, Deep Learning is performed very well in the field of object recognition Researchers in [28], sought to determine the location of the license plate correctly with a new method; by improving the feature recognition algorithm, where the final accuracy of the algorithm increased. Their proposed method consists of three steps: 1) by applying the edge detection operation and special morphology, the correct places of the plates are determined, 2) the license plate image is binaries and the features are separated and 3) the features are identified, thus, the license plate is identification. An open-source technology platform named Tensorflow is applied in [29], where, first, the existing images are processed and converted into grayscale to reduce the image noise level and identify license plates with different colors, next, the license plate letters in that image are extracted through the optical feature identification. The extracted text is stored in an Excel document for further purposes, and finally, the vector technique and the Bayesian rule-three methodology overtake the working framework. A Deep Learning model is applied in [2] to detect license plates in intelligent transportation systems, which increases the speed and accuracy in identifying the plate location in the given images together with the license plate location. The architecture of the proposed model is based on convolutional networks, where a deep convolutional network and LSTM are applied to identify the features related to the license plate. The accuracy of their proposed model for vehicle identification is 100%, license plate identification is 100% and feature identification is 99.37%.

One of the applications where Deep Learning is successful is in the encoding-decoding structures, the coders. A self-encoder network, first, maps the input data into the property space, and next maps this property space into the primary space. The training requirement of this network is the reversibility of data in the decoder output, in the sense that input information is not deleted in the encoder output, and according to the type and idea of the training, it can extract information and features appropriate for other machine learning activities [30].

Among the neural networks, the recursive networks are commonly applied to process a variety of sequences. In this network, unlike most neural networks each learning unit is connected only to its consecutive neurons layer allowing each neuron to be connected to the units in the same layer.

3 Proposed model

Due to the efficiency of these networks, in this study, only three reliable models of Deep Learning are selected to be the control in identifying license plate feeds. The stages of the proposed model are flowcharted in Fig. 1.

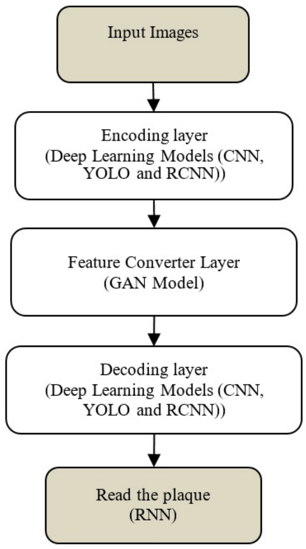

Fig. 1. Stages of the proposed model

This model consists of two steps highlighting the plate ID and reading the identifiers. The prominence of plate identifiers is determined by combining the well-known deep neural CNN, RCNN, and YOLO networks, and the competitive (GAN) with the encoder-decoder structure. A recursive neural network (RNN) is used to read the plaque.

The GAN consists of two generator and discriminator segments, applied to generate new data, to increase the training data count. After noise is added to the competitive GAN its generator segment obtains new images from the training image set which are matched with the training data image through discrimination. Provided that there is no match, the discriminator rejects images, otherwise artificial images are generated. The structure of the GAN is shown in Fig. 2.

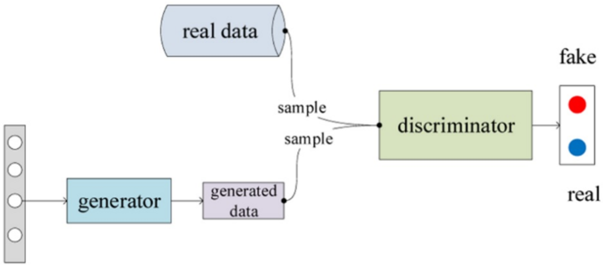

Fig. 2. The structure of GAN

In this method, the network learns how to generate new data from the training data that are statistically identical to the training data and the generated data.

This proposed model can learn the smoked image of the license plates in different conditions and then highlight the IDs in the new images. The purpose of highlighting vehicle IDs is to generate an image of the license plate where its ID is in black and other components as the background fades from the image and becomes white. The input images of this proposed method consist of vehicle license plate images.

The target images for training are binary images of plates that are modified by the user. Because different optical conditions introduce many changes in the color of the license plate, the color feature cannot be applied as an efficient feature. More learning components are required to extract the features from the colored plate networks. An increase in the input neural network volume decreases the computational speed and increases the estimation time of the network. Consequently, to save resources, the input image is first converted into a gray spectrum followed by a scale change of 398 lengths and 80 widths.

3.1 Encoding layer

Plate identifiers are generally white and black. Because the frequency of plates with black identifiers is higher than those of white, the network may go through hardship to learn to identify plates with white identifiers, therefore, plates are inversely painted with white identifiers before training. In this layer, deep neural networks are applied to encode to convert the plate image of the gray spectrum into a binary image of the license plate.

Because this proposed encoder network training process contains a training decoder, this structure is not able to highlight white license plate identifiers. One of the most common solutions is to reverse the plate image, with a focus on image issues. There exist many solutions among which zoning of license plate identifiers, is the most practical. The solution is based on the time and computational cost of adding the middle network after the encoder, where the features related to the initial image inverse are estimated through the information in the encoder. In this case, in the decoder input, both categories of features related to the original image and its inverted version are available, and the decoder can generate the endzoned image in the output without determining the color and reverse the image in the encoder input. The converted properties of the input image obtained in the encoder output into image-related features in this middle network, make this network to be known as the property converter.

3.2 Property transform layer

This proposed decoder structure is not able to generate a proper output image of the plate with white identifiers, so the network structure can be changed in such a manner that the decoder input has the features of both the input image mode and its converted position. Bear in mind that repetition of the feature extraction in the encoder is time-consuming; consequently, it is necessary to adopt a method that includes features related to the inverted image in the encoder output.

The objective of this method is to apply a small network that converts the output properties of the encoder into the properties related to the initial image inversion, named the property converter network. Though small, this network is time-efficient. Because the property conversion network is applied to all input images, there is no need to categorize and separate the plates before the encoder or feature converter input. With this method, different types of plates can be identified. This feature converter can convert the extracted features from the input image into features obtained from the inverted image. To train this network, the features extracted from images with white identifiers are applied as the training data in the input. The target data are the properties extracted in the encoder for the inverse of those images. In the feature transforming layer, a competitive GAN with its two producer and separator segments is applied to generate new data, to increase the training data count. In this method, the network learns how to generate new data from the training data statistically matched in the statistical context of the training and generated data.

3.3 Decoder layer

The idea of applying a feature converter is to have features related to images and their converted position at the same time in the decoder input; therefore, it is necessary that the proposed decoder architecture be designed with the ability to receive both sets of features at the entrance. To highlight the license plate free of identifiers, this decoder acts as a free identifier's color.

3.4 Reading prominent license plate IDs

After highlighting the vehicle's license plate IDs, the next step is to read its IDs. In this paper, the recursive neural network is applied to read vehicle license plate IDs from the image with prominent identifiers. The advantage of this method is that the IDs do not need to be separated from each other to read and train the network. The RNN where the input image is given to it as a sequence can take the decoder output image and return the plate number to the output. Information from each location area is stored on the input image in its learning units' memory. The input image is labeled as a sequence. By applying the last layer of the recursive neural network, the combination and output connection method, a single sequence of plate identifiers is returned.

4 Experiments

4.1 Datasets

Two standard data sets are applied in the tests in this field. The primary dataset is the FZU Cars consists of 297 model vehicles with 43,615 images. The Stanford Cars dataset consists of 196 model vehicles with 16,185 images [1] . Because these datasets do not have separate educational and test sets, all the tests in this article are based on the mutual cross-validation method. In this context, the whole dataset is randomly proportioned into five segments: three as training and two as validation and testing. To evaluate this proposed model, the evaluating precision, recall, and F-score criteria are applied.

4.2 Results

This proposed model consists of two main stages highlighting the license plate and reading its ID. As to highlighting, the combination of neural networks and competitive GAN is applied in the encoder-decoder network. The innovation here is the same work hierarchy, comparing three models of Deep Learning and selecting the best therein. The structure of the proposed model, consists of the encoder, attribute transform, and decoder segments, Fig. 1. The input is the vehicle license plate image, and the target images as the binary images of the license plate already labeled by the user, which enter the next stage. The coder layer takes the binary image and highlights the license plate IDs in new images to generate an image of a license plate, the identifiers of which are in black and other components as the background fades into white. Because the decoder is not able to generate an appropriate image of the plate with white identifiers, the encoder input should be changed in the sense that it has the features of its two input image modes. In the feature transforming layer, a small network is applied that converts the encoder output attributes into the corresponding features of the original image inversion. Moreover, in the feature transforming layer, a competitive GAN with two segments of producer and separator is applied to generate new data to increase the training data count. It is expected that applying the decoder-coder layer together with the feature convertor layer would considerably increase the accuracy of license plate identification systems.

The proposed algorithms are run by Python software to implement the deep convolutional neural network, with 150 filters sized 16, 32, and 64. The nonlinear ReLU function is applied as an activation function in this network. The ADADELTA weight update rule with a 0.01 learning rate and 0.05 dropout rate is applied to train the model. The YOLO algorithm is a one-shot object detection system that uses a CNN to perform object detection [31]. The network takes an input image and divides it into a network of cells, each responsible for predicting the presence of objects in a certain area of the image. The network generates a set of bounding boxes and class probabilities for each cell. In this paper, the input image is resized into 448×448 before passing through the convolution network. A 1 by 1 convolution is applied to reduce the channels' count followed by a 3 by 3 convolution to produce a cubic output. The activation function is ReLU, except for the final layer, where a linear activation function is applied. In the RCNN method, training is a multi-step pipeline consisting of feature extraction, fine-tuning the network with input failure, training SVMs [32, 33], and fitting bounded box regression. In this paper, in the RCNN method, 2000 candidate regions are extracted from the resized images. Each one of these regions is fed into the network independently. The ROI pooling size in this implementation phase is 7×7×512.

The well-known OKM-CNN [6], CNN, YOLO, and RCNN Deep learning models are applied here and the results are compared in Table 1. By assessing the results, it is revealed that the accuracy of the RCNN model with GAN is higher than that of the YOLO and CNN models with GAN and OKM-CNN. According to the available findings, it is observed that the results of the proposed models on the FZU Cars dataset are better than those of the Stanford Cars dataset.

The comparison of the CNN, YOLO, and RCNN with GAN models is shown in Fig. 3, where, as observed the best result is attributed to implementation by combining RCNN and GAN. Fig. 4 demonstrates

[1] See https://www.kaggle.com/datasets/jessicali9530/stanford-cars-dataset and https://campuscommune.tcs.com/en_in/intro/contests/tcs-humain-2019.

Table 1. Results of experiments on tested datasets.

Models	FZU Cars Dataset			Stanford Cars Dataset		
	Precision	Recall	F-Score	Precision	Recall	F-Score
ZF	0.916	0.948	0.932	0.856	0.897	0.872
OKM-CNNVGG16	0.925	0.955	0.940	0.911	0.954	0.907
ResNet50	0.938	0.951	0.944	0.912	0.946	0.918
ResNet101	0.945	0.958	0.951	0.941	0.952	0.909
DA-Net136	0.961	0.964	0.962	0.953	0.962	0.932
DA-Net160	0.965	0.966	0.965	0.949	0.954	0.938
DA-Net168	0.966	0.968	0.967	0.962	0.967	0.945
DA-Net200	0.969	0.971	0.970	0.955	0.959	0.941
OKM-CNN	0.973	0.979	0.972	0.965	0.970	0.948
CNN+GAN	0.984	0.983	0.979	0.972	0.981	0.957
YOLO+GAN	0.985	0.983	0.980	0.973	0.983	0.965
RCNN +GAN	**0.987**	**0.985**	**0.986**	**0.976**	**0.984**	**0.967**

the ROC curve of the RCNN+GAN algorithm. The results show that the true positive rate in RCNN is high, because the RCNN is more proper for working on data with consecutive inputs, and can accept inputs of different lengths. Another advantage of the RCNN is its ability to store past data; consequently, it can be claimed that it obtains better results in automatic license plate identification.

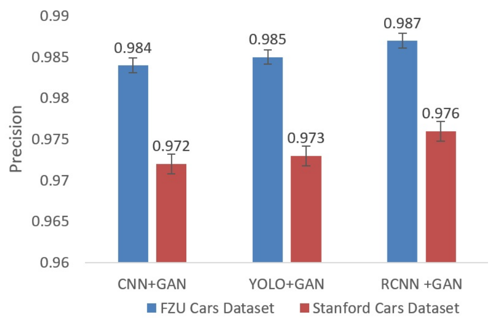

Fig. 3. Comparision of the proposed models.

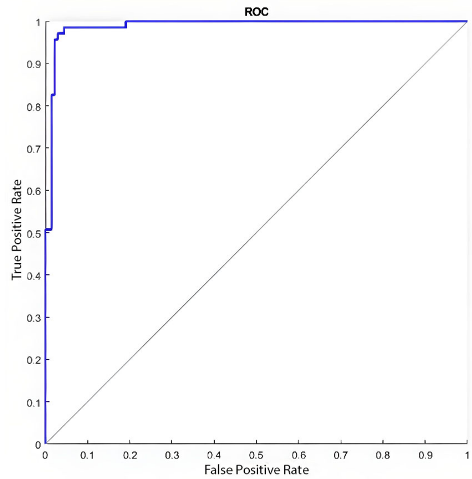

Fig. 4. ROC curve of the RCNN+GAN algorithm.

5 Conclusion

Automatic license plates' identification of vehicles highly contributes to planning and controlling mass traffic. The importance of vehicle license plate identification systems has made researchers develop different models based on neural networks. This proposed model consists of two stages highlighting vehicle license plate and reading its IDs, where the Deep Learning with encoding-coders structure is applied. This method highlights the vehicles' plates' IDs free of their black or white colors. Because the coder is not able to generate a proper image of the plate with white identifiers, the encoder input must be changed in the sense that it has the features of its two input image modes. Consequently, a small network is applied in the feature transform layer which converts the feature of the encoder output to the features related to the inversion of the original image. The proposed models are tested on FZU Cars and Stanford Cars datasets. By assessing the obtained results, it is revealed that the accuracy of the RCNN model with GAN with 98% accuracy outperforms the YOLO and CNN models with GAN. According to the available findings, it is observed that the results of the proposed models on the FZU Cars dataset are better than those of the Stanford Cars dataset.

Conflict of interest

The authors declare that they have no conflict of interest.

References

[1] C.-H. Huang, Y. Sun, and C.-S. Fuh, *Technologies to Advance Automation in Forensic Science*

and Criminal Investigation. IGI Global, Hershey, PA, 2022, ch. Vehicle License Plate Recognition With Deep Learning, pp. 161–219, doi: 10.4018/978-1-7998-8386-9.ch009.

[2] J. Pirgazi, M. M. Pourhashem Kallehbasti, and A. Ghanbari Sorkhi, "An end-to-end deep learning approach for plate recognition in intelligent transportation systems," *Wirel. Commun. Mobile Comput.*, vol. 2022, no. 1, p. 3364921, 2022, doi: 10.1155/2022/3364921.

[3] W. Wang and T. Jiaoyang, "Research on license plate recognition algorithms based on deep learning in complex environment," *IEEE Access*, vol. 8, pp. 91 661–91 675, 2020, doi: 10.1109/ACCESS.2020.2994287.

[4] Y. Zhang, Z. Wang, and J. Zhuang, "Efficient license plate recognition via holistic position attention," in *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*. AAAI Press, 2021, pp. 3438–3446, doi: 10.1609/AAAI.V35I4.16457.

[5] R. Balia, S. Barra, S. Carta, G. Fenu, A. S. Podda, and N. Sansoni, "A deep learning solution for integrated traffic control through automatic license plate recognition," in *Computational Science and Its Applications - ICCSA 2021 - 21st International Conference, Cagliari, Italy, September 13-16, 2021, Proceedings, Part III*, ser. Lecture Notes in Computer Science, O. Gervasi, B. Murgante, S. Misra, C. Garau, I. Blecic, D. Taniar, B. O. Apduhan, A. M. A. C. Rocha, E. Tarantino, and C. M. Torre, Eds., vol. 12951. Springer, 2021, pp. 211–226, doi: 10.1007/978-3-030-86970-0_16.

[6] T. Vaiyapuri, S. Mohanty, M. Sivaram, I. Pustokhina, D. Pustokhin, and K. Shankar, "Automatic vehicle license plate recognition using optimal deep learning model," *Comput. Mater. Continua.*, vol. 67, pp. 1881–1897, 12 2020, doi: 10.32604/cmc.2021.014924.

[7] V. Gnanaprakash, N. Kanthimathi, and N. Saranya, "Automatic number plate recognition using deep learning," *IOP Conf. Ser.: Mater. Sci. Eng.*, vol. 1084, no. 1, p. 012027, 2021, doi: 10.1088/1757-899X/1084/1/012027.

[8] T. Toroyan, K. Iaych, and M. Peden, "Global status report on road safety," World Health Organization, Tech. Rep., 2015.

[9] N. N. Kyaw, G. R. Sinha, and K. L. Mon, "License plate recognition of myanmar vehicle number plates A critical review," in *IEEE 7th Global Conference on Consumer Electronics, GCCE 2018, Nara, Japan, October 9-12, 2018*. IEEE, 2018, pp. 771–774, doi: 10.1109/GCCE.2018.8574751.

[10] A. Kashyap, B. Suresh, A. Patil, S. Sharma, and A. Jaiswal, "Automatic number plate recognition," in *2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN)*, 2018, pp. 838–843, doi: 10.1109/ICACCCN.2018.8748287.

[11] K. Deb, M. I. Khan, M. R. Alam, and K.-H. Jo, "Optical recognition of vehicle license plates," in *Proceedings of 2011 6th International Forum on Strategic Technology*, vol. 2, 2011, pp. 743–748, doi: 10.1109/IFOST.2011.6021129.

[12] F. Wang, L. Man, B. Wang, Y. Xiao, W. Pan, and X. Lu, "Fuzzy-based algorithm for color recognition of license plates," *Pattern Recognit. Lett.*, vol. 29, no. 7, pp. 1007–1020, 2008, doi: 10.1016/J.PATREC.2008.01.026.

[13] S. Yu, B. Li, Q. Zhang, C. Liu, and M. Q. Meng, "A novel license plate location method based on wavelet transform and EMD analysis," *Pattern Recognit.*, vol. 48, no. 1, pp. 114–125, 2015, doi: 10.1016/J.PATCOG.2014.07.027.

[14] E. Rashedi and H. Nezamabadi-pour, "A hierarchical algorithm for vehicle license plate localization," *Multim. Tools Appl.*, vol. 77, no. 2, pp. 2771–2790, 2018, doi: 10.1007/S11042-017-4429-Z.

[15] O. Ibitoye, T. Ejidokun, O. Dada, and O. Omitola, "Convolutional neural network-based license plate recognition techniques: A short overview," in *2020 International Conference on Computational Science and Computational Intelligence (CSCI)*, 2020, pp. 1529–1532, doi: 10.1109/CSCI51800.2020.00283.

[16] Y. Nakagawa and A. Rosenfeld, "Some experiments on variable thresholding," *Pattern Recognit.*, vol. 11, no. 3, pp. 191–204, 1979, doi: 10.1016/0031-3203(79)90006-2.

[17] T. Nukano, M. Fukumi, and M. Khalid, "Vehicle license plate character recognition by neural networks," in *Proceedings of 2004 International Symposium on Intelligent Signal Processing and Communication Systems, 2004. ISPACS 2004.*, 2004, pp. 771–775, doi: 10.1109/ISPACS.2004.1439164.

[18] I. Giannoukos, C. Anagnostopoulos, V. Loumos, and E. Kayafas, "Operator context scanning to support high segmentation rates for real time license plate recognition," *Pattern Recognit.*, vol. 43, no. 11, pp. 3866–3878, 2010, doi: 10.1016/J.PATCOG.2010.06.008.

[19] T. Duan, D. Tran, P. Tran, and N. Hoang, "Building an automatic vehicle license-plate recognition system," in *Proc. Int. Conf. Comput. Sci. RIVF*, 02 2005.

[20] B. Shan, "Vehicle license plate recognition based

on text-line construction and multilevel RBF neural network," *J. Comput.*, vol. 6, no. 2, pp. 246–253, 2011, doi: 10.4304/JCP.6.2.246-253.

[21] S. Nomura, K. Yamanaka, O. Katai, H. Kawakami, and T. Shiose, "A novel adaptive morphological approach for degraded character image segmentation," *Pattern Recognit.*, vol. 38, no. 11, pp. 1961–1975, 2005, doi: 10.1016/J.PATCOG.2005.01.026.

[22] A. Kendall, V. Badrinarayanan, and R. Cipolla, "Bayesian segnet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding," in *British Machine Vision Conference 2017, BMVC 2017, London, UK, September 4-7, 2017*. BMVA Press, 2017.

[23] T. Naito, T. Tsukada, K. Yamada, K. Kozuka, and S. Yamamoto, "Robust license-plate recognition method for passing vehicles under outside environment," *IEEE Trans. Veh. Technol.*, vol. 49, no. 6, pp. 2309–2319, 2000, doi: 10.1109/25.901900.

[24] M. Kim and Y. Kwon, "Multi-font and multi-size character recognition based on the sampling and quantization of an unwrapped contour," in *13th International Conference on Pattern Recognition, ICPR 1996, Vienna, Austria, 25-19 August, 1996*. IEEE Computer Society, 1996, pp. 170–174, doi: 10.1109/ICPR.1996.546816.

[25] P. Hu, Y. Zhao, Z. Yang, and J. Wang, "Recognition of gray character using gabor filters," in *Proceedings of the Fifth International Conference on Information Fusion. FUSION 2002. (IEEE Cat.No.02EX5997)*, vol. 1, 2002, pp. 419–424, doi: 10.1109/ICIF.2002.1021184.

[26] M. Ko and Y. Kim, "A simple OCR method from strong perspective view," in *33rd Applied Image Pattern Recognition Workshop (AIPR 2004), Emerging Technologies and Applications for Imagery Pattern Recognition, 13-15 October 2004, Washington, DC, USA, Proceedings*. IEEE Computer Society, 2004, pp. 235–240, doi: 10.1109/AIPR.2004.8.

[27] J. Jiao, Q. Ye, and Q. Huang, "A configurable method for multi-style license plate recognition," *Pattern Recognit.*, vol. 42, no. 3, pp. 358–369, 2009, doi: 10.1016/J.PATCOG.2008.08.016.

[28] S. Fadaei, A. Kavari, A. Dehghani, and K. Rahimizadeh, "Automatic license plate recognition system based on morphological operations and vertical histogram image processing," *Soft Comput. J.*, vol. 10, no. 2, pp. 84–97, 2022, doi: 10.22052/scj.2022.243224.1014 [In Persian].

[29] N. Girinath, C. G. Babu, B. Vidhya, S. Agathiyan, P. A. Nandha, S. A. Rigin, and S. A. Pandian, "Automatic number plate detection using deep learning," in *2022 Smart Technologies, Communication and Robotics (STCR)*, 2022, pp. 1–5, doi: 10.1109/STCR55312.2022.10009582.

[30] P. Vincent, H. Larochelle, Y. Bengio, and P. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Machine Learning, Proceedings of the Twenty-Fifth International Conference (ICML 2008), Helsinki, Finland, June 5-9, 2008*, ser. ACM International Conference Proceeding Series, W. W. Cohen, A. McCallum, and S. T. Roweis, Eds., vol. 307. ACM, 2008, pp. 1096–1103, doi: 10.1145/1390156.1390294.

[31] M.-R. Feizi-Derakhshi, Z. Mottaghinia, and M. Asgari-Chenaghlu, "Persian text classification based on deep neural networks," *Soft Comput. J.*, vol. 11, no. 1, pp. 120–139, 2022, doi: 10.22052/scj.2023.243182.1010 [In Persian].

[32] S. Najafi and S. Noferesti, "Determining dynamic time quantum in round-robin scheduling algorithm using machine learning," *Soft Comput. J.*, vol. 10, no. 2, pp. 32–43, 2022, doi: 10.22052/scj.2022.243181.1002 [In Persian].

[33] M. Ebtia, S. M. Hoseini, and R. Khochiani, "Credit rating of bank customers using a new ensemble method based on support vector machine: a case study of pasargad bank," *Soft Comput. J.*, vol. 10, no. 2, pp. 2–15, 2022, doi: 10.22052/scj.2022.243227.1016 [In Persian].