

# License Plates Detection and Recognition with Multi-Exposure Images

Seong-O Shim<sup>1\*</sup>

Department of Computer & Network  
Engineering, College of Computer  
Science and Engineering  
University of Jeddah  
Jeddah, Saudi Arabia

Romil Imtiaz<sup>2</sup>, Asif Siddiq<sup>3</sup>

Electrical Engineering Department  
Pakistan Institute of Engineering &  
Technology  
Multan, Pakistan

Ishtiaq Rasool Khan<sup>4</sup>

Department of Computer Science  
and AI, College of Computer  
Science and Engineering  
University of Jeddah  
Jeddah, Saudi Arabia

**Abstract**—Automatic License Plate Recognition (ALPR) has been an important research topic for many years in the intelligent transportation system and image recognition fields. License Plate (LP) detection and recognition has always been a challenging issue due to several factors, including different weather and lighting, unavoidable data acquisition noise, and requirement for real-time performance in state-of-the-art Intelligent Transportation Systems (ITS) applications. Different techniques have been proposed based on machine learning, deep learning, and image processing for the detection and recognition of LPs. This paper proposes a method that performs vehicle LP detection and character recognition with high accuracy by using artificially generated multi-exposure images of the LP. First, one under-exposed and three over-exposed images are generated from a reference image taken from the camera. Then, LP detection and character recognition algorithms are applied on these five images – one real image and four synthesized images. At each character location in LP, the character detected with the highest confidence level among these images is selected as the final predicted character. The system is fully automated, and no pre-processing, calibration, or configuration procedures are needed. Experimental results show that the proposed method achieves high accuracy and works robustly even in challenging conditions. The proposed method can be used in any existing ALPR system and the results on three recent ALPR techniques show that the accuracies are further improved when they are combined with the proposed method.

**Keywords**—License plate detection; character recognition; multi-exposure images; convolutional neural networks

## I. INTRODUCTION

Automatic License Plate Recognition (ALPR) has become a popular topic of study and development in recent years. In modern urban societies, there is a constant demand for powerful intelligent systems for transport security and monitoring, as well as to assist in automating toll collection [1], traffic law enforcement [2], border control [3], private space access control [4], and road traffic monitoring [5]. Intelligent Transportation Systems (ITS) became a necessity in the quest to turn cities into smart cities. ALPR is an important part of ITS because it includes complex computer vision operations of object segmentation, and recognition [6, 7, 8]. As a result, there is a need to make the license plate (LP) recognition systems robust in a wide variety of backdrops and other environmental elements such as illumination, camera angle, and noise and

distortion level in images, which make ALPR a challenging task. To recognize car licenses plates in restricted backdrops, a variety of advanced computer vision technologies and artificial intelligence algorithms have been developed [5, 7, 9], which may not work very well in dynamic conditions. In addition, it is impossible to cover entire range of roads, highways, or motorways with stationary/fixd ALPR cameras on ground, bridges, and utility poles [10]. Therefore, it is important to solve the relevant technical issues and develop ALPR systems for dynamic backgrounds, which can be mounted on mobile platform to patrol the area.

The main contribution of this paper is to increase the accuracy of detection and recognition of licenses plates with the help of multiple images taken at different exposure times. However, to avoid the use of complex hardware, and to keep the technique compatible with the existing ALPR systems, we capture only one image at a fixed exposure setting and synthesise others variants using a camera model. This model, for each image captured by a mobile camera, generates four additional images for different exposure times 0.5, 1.5, 2, and 2.5. Thus, we have five images of a scene at different levels of brightness. Our hypothesis is that in dark and very bright conditions when the original image is under/over exposed, some of the synthesised images should show the LP and its characters better than the original image captured by the camera. ALPR can be done on each of the five images and results can be combined to detect and recognize LP at high accuracy.

To validate the performance of the proposed technique, we develop a deep learning based baseline ALPR system, and observe the improvement in its accuracy as result of applying the proposed method. Our ALPR system uses You Only Looks Once (YOLO) algorithm [11] to detect licenses plates. After the detection process, the images are fed into a Convolutional Neural Network (CNN) model for the recognition of characters and digits on the plate. We train both modules using a custom dataset of LPs extracted from real traffic videos captured in Saudi Arabia. Using multiple exposures as suggested above, the accuracy of the plate detection reaches 100%. The recognition phase is more challenging, as we have included some very challenging LP images, having faded digits, rusted plates, and varying size and lighting. However, we see a very significant improvement in recognition too, when images at

multiple exposures are used. Our CNN model returns the confidence level with which a character is recognized. We pick the individual characters predicted with the highest confidence level, if there are mismatches found in the results obtained from multiple plates.

As we capture only one real image of the vehicle at a fixed exposure setting, and generate the rest artificially, the proposed technique works with existing ALPR systems without requiring any changes in their capturing modules. It can be used as an add-on to their existing software. We experiment with several existing ALPR systems and observe significant increase in their accuracy, especially in the challenging scenarios.

The rest of work is organized as follows. In Section II, existing works related to each ALPR stage are discussed. This is followed by presenting the fundamental idea of the proposed ALPR method in Section III. Experiment results are presented in Section IV. Concluding remarks and ideas for future work are given in Section V.

## II. RELATED WORK

For LP detection, segmentation, and identification, several ALPR systems have been suggested. Various object detection approaches have been used by researchers to handle the LP detection step. Chen et al. [12] have reduced the YOLO model layers from 27 to 13, consisting of 7 CNN layers and 6 fully connected layers. They call this a tiny model which is used to detect only one single class. A total of 36 classes are used in which 10 classes are used for numeric data, 25 for alphabets, and 1 for plate recognition. A Taiwan License plates dataset was used and results were reported with 98.22% detection accuracy and 78% recognition accuracy.

Tourani et al. [13] divide LPR into two parts: plate detection and character recognition. A YOLOv3 model is used to detect the LP which is cropped from the image and resized to 224 x 224 pixels. Both color and grayscale images are used and the last layer of the first YOLO model is modified. For character recognition, a second YOLOv3 model is used. The method got 97.77% accuracy in detection and 95.05% in character recognition on Iranian license plates dataset.

Laroca et al. [1] used two CNN arranged in a cascaded manner to detect car frontal or back-views and LPs, having the lowest false positive rate. Rashtehroudi et al. [14] have combined the Optical Character Recognition (OCR) and segmentation method which is used for digit recognition with the help of the YOLO algorithm. Using some preprocessing steps in the first stage of the technique, they remove camera angle issues by applying a Hough Line transform and rotational filters based method on the license plate. In the second stage, they apply the Bradley method to minimize the effect of light angle on the image. After this, the noise level is reduced, and a YOLO model is trained with 1000 images. Ozbay et al. [2] used a smearing method which is a morphological process on preprocessed images. Gou et al. [15] used top-hat transformation, vertical edge detection, and morphological procedures to detect the license plate.

Yonetsu et al. [16] used two-stage YOLOv2 for precise license plate detection to decrease false detection, and thus

boost the detection rate. Jain et al. [17] used edge information and geometric characteristics to extract license plate candidates, which are then fed to a CNN classifier for license plate detection. Liu et al. [18] used a YOLOv3 detector. In their method, the picture is divided into  $T \times T$  (where  $T$  is a natural integer) rectangles, and bounding boxes and class probability are predicted in each rectangle. The rectangle with the highest confidence level is detected in the proposed technique. Wang et al. [19] utilize a vertical projection approach in which they scan license frames from left to right until a projection region with a width bigger than a preset threshold is discovered. Izidio et al. [20] created a CNN model based on synthetic imagery. Khan et al. [21] purposed a framework for LP recognition addressing the issues of low image resolution, light conditions, blurriness effect due to motion of vehicle, height, and noise. Zhang et al. [22] created three networks in their proposed method. The first network is a CNN cascade model for fast and accurate plate detection. The second network is Recurrent Neural Network (RNN) for a segmented free recognition. The third network is Generative Adversarial Networks (GAN) to enhance ALPR accuracy.

## III. PROPOSED METHOD

In this section, we explain the method to synthesise images at multiple exposure from a given image captured at a fixed exposure. Then we explain several steps in designing an ALPR system that leverages the information contained in multiple images to achieve high detection and recognition rates.

### A. Synthesising Multiple Exposures

The quality of images taken by a camera is affected by ambient light, and in very dark or bright images, it becomes difficult to discern the characters. Therefore, it is necessary to adjust the exposure time of the camera according to surrounding lighting conditions. However, it is not always possible to find a suitable setting automatically, and therefore an easier way could be to acquire multiple images at different exposure times and read the LP from the best image to increase the recognition accuracy. Capturing multiple images of a scene at different exposure settings requires more sophisticated hardware to achieve higher frame rate, especially when the vehicle and the camera are moving. In this article, we use a camera model to artificially create various images corresponding to different camera exposure times (multi-exposure images) from a single image. The proposed technique does not require changes in the existing hardware used for LP recognition and is a post-processing module that can be added to the existing systems for enhanced accuracy.

The accumulated light at a sensor location  $p$  for  $\Delta t$  units of time defines a sensor exposure  $E(p) \cdot \Delta t$ , where  $E(p)$  is sensor irradiation at  $p$ . Then, the pixel intensity value  $L(p)$  at sensor location  $p$  is defined as a function of sensor exposure as.

$$L(p) = f(E(p) \cdot \Delta t) \quad (1)$$

where,  $f$  is called the camera response function. Among various models to estimate the camera response function [23, 24], many camera manufacturers adopt a gamma curve to design the response function [25, 26]. In general, a camera response function is assumed to be a gamma curve, in which case the intensity value at pixel location  $p$  is represented as.

$$L(p) = f(E(p) \cdot \Delta t) \approx (E(p) \cdot \Delta t)^\gamma \quad (2)$$

If multiple images  $L_1, L_2, \dots, L_n$  are acquired at different camera exposure times  $\Delta t_1, \Delta t_2, \dots, \Delta t_n$ , we can obtain a gamma corrected exposure time ratio  $\alpha_i^j(p)$  [26] between  $L_i(p)$  and  $L_j(p)$  as

$$\alpha_i^j(p) = \frac{L_j(p)}{L_i(p)} \approx \left( \frac{E(p) \cdot \Delta t_j}{E(p) \cdot \Delta t_i} \right)^\gamma = \left( \frac{\Delta t_j}{\Delta t_i} \right)^\gamma \quad (3)$$

Since  $\Delta t_i, \Delta t_j, \gamma$  are known constants and the value  $\alpha_i^j(p)$  is the same for all the pixels in an image, the image  $L_k$  corresponding to camera exposure time  $\Delta t_k$  can be artificially generated from the reference image  $L_r$  by multiplying  $\alpha_r^k$  at each pixel in  $L_r$ . In this article, four images are generated: one under-exposed image ( $\alpha = 0.5$ ) and three over-exposed images ( $\alpha = 1.5, 2, 2.5$ ) with respect to the reference image  $L_r$  with the expectation that one of these five images (including the reference image) will provide better quality for discerning LP numbers under given ambient lighting conditions.

We apply our LP detection and recognition algorithm proposed next in this section to each of these five images. The algorithm returns a value of confidence level for each detected alphanumeric. Among the characters detected multiple times in these five images, we take the one detected with the highest confidence level.

### B. Data Collection

To improve the detection model's accuracy, an adequate amount of labeled data is required. In this part, we explain the data collection procedure to train and test the LP detection and recognition networks. First, the videos of vehicles were recorded using a camera positioned on a moving car. The data is collected from the real traffic in Saudi Arabia and includes different LPs of different color and size according to vehicle type and size. In our study, we only consider cars' LPs which are 32cm by 16cm in size. We created a framing algorithm that extracts frames from the recorded videos. A total of 950 frames containing cars were extracted and stored in the Portable Network Graphics (PNG) format at 1920x1080 resolution. The dataset comprises frames containing LPs with easily readable

text, as well as several challenging LPs in which text is faded, covered with mud, or distorted in several other ways. Some samples are shown in Fig. 1.

After extracting these frames, we applied different exposures ( $\alpha = 0.5, 1.5, 2, 2.5$ ) to them using the method described above and thus a total of 3800 frames were obtained. A typical set of five frames is shown in Fig. 2, where the frame marked as 1x is captured by the camera while the rest are generated using our model stated above. To produce more realistic scenarios, we applied augmentation techniques such as gray scaling, brightness variation, rotation of LPs, and flipping and rotation of digits and characters. This way, in total, we got 5000 unique images of LPs in our dataset.

Each frame in the dataset is labeled using LabelImg tool [27]. The software gives XML files that contain coordinates of bounding boxes.

### C. Detection and Recognition

We train YOLOv5 network [28] for detection of LPs, on a computer equipped with Rx 570 GPU. YOLOv5 is considered a better model for object detection compared to its competitors YOLO [18], YOLOv3 [20], and Faster-RCNN [29] in terms of storage, processing speed, and auto anchoring. The training images are fed into the object detection model YOLOv5, along with the XML files extracted in the previous step mentioned above. Using transfer learning, we extract the weights of YOLOv5. As shown in Table I, we split the dataset: 70% data for training the model, 10% for the validation process, and 20% for testing. At convergence, we got 98% accuracy in LP detection on the testing dataset. After detecting LP, we move to the next step which is the segmentation of the dataset and distributing the LP characters among 27 classes of digits and alphabets. LPs in Saudi Arabia use only 17 characters and 10 digits, as shown in Fig. 3, and there are three letters and four digits in an LP. For the recognition process, a convolutional neural network with convolution layers at the beginning and fully linked layers at the end is used, as shown in Fig. 4, where the layers are labeled as follows:

Conv<number of filters><size of filters>



Fig. 1. Some Samples of License Plates in our Dataset.



Fig. 2. Five Images of the Same Scene at different Camera Exposures. The Second Image from Left is Captured with Camera while the Rest are Synthesies using a Camera Model.

TABLE I. DATA SPLIT RATIO OF TRAINING DATASET FOR LP DETECTION

Dataset	Split Percentages	Number of Frames
Training	70%	3500
Testing	20%	1000
Validation	10%	500



Fig. 3. LP Dataset.

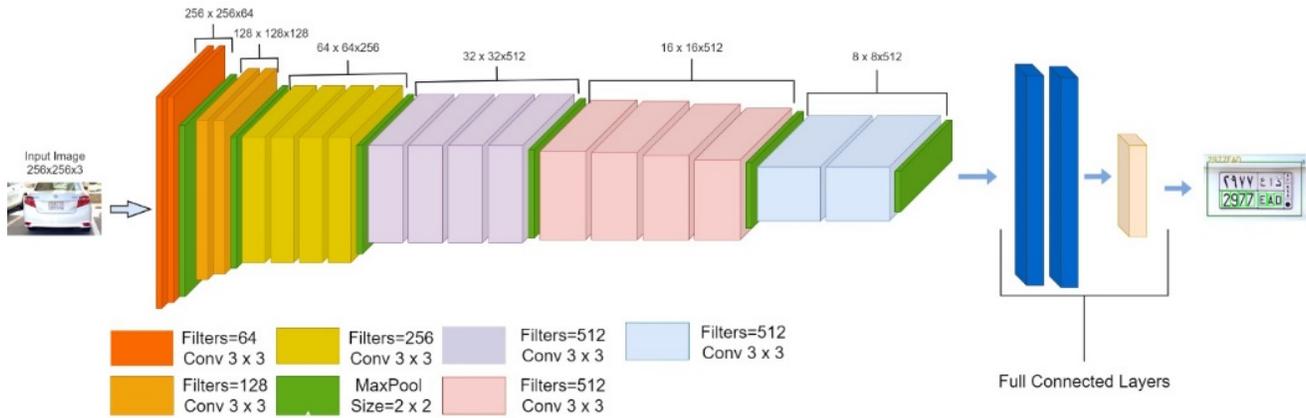


Fig. 4. Proposed CNN Architecture.

The size of the kernel is 3 x 3 and input image size is 416 x 416 x 3. To extract rich information from a given image, we used several alternating structures of multiple convolutional layers and nonlinear activation layers. We used activation function ReLU with stride size 1 and Maxpool layer with size 2x2. The output is passed to another completely linked layer with 4096 neurons and then to another fully connected layer with 1000 neurons. At the end of the network, SoftMax is used to make the final decision. At the input stage, the ReLU [30] function is applied while the Sigmoid [31] function is used at the output stage. For optimization, we employ the Adam optimizer with an initial learning rate of  $\alpha_0 = 1E - 4$  and we use categorical cross-entropy as the loss function which is commonly used in multi-class classification problems. This function deals with problems in which the output of the model belongs to a specific class. To protect our model from overfitting, we drop some values of our neurons and use early stopping which saves us from the long execution time taken by the model for training. For the training of this model, we split the dataset into training, validation, and testing sets with the same ratio as shown above in Table I.

Several methods have been proposed for the detection and recognition of LP. In the proposed method, we extract a frame

from a video containing vehicle LP and apply the proposed multi-exposure method to that frame. YOLOv5 is used to detect the LP from frames. Once the LP is detected, we crop the LP part from the images and apply a sliding window approach [32] with our custom CNN model on that cropped image to detect and recognize LP digits. The output of the CNN contains 27 confidence levels, one for each class. After going through the whole frames, we get coordinate of each character with its class label and confidence level. Then, at each character coordinate in LP, the confidence levels of class labels derived from images with different exposure times are compared. Finally, the class label with the highest confidence level is determined as the final predicted character.

In the proposed technique, only the LPs with Intersection over Union (IoU) values greater than 0.69 are detected, and the character class label is selected based on the highest confidence level obtained from the CNN model. IoU is a measure of the overlapping of the annotated and the predicted bounding boxes. We come up with that number after conducting a series of experiments on different LPs. A number of experiments are conducted on the dataset for the optimum value of IoU. If we chose a value less than 0.69, then it detects some objects containing features like LPs.

#### IV. RESULT AND DISCUSSION

In this section, we conducted experiments to verify the effectiveness of our proposed methodology. The PyTorch framework was used for all the experiments. To calculate the accuracy of detection and recognition we use the following equations:

$$\text{Detection accuracy} = \frac{\text{total number correct detections}}{\text{total number of images}} \quad (4)$$

$$\text{Recognition accuracy} = \frac{\text{total number of correct recognitions of alphanumeric characters}}{\text{total number of alphanumeric characters}} \quad (5)$$

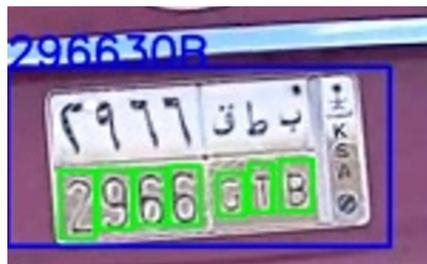
We test the trained modules of detection and recognition of LP using our testing dataset. Results are shown in Table II. A 95.34% detection accuracy is achieved on 1x exposure time images and 95.75% on multi-exposure images. For recognition of each digit and alphabet, augmentation and multi-language

repudiate techniques [33] proved helpful. We get the overall recognition accuracy of 68.2% for 1x exposure time images and 75.965% recognition accuracy for different exposure time images. One example demonstrating the effectiveness of using multi-exposure images is shown in Fig. 5. The alphabet G and T are not recognised in the single exposure time image whereas they are recognized by the proposed technique of using multi-exposure images.

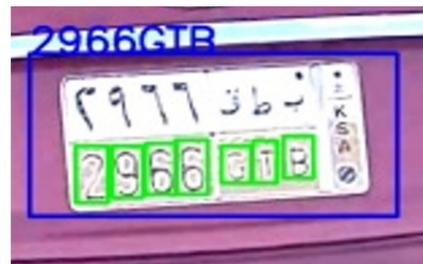
The proposed multi-exposure techniques can work with other existing techniques and improve their accuracy. We demonstrate this using three recent techniques [11, 33, 34]. In [11], the authors used YOLOv5 for the detection of LPs and digits, and proposed a custom CNN architecture for characters recognition. The authors of [33] used YOLOv5 with 19 CNN layers for detection of each class on Saudi Arabia LPs. In [34], thresholding and OCR techniques were used to recognize the digits. We applied the proposed multi-thresholding technique to all these three methods [11, 33, 34] and observed significant improvement in their accuracy, as shown in Table II.

TABLE II. COMPARISON OF DETECTION AND RECOGNITION ACCURACY WITH SINGLE AND MULTI-EXPOSURE IMAGES

Methods	Detection Single Exposure	Detection Proposed multi-exposures	Recognition Single Exposure	Recognition Proposed multi-exposures
[11]	94.39%	95.15%	66.58%	73.45%
[33]	95.34%	95.75%	66.34%	74.585%
[34]	83.3%	83.49%	69.975%	71.8%
Proposed Method	95.34%	95.75%	68.2%	75.965%



(a) Single exposure time



(b) Multi-exposure times

Fig. 5. Detection and Recognition Result on Single and Multi-exposure Images.

#### V. CONCLUSION

In this paper, we proposed a method utilizing multi-exposure images to improve both detection and recognition accuracies in ALPR. The proposed approach aims to present an efficient system for automatically detecting and recognizing license plates (LP) in unconstrained real-time settings with the help of different exposure time images. The proposed ALPR system is trained and evaluated on a custom dataset containing 5000 images. Experimental results show that the accuracies of the existing ALPR methods can be improved by using multi-exposure images. In this work, the multi-exposure images are generated artificially. Therefore, for the pixels having very dark or bright shades, the synthesised values may not be very accurate, however, still a significant improvement in accuracy of recognition is observed. In future work, we plan to further improve the accuracy of ALPR by using real images captured at multiple exposure times.

#### ACKNOWLEDGMENT

The authors extend their appreciation to the Deputyship for Research & Innovation, Ministry of Education in Saudi Arabia for funding this research work through the project number MoE-IF-G-20-12.

#### REFERENCES

- [1] I. R. Laroca, E. Severo, L. A. Za nlorensi, L. S. Oliveira, G. R. Gonçalves, W. R. Schwartz, and D. Menotti, "A robust real-time automatic license plate recognition based on the YOLO detector," International Joint Conference on Neural Networks, pp. 1–10, 2018.
- [2] S. Ozbay and E. Ercelebi, "Automatic vehicle identification by plate recognition," World Academy of Science, Engineering and Technology, vol. 9, pp. 222–225, 2005.
- [3] A. Roy and D. P. Ghoshal, "Number Plate Recognition for use in different countries using an improved segmentation," 2nd National Conference on Emerging Trends and Applications in Computer Science, pp. 1–5, 2011.

- [4] I. Kilic and G. Aydin, "Turkish vehicle license plate recognition using deep learning," International Conference on Artificial Intelligence and Data Processing (IDAP), pp. 1–5, 2018.
- [5] H. Xiang, Y. Zhao, Y. Yuan, G. Zhang, and X. Hu, "Lightweight fully convolutional network for license plate detection," *Optik*, vol. 178, pp. 1185–1194, 2019.
- [6] Z.-X. Chen, C.-Y. Liu, F.-L. Chang, and G.-Y. Wang, "Automatic license-plate location and recognition based on feature salience," *IEEE Trans. Veh. Technol.*, vol. 58, pp. 3781–3785, 2009.
- [7] O. Bulan, V. Kozitsky, P. Ramesh, and M. Shreve, "Segmentation-and annotation-free license plate recognition with deep localization and failure identification," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, pp. 2351–2363, 2017.
- [8] A. Safaei, H. L. Tang, and S. Sanei, "Real-time search-free multiple license plate recognition via likelihood estimation of saliency," *Comput. Electr. Eng.*, vol. 56, pp. 15–29, 2016.
- [9] P. Marzuki, A. Syafeeza, Y. Wong, N. Hamid, A. N. Alisa, and M. Ibrahim, "A design of license plate recognition system using convolutional neural network," *Int. J. Electr. Comput. Eng.*, vol. 9, pp. 2088–8708, 2019.
- [10] Y. Wen, Y. Lu, J. Yan, Z. Zhou, K. M. von Deneen, and P. Shi, "An algorithm for license plate recognition applied to intelligent transportation system," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, pp. 830–845, 2011.
- [11] R. Brüngel, J. Rückert, and C. M. Friedrich, "DFUC 2020: An Approach on Diabetic Foot Ulcer Detection via YOLOv5."
- [12] R. C. Chen, "Automatic License Plate Recognition via sliding-window darknet-YOLO deep learning," *Image Vis. Comput.*, pp. 47–56, 2019.
- [13] A. Tourani, A. Shahbahrami, S. Soroori, S. Khazae, and C. Y. Suen, "A robust deep learning approach for automatic iranian vehicle license plate detection and recognition for surveillance systems," *IEEE Access*, vol. 8, pp. 201317–201330, 2020.
- [14] A. R. Rashtehroudi, A. Shahbahrami, and A. Akoushideh, "Iranian license plate recognition using deep learning," International Conference on Machine Vision and Image Processing (MVIP), pp. 1–5, 2020.
- [15] C. Gou, K. Wang, Y. Yao, and Z. Li, "Vehicle license plate recognition based on extremal regions and restricted Boltzmann machines," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, pp. 1096–1107, 2015.
- [16] S. Yonetsu, Y. Iwamoto, and Y. W. Chen, "Two-stage YOLOv2 for accurate license-plate detection in complex scenes," *IEEE International Conference on Consumer Electronics (ICCE)*, pp. 1–4, 2019.
- [17] V. Jain, Z. Sasindran, A. Rajagopal, S. Biswas, H. S. Bharadwaj, and K. Ramakrishnan, "Deep automatic license plate recognition system," Proceedings of the Tenth Indian Conference on Computer Vision, Graphics and Image Processing, pp. 1–8, 2016.
- [18] Z. Liu, Z. Wang, and Y. Xing, "Wagon number recognition based on the YOLOv3 detector," *IEEE 2nd International Conference on Computer and Communication Engineering Technology*, pp. 159–163, 2019.
- [19] Q. Wang, "License plate recognition via convolutional neural networks," *IEEE International Conference on Software Engineering and Service Science (ICSESS)*, pp. 926–929, 2017.
- [20] D. M. Izidio, A. P. Ferreira, H. R. Medeiros, and E. N. D. S. Barros, "An embedded automatic license plate recognition system using deep learning," *Des. Autom. Embed. Syst.*, pp. 1–21, 2019.
- [21] K. Khan and M. R. Choi, "Automatic license plate detection and recognition framework to enhance security applications," *J. Electron. Imaging*, vol. 28, no. 1, pp. 013036, 2019.
- [22] X. Zhang, N. Gu, H. Ye, and C. Lin, "Vehicle license plate detection and recognition using deep neural networks and generative adversarial networks," *J. Electron. Imaging*, vol. 27, no. 4, pp. 043056, 2018.
- [23] J. Qiu and H. Xu, "Camera response prediction for various capture settings using the spectral sensitivity and crosstalk model," *Appl. Opt.*, vol. 55, pp. 6989–6999, 2016.
- [24] M. D. Grossber and S. K. Nayar, "What is the Space of Camera Response Functions," *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. II–602, 2003.
- [25] S. -O. Shim, S. Alharbi, I.R. Khan, and W. Aziz, "De-ghosting in High Dynamic Range Imaging Based on Intensity Scaling Cue", *Adv. Electr. Comput. Eng.*, vol. 20, no. 3, pp. 3–10, 2020.
- [26] S. -O. Shim, "Estimation of gamma-corrected exposure time ratio in multi-exposure images for removal of moving objects", *Appl. Opt.*, vol. 59, no. 13, pp. 4076–4080, 2020.
- [27] C. -W. Yu, Y. -L. Chen, K. -F. Lee, C. -H. Chen, and C. -Y. Hsiao, "Efficient Intelligent Automatic Image Annotation Method based on Machine Learning Techniques," *IEEE International Conference on Consumer Electronics-Taiwan (ICCE-TW)*, pp. 1–2, 2019.
- [28] <https://github.com/ultralytics/yolov5>.
- [29] M. A. Rafique, W. Pedrycz, and M. Jeon, "Vehicle license plate detection using region-based convolutional neural networks," *Soft Comput.*, vol. 22, pp. 6429–6440, 2018.
- [30] A. F. Agarap, "Deep learning using rectified linear units (relu)." *arXiv preprint arXiv:1803.08375*, 2018.
- [31] D. Costarelli and R. Spigler, "Approximation results for neural network operators activated by sigmoidal functions," *Neural Netw.*, vol. 44, pp. 101–106, 2013.
- [32] N. I. Glumov, E. I. Kolomiyetz, and V. V. Sergeev, "Detection of objects on the image using a sliding window mode," *Opt. Laser Technol.*, vol. 27, no. 4, pp. 241–249, 1995.
- [33] S. T. A. Ali, A. H. Usama, I. R. Khan, M. M. Khan and A. Siddiq, "Mobile Registration Number Plate Recognition Using Artificial Intelligence," *IEEE International Conference on Image Processing (ICIP)*, pp. 944–948, 2021.
- [34] <http://34.https://platerecognizer.com>.