

Article

Towards Automatic License Plate Recognition in Challenging Conditions

Fahd Sultan ¹, Khurram Khan ², Yasir Ali Shah ³, Mohsin Shahzad ³ , Uzair Khan ³  and Zahid Mahmood ^{3,*} 

¹ Department of Software Development, Axispoint Technology Solutions Group (ATSG), New York, NY 10119, USA

² Faculty of Computer Science and Engineering, GIK Institute of Engineering Sciences and Technology, Topi 23460, Pakistan

³ Department of Electrical and Computer Engineering, COMSATS University Islamabad, Abbottabad Campus, Abbottabad 22060, Pakistan

* Correspondence: zahid0987@cuiatd.edu.pk

Abstract: License plate recognition (LPR) is an integral part of the current intelligent systems that are developed to locate and identify various objects. Unfortunately, the LPR is a challenging task due to various factors, such as the numerous shapes and designs of the LPs, the non-following of standard LP templates, irregular outlines, angle variations, and occlusion. These factors drastically influence the LP appearance and significantly challenge the detection and recognition abilities of state-of-the-art detection and recognition algorithms. However, recent rising trends in the development of machine learning algorithms have yielded encouraging solutions. This paper presents a novel LPR method to address the aforesaid issues. The proposed method is composed of three distinct but interconnected steps. First, a vehicle that appears in an input image is detected using the Faster RCNN. Next, the LP area is located within the detected vehicle by using morphological operations. Finally, license plate recognition is accomplished using the deep learning network. Detailed simulations performed on the PKU, AOLP, and CCPD databases indicate that our developed approach produces mean license plate recognition accuracy of 99%, 96.0231%, and 98.7000% on the aforesaid databases.

Keywords: Faster RCNN; license plate recognition; object detection



Citation: Sultan, F.; Khan, K.; Shah, Y.A.; Shahzad, M.; Khan, U.; Mahmood, Z. Towards Automatic License Plate Recognition in Challenging Conditions. *Appl. Sci.* **2023**, *13*, 3956. <https://doi.org/10.3390/app13063956>

Academic Editor: Sungho Kim

Received: 6 February 2023

Revised: 13 March 2023

Accepted: 16 March 2023

Published: 20 March 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

With the growth of big data, object detection and recognition have attracted excellent interest in research communities. This is because it can be used for a wide range of real-world applications, such as medical imaging, augmented reality, sports applications, independent driving, and video surveillance [1–6]. Particularly, the license plate recognition (LPR) is getting more attention due to its widespread applications in various fields, for instance, traffic monitoring, toll collection, and criminal searches [7,8]. Although many of the LPR systems, for instance [9,10], are available in the literature, most of them have been validated and tested on a pre-defined LP specification. Few of these works are also capable of processing multiple LPs. The LPR systems can be categorized into two major categories, which are (i) traditional LPR and (ii) deep learning-based LPR systems. Traditional methods process limited features and utilize hand-crafted features, for example, contours, colors, and edges, to locate the LP. Deep learning-based techniques automatically learn robust features from the data and have recently produced promising results. Deep learning-based techniques consider LP detection as object detection and analyze the recognition as an optical character recognition (OCR) process. Since the number of characters processed by any LPR method is limited, character recognition is also considered an object detection process, so that LP detection and recognition are handled simultaneously. To develop and analyze a LPR algorithm that can deal with multi-style license plate recognition, there are several challenges, as briefly described below.

Lack of Standard LPs: Standardizing the LPs is a significant challenge throughout the world. For instance, license plates in Macao, China, have to meet a strict standard, as shown in Figure 1a [11], whereas the LP of Hong Kong usually has 1–8 characters and Macao LPs are composed of five to six characters, as shown in Figure 1b,c, respectively. General observation in Figure 1a indicates that the first column has an entirely different appearance than the second column. Similarly, the second and third columns have huge contrast and appearance variations. Few of the plates have yellowish and greenish backgrounds, while few have colorless backgrounds. Moreover, the distance between the characters on the plates shown in the first column takes up much less space than those in the third column.

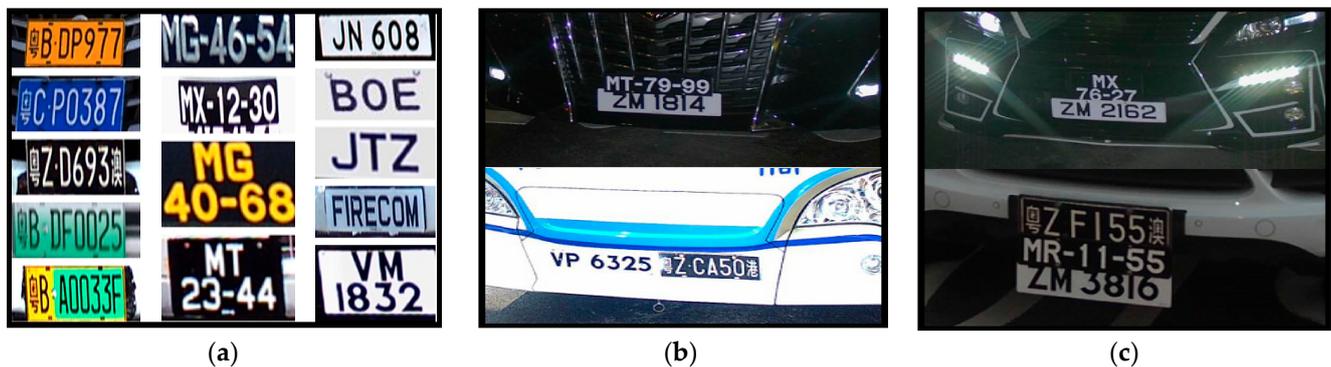


Figure 1. Various license plate challenges.

Lack of LP Outlines: Many times, the license plates have no outlines, which makes it an extremely difficult task to classify. This task becomes more challenging when the color of the vehicle is the same as that of the LPs. One such example is indicated in Figure 1b. The top image shown in Figure 1b contains a black background and white ground on a black vehicle. Moreover, the bottom vehicle shown in Figure 1b shows a LP case that has a white background due to the white color of the car and a black front ground. These two cases may appear trivial, but for any machine learning algorithm, the aforescribed task is not easy. The algorithm should be capable enough to distinguish such cases accurately.

The appearance of LPs: Another challenge in LP localization and recognition is to accurately handle the appearance and occlusion challenges of license plates installed at various locations on a vehicle. If a vehicle has more than one license plate, then characters get matched with the background of other plates, which makes it difficult to distinguish LP character regions. A few cases of this challenge are shown in Figure 1c. Figure 1c depicts two license plates with total complement variations on the top vehicle. Moreover, along with the low-intensity light beam, one plate has a white background with black characters on the plate region, while the other plate has a black background with white alphabets on top. In both cases, the vehicle color appears blackish–green. In each of these cases, it poses a significant challenge to any recognition algorithm.

Therefore, one of the aims of this research is to design an accurate LP recognition technique with the capability to handle diverse license plates. Fortunately, numerous research groups have compiled numerous LP datasets. Few of the databases also contain clear vehicle images in different environments and road conditions. We aimed to contribute to the field with this manuscript, as highlighted below.

- Inspired by recent trends in machine learning, a robust LP recognition method is proposed in this paper that accurately recognizes various license plates. Particularly, our developed system uses an intelligent combination of Faster RCNN to detect various vehicles, morphological image processing methods to locate the LP area, and finally, the deep learning-based method to recognize the detected license plate. The systemic application of various modules enables us to achieve a reliable and accurate LP recognition method.

- We consider the license plate as an object for detection and recognition tasks. The output of our created approach for detecting vehicles is a rectangle encompassing the vehicle and license plate region. In contrast, for the license plate recognition task, our system displays the vehicle license plate alphabets and characters above the real license plate once it has been located in the image.
- Our developed technique supports different types of plates from PKU, AOLP, and CCPD datasets. Our obtained results indicate that our developed method effectively recognizes the LPs of these databases. Moreover, our developed technique is intelligent, as it systematically achieves the aforescribed tasks.

This manuscript is organized as follows: In Section 2, recent license plate recognition techniques are briefly listed. In Section 3, our proposed method is described in detail. Our findings during simulations are listed in Section 4, followed by the conclusions, which also hint at the future extension of this work.

2. Related Work

This section details recent related works on license plate recognition. In [11], a region-based license plate detection method is discussed that initially shifts mean to filter and segment a color vehicle image to get candidate regions. These candidate regions are then analyzed to decide whether a candidate region contains a license plate. Since this method focuses on regions, so it is more robust to interference characters. In [12], the proposed method uses the YOLOv2 to detect vehicles. This work uses a CNN-based method that they refer to as WPOD-NET for LP detection. Meanwhile, a modified YOLO architecture recognizes the LP characters. However, this work also uses character segmentation, which makes this method a bit more complex than the compared methods. In [13], an improved YOLO architecture is deployed for character recognition. This work is tested on the SSIG dataset, which has 2000 images. In [14], a technique is developed that customizes the YOLO network to detect LPs from images that are captured in different conditions, such as different weathers, varying lighting, and other factors. The authors conclude that the YOLO can strike a balance between precision and recall. However, the YOLO is not suitable for detecting angular or small objects. Therefore, its performance in scenarios where the vehicle is far away from the camera needs to be further checked.

In [15], a framework to detect and recognize license plates is discussed for complex scenes, which is based on mask region convolutional neural networks. The evaluation of this framework is further enhanced on four publicly available datasets for different countries. Moreover, this method is tested on diverse range of images, which are captured from multiple scenes, such as varying orientations, poor image quality, blurred images, and complex backgrounds. In [16], a convolutional filter of size 3×3 is used in deep networks to analyze the increasing depth of the architecture by using 16 to 19 layers to process the 24×24 pixel colored image. This work also introduces a pre-processing step by subtracting the average RGB value from each individual pixel. This paper reports significant improvements to ConvNets in the realm of image recognition as a result. In addition, this work also uses a large number of 3×3 convolutional filters that fit well on the investigated datasets only. In [17], initially, candidate regions are selected through a sparse network using winnows classification, followed by filtration through CNN. An interesting novelty introduced in this work is the minimization of training and target domains in an unsupervised manner. However, this work also considers artificially generated synthetic LP images. In [18], the developed method utilizes thin-plate spline transformation and adaptively rectifies a textual LP image. Moreover, a recognition model predicts a character sequence immediately from the rectified image. This work only considers qualitative results on several images. In [19], a 2D attention-based encoder–decoder architecture is developed. This method extracts features by applying the ResNet CNN architecture. The 2D model introduced is capable of accommodating text with different layouts, arbitrary shapes, and different angles. Their reported results are encouraging, and their development reduces data bias and increases model generalization capacity. This method is simple; however, its

generalization to standard datasets has not been explored. In a previous study [20], the authors used CycleWGAN to create LP images to improve the performance of recognition. Their work simultaneously generates images of different conditions. Meanwhile, a modified version of the CTC is used to recognize the LP. Their work simultaneously generates images of different conditions. Meanwhile, a modified version of the CTC is used to recognize the LP.

In [21], an end-to-end irregular LPR (EILPR) is proposed using plate-level annotations during training. In the EILPR method, a coarse-to-fine approach is implemented that extracts the LP features for sequence recognition. This work assumes the fact that a LP may generate a perspective bias in the image; therefore, to cater to this fact, an automatic perspective alignment network (APAN) is introduced to extract the fine license plate features. To classify the international license plates, a location-aware 2D attention-based recognition network is used. In [22], a novel ALPR technique, which is referred to as VSNet, is developed. The VSNet contains two CNNs that are combined in a cascading manner. Meanwhile, an integration block is introduced that extracts the spatial features. With vertex supervisory information, authors develop a vertex-evaluation module in VertexNet such that a LP can be repaired as the input images of SCR-Net. A horizontal encoding algorithm is used in the SCR-Net to extract left-to-right features and then recognize a license plate. This work performs well on standard LPs. However, its generalization capability on tilting and rotating LPs has not been explored.

Additionally, ALPRNet is developed to detect and recognize mixed-style license plates [23]. Two fully convolutional object detectors are used in the proposed ALPRNet to classify and recognize LPs. The proposed ALPRNet processes LP and character equally. In this work, object detectors output bound boxes of LPs along with corresponding labels without the application of the RNN branches of the OCR. This is because this is a single-stage network-based method. Therefore, its detection accuracy on challenging datasets has not been explored. In [24], image processing and OCR-based techniques are merged to recognize the LPs. The image processing methods utilize color conversion, Otsu's thresholding, and noise removal. The OCR method uses template matching to predict the characters of LPs. The authors of this work have not examined the scalability of this method and have only used basic tools from signal and image processing. In [25], the proposed LPR method consists of three steps: LP detection, unified character recognition, and multinational LP layout detection. This work is primarily based on the YOLO networks. To extract the correct sequence, a layout detection scheme is introduced, which extracts the sequence of LP numbers from multinational LPs. This study is extensively tested on standard Korean and Taiwan LPs. In [26], the developed LPR method uses a joint combination of adaptive boosting and the LDA to extract features. The CNNC is then used to separate the LP region from irrelevant samples. This work is segmentation-free. However, its recognition capability on real-world images has not been explored. In [27], the developed algorithm uses a distinct, fine-tuned YOLO-v3 platform to extract LP characters from input images. During the training and testing stages, a wide range of LP images have been analyzed. However, this system is fully annotated and consumes over 100 ms to accomplish the task of LP recognition. In addition, an intriguing review article is released that summarizes the many approaches currently utilized to detect various objects [28].

In [29], researchers introduced a robust vehicle detection method using a multi-scale deep convolutional neural network. This work utilizes a standard Gaussian mixture probability hypothesis density filter along with hierarchical data associations (HDA) that isolate detection-to-track and track-to-track associations. Particularly, the cost matrix of various phases is solved using the Hungarian algorithm. For quick execution, detection information, such as bounding boxes and detection scores, is used in the HDA without visual feature information. Although this is an interesting work, the computational difficulty of the approach is not covered. In [30], a region proposal network (RPN) is developed that shares full-image convolutional features with the detection network. The RPN is a fully convolutional network that forecasts object bounds and scores at various positions. The

RPN is trained end-to-end to generate high-quality region proposals that are later used by Fast R-CNN for detection. In this work, the RPN and Fast R-CNN are also merged into a single network by sharing their convolutional features. For the very deep VGG-16 model, this system has a frame rate of 5 fps on a GPU, while achieving encouraging object detection accuracy on several datasets with only 300 proposals per image.

In [31], a wavelet transform based technique to extract license plates from cluttered images is developed. This method comprises of three major stages, which are (i) extracting important contrast features through wavelets. Then, finding a reference line in HL subimage plays an important role to locate the desired license plate region roughly. According, (ii) decrease the searching region of license plate to speed up the execution time, and (iii) localization of license plate through manual adjustments. More importantly, the proposed detection method can locate multiple plates with different orientations in one image. Since the feature extracted is robust to complex backgrounds, the proposed method works well in extracting differently illuminated and oriented license plates. The average accuracy of detection is 92.4%. In [32], authors made use of a combination of the MSER and the stroke width transform (SWT) to detect and isolate the LP character regions. The license plates were finally bordered using the probabilistic Hough transform. The authors discuss that character-based methods are reliable and can lead to a high recall. However, the other text in the image background has a significant impact on performance. This method requires multiple cameras before the system is placed for evaluation. In [33], an interesting license plate recognition system is developed using a sequence of deep CNNs. These CNNs are trained and fine-tuned so that they are robust under different conditions (for instance, lighting, occlusion, or tilt) and work across a variety of license plate templates that include different sizes, backgrounds, or fonts. In [34], a novel line density filter approach was developed that connects regions with high edge density and removes sparse regions in each row and column from a binary edge image. This study indicates that edge-based methods are fast in computation but cannot be applied to complex images because they are too sensitive to unwanted edges.

In [35], the developed LP method consists of three modules for plate detection, character segmentation, and recognition. This method also formulates edge clustering to solve plate detection for the first time. A bilayer classifier, which is improved with an additional null class, is empirically proven to be better than previous methods for character recognition. However, this method is evaluated only on a single dataset, which was also gathered by the authors themselves. In [36], license plate detection and recognition are tackled in standard natural scene images via the development of a segmentation-free method. Inspired by the success of DNNs, these are deployed to learn high-level features in a cascade framework, which leads to improved performance on both detection and recognition. This work also trains 37 CNNs to detect all characters in an image, which results in a high recall. Later, to improve the IoU ratio, bounding box refinement is carried out based on the edge information of the LPs. This method extracts license plates effectively with both high recall and precision. Last, a recurrent neural network with long short-term memory (LSTM) is trained to recognize the sequential features extracted from the whole license plate via CNNs. For scene and lighting variations, this method needs to be further explored. In [37], a unified deep neural network is proposed that localizes license plates and recognizes the letters simultaneously in a single forward pass. This whole network is trained end-to-end and achieves the LP recognition task in a single network, avoiding intermediate error accumulation and resulting in faster processing speed. For performance evaluation, a few datasets that include images captured from various scenes under different conditions are tested. However, this method does not consider the complexity of the developed method.

In [38], researchers use computer graphic scripts and GANs to generate and augment a large number of annotated, synthesized LPs with realistic colors, fonts, and character composition from a small number of real, manually labeled license plate images. In this work, generated and augmented data are mixed and used as training data for the LP recognition network modified from the DenseNet. Simulations reveal that the model

trained from the generated mixed training data has much better generalization ability and achieves encouraging detection and recognition accuracy on multiple datasets, even with a very limited number of original real license plates. In [39], a new license plate recognition technique is developed in the wild. This method comprises a tailored CycleGAN model for license plate image generation and an elaborately designed image-to-sequence network for plate recognition. The CycleGAN-based plate generation engine eases the exhausting human annotation work. In this work, huge amounts of training data are obtained with a more balanced character distribution and various shooting conditions that boost the recognition accuracy to a large extent. Moreover, a 2D attentional-based license plate recognizer with an Xception-based CNN encoder is developed that is capable of recognizing various LPs with different patterns under various scenarios accurately.

In [40], a new license plate dataset, to which the authors refer as the CCPD, is developed and tested under different circumstances, for instance, tilt, blur, rotate, or varying weather conditions. This work is novel in the sense that it provides a single platform for researchers to investigate the LP's prevailing issues. In [41], a novel end-to-end method for LP recognition without initial character segmentation is presented as LPRNet. Particularly, this method is inspired by recent breakthroughs in the DNNs and works in real-time with recognition accuracy up to 95% for Chinese license plates: 3 ms/plate on NVIDIA^R GeForceTM GTX 1080 and 1.3 ms/plate on the Intel R CoreTM i7-6700K CPU. The LPRNet consists of the lightweight CNN and can be trained end-to-end. The authors of this work recommend that the LPRNet algorithm may be used to create embedded solutions for LPR that feature high levels of accuracy even on challenging Chinese license plates. In [42], a multi-object rectified attention network (MORAN) is proposed for text recognition. The MORAN consists of a multi-object rectification network and an attention-based sequence recognition network. The multi-object rectification network is designed to rectify images that contain irregular text. It decreases the difficulty of recognition and enables the attention-based sequence recognition network to read irregular text. The attention-based sequence recognition network focuses on target characters and sequentially outputs the predictions. Further, to improve the sensitivity of the attention-based sequence recognition network, a fractional pickup algorithm is also developed for an attention-based decoder during the training phase. In [43], a novel decoupled attention network (DAN) is developed that decouples the alignment operation from using historical decoding results. The DAN is an effective, flexible, reliable, and robust end-to-end text recognizer and consists of three components: a feature encoder, a convolutional alignment module, and a decoupled text decoder that generates final predictions by jointly using the feature map and attention maps. Yu et al. [44] used a wavelet transform at first to get the horizontal and vertical details of an image. Meanwhile, empirical mode decomposition (EMD) analysis was employed to deal with the projection data and locate the desired wave crest that indicates the position of a license plate appearing in any corner of the input image. Different versions of YOLO [45–47], which give state-of-the-art accuracy for object detection, have been published in the last few years.

The attempts outlined above are just a few examples of the numerous object detection and recognition algorithms that aim to overcome various LP recognition challenges. The following are a few of the primary reasons that prompted us to create a state-of-the-art license plate recognition algorithm.

- Most of the above-described methods and works have been carried out on standard databases that are gathered by researchers at different times under different conditions. Therefore, it prompted us to develop an algorithm that can reliably handle real-life images in real time while maintaining high recognition accuracy.
- Our study indicates that the methods, which use RNNs as the OCR, are costly in terms of execution time. Similarly, segmentation-based methods are mostly dependent on segmentation performance and highly susceptible to environmental conditions, such as varying illumination conditions, wild weather, or blurring. Therefore, these methods result in low recognition accuracy in such conditions. Even a strong recognizer, if

applied, would produce much lower recognition rates. Therefore, inspired by the aforementioned fact, we aimed to develop a license plate recognition method that could perform well under the scenarios described above.

- The PKU dataset, which is also investigated in this study, contains five prominent classes of vehicles on main highways. These categories cover different day times, varying weather conditions, multiple vehicles and license plates per image, occlusions, and crosswalks on the main highways. The scenarios mentioned are from real life, in which the detection and recognition accuracy of any algorithm might be significantly challenged.
- Many times, the cameras installed on the main highways of various countries in the world capture vehicle images in which license plates appear at an angle, tilted, or partially obscured. This motivates us to develop a system that could facilitate the traffic control and monitoring staff’s ability to reliably recognize any suspicious license plate.

3. Methodology

Our developed method has three major modules, which are vehicle detection, license plate detection, and license plate recognition. Figure 2 illustrates the complete flow of our developed method that achieves the aforementioned tasks. The details of each component of the developed method are described below.

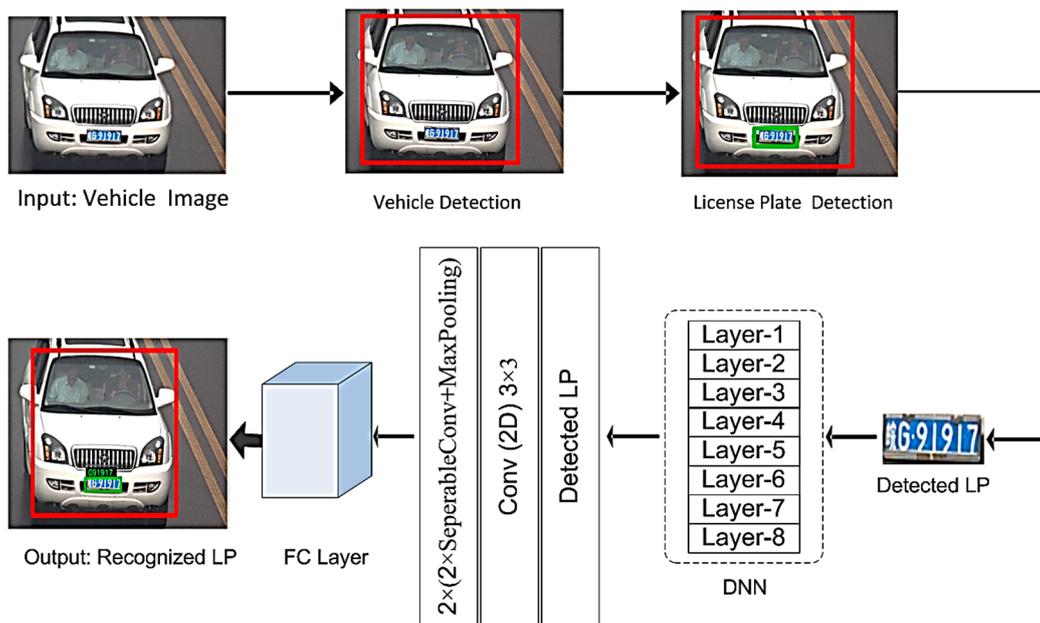


Figure 2. Flow of the proposed license plate recognition procedure.

3.1. Vehicle Detection

To locate objects, for instance, vehicle detection is a critical phase in developing an intelligent traffic monitoring system. In the past few years, the computer vision domain has introduced efficient object detection algorithms. Particularly, Faster RCNN and deep learning-based vehicle detection methods report high detection accuracy in near real-time in different environments [29]. Ultimately, these approaches have become a significant part of autonomous vehicles and self-driving applications. Our research reveals that real-time processing to locate vehicles, as well as good detection accuracy, are essential requirements that any object or vehicle detector should meet. We use a fine-tuned version of the Faster R-CNN [30] to find a vehicle quickly. The reason to detect the vehicle is that it considerably reduces the area to be explored for the existence of the LP in later stages. The purpose of using the Faster R-CNN at this stage is that, during the data training phase, it is at least

nine times more rapid than the standard R-CNN. Moreover, it is $213\times$ faster during the test phase and yields higher detection accuracy than its counterpart [30].

Algorithm 1 demonstrates the pseudocode of the employed vehicle detection module. In lines (2) to (17), Faster RCNN is used to locate vehicles' positions. In lines 3–9, Faster R-CNN is fine-tuned to obtain the appropriate region of interests (RoIs) to look for the possible existence of a vehicle in an input image. Therefore, we perform the mini-batch sampling by empirically choosing 128 region proposal networks (RPNs). To generate the RPN, a small network is made to slide over the *conv* feature map, which is output by the last shared *conv* layer. This small network takes as input an $N \times N$ spatial window of the input *conv* feature map. This feature is fed into two siblings' fully connected layers. We use $N = 2$ during our tuning, keeping in mind the fact that the effective receptive field on the input image is large. As a result, 64 RoIs are extracted from an input image. Moreover, to describe the foreground of an object mask, we choose an object proposal with an IoU overlap that contains at least 0.5 ground truth. In lines 10–16 of Algorithm 1, we process an RGB vehicle image with thirteen *conv* layers. As a result, a *conv* feature map (Ψ) is obtained.

Algorithm 1: Pseudocode of the vehicle detection method.

```

1. Input: colored RGB vehicle image
2.   begin Faster R-CNN
3.     initialize fine-tuning
4.     do
5.       extract features ► during training initialization
6.       perform mini-batch sampling by  $\left(\frac{RPN=128}{N=2}\right)$ ; 64 RoIs from each image
7.       select IoU overlap with ground truth > 0.5
8.       back-propagate errors across network layers ► weights optimization for nodes
9.     end
10.    for  $I \in \{R, G, B\}$  do
11.      process RGB data with 13 conv layers to obtain  $\Psi$ 
12.      generate the RPN by using 3 scales and aspect ratios on  $\Psi$ 
13.      feature map ( $\Psi$ ) and region proposals are fed to the RoI pooling layer ('I)
14.      'I  $\rightarrow (r, c, h, w)$ 
15.      for all feature vectors ( $\mathcal{J}$ ), generate the FC layer
16.    end
17.  end
18. Output: vehicle detection

```

For every region of the vehicle region proposal network (RPN), we apply nine diverse anchors on Ψ that calculate the probable vehicle regions. Meanwhile, for anchors, three scales are employed, which have resolutions of 128^2 , 256^2 , and 512^2 pixels along with three aspect ratios, which are 1:1, 1:2, and 2:1, respectively. As shown in line 14, max pooling is performed using five layers on $\left(\frac{h}{Height} \times \frac{w}{Width}\right)$ with a 7×7 of *Height* \times *Width* with h and w being the layer hyper-parameters that are independent of any particular RoI. Each RoI is expressed by a four-tuple (r, c, h, w) , which specifies its top-left corner (r, c) and its width (h, w) . Every feature vector (\mathcal{J}) is passed as an input into a sequence of fully connected (FC) layers. In between, the RoI pooling layer uses max pooling to transform the features inside a binding region into a small feature map. Moreover, only a few RPN proposals highly overlap with each other. Therefore, to reduce redundancy, we adopt non-maximum suppression (NMS) in the proposal regions.

We fix the IoU threshold for $NMS \geq 0.5$, which leaves us about 2000 proposal regions per image with a significant decrease in the number of proposals. After the NMS, the *top-N* ranked proposal regions are estimated to detect vehicles and draw a red rectangle around them. Once the vehicle is located in the input image, we apply our method to locate a LP within the bounding box that contains the vehicle.

3.2. The LP Localization

The detected vehicle, confined by a bounding box that is obtained in the last step of Algorithm 1, is nursed to the LP localization module that aims to detect the LP. Our developed LP localization method has a few interconnected steps. The LP localization method processes the RGB image and transforms it into the HSV components as shown in Equations (1)–(3).

$$H = \cos^{-1} \left[\frac{\frac{1}{2}[(R - G) + (R - B)]}{\sqrt{\frac{1}{4}[(R - G)^2 + (R - B)(G - B)]}} \right] \tag{1}$$

$$S = 1 - \frac{3}{R + G + B} [\min(R, G, B)] \tag{2}$$

$$V = \frac{1}{3} (R + G + B) \tag{3}$$

where H denotes hue, S represents saturation, and V stands for the value components of the transformed image. Our general observation is that an LP in actuality may have diversity and huge color variations. Considering this fact, in Algorithm 2, we introduce colors segmentation from lines 5–13 on each of the HSV components.

During our simulations, we empirically vary the HT_{low} value from 0.02 to 0.40 and HT_{high} from 0.409 to 0.620. Similarly, for the saturation and value channels, their relevant low and high thresholds are ST_{low} , ST_{high} , VT_{low} , and VT_{high} , respectively, as indicated in lines 8–11 of Algorithm 2. For ST_{low} , the values are changed from 0.370 to 0.500, whereas for ST_{high} , they are changed from 0.909 to 1.10. For the V channel in the HSV image, the VT_{low} is set to 0.750 and the VT_{high} is kept at 1.0. After these thresholds are set, the mask images are obtained for each of the H , S , and V channels. For the H channel, the H_{mask} is set to 1 when the H_{Image} obtained is greater than or equal to the low threshold and less than or equal to the high threshold. A similar mechanism is applied to obtain the masks of the S and V channels.

Consequently, a blob image (A) is attained, which is indicated in line (11), which is analyzed by using morphological operations to enhance LP blobs in a sample space (z). Here, dilation (\oplus) is applied using Equation (4), which enlarges the features and adds pixel layers across the regions of associated elements.

$$A \oplus B = \left\{ z \mid \left(\overset{\bullet}{B} \right) z \cap A \neq \phi \right\} \tag{4}$$

where B indicates a structuring element through which the blob image is dilated. Meanwhile, the closing (\bullet) operation is applied using Equation (5), in which the license plate blob image is first dilated by structuring element B and then eroded by B . The closing operation results in the smoothing of the contour and filling of the holes in the license plate blob.

$$A \bullet B = (A \oplus B) \ominus B \tag{5}$$

When the luminance is unsatisfactory, in Algorithm 2, we suggest illumination rectification as shown in lines 17–21. We use the PCA on the detected input vehicle image to fix the dimming of the image. By applying the PCA, we extract the Luminance and Chrominance channels of the RGB-colored vehicle image. In our work, only the luminance channel is processed further due to the fact that it contains a large amount of energy. After the mean of the luminance vector is calculated, we empirically estimate the low and upper limits of the threshold as shown in lines 19–21 in Algorithm 2. From lines 20–21, the luminance is adjusted to finally obtain the output image (X') with a much better luminance that can be handled later by the developed license plate detection module. We empirically estimate the low and upper limits of the threshold as shown in line (20) in Algorithm 2. We set the value of $threshold_{low}$ to 0.25 and $threshold_{high}$ to 0.95. From lines 21–22, the luminance is adjusted to obtain the final neat and clean enhanced output image (X').

Algorithm 2: License plate detection procedure.

```

1. Input: Vehicle image confined by bounding box
2.   For satisfactory luminance, do;
3.     begin LP Localization
4.       transform the vehicle-detected image to the HSV domain using Equations (1)–(3)
5.       do segmentation
6.         define HSV threshold limits for every channel
7.         obtain mask images
8.         If  $H_{Image} \geq HT_{low}$  and  $H_{Image} \leq HT_{high}$  then  $H_{mask} = 1$ 
9.         if  $S_{Image} \geq ST_{low}$  and  $S_{Image} \leq ST_{high}$  then  $S_{mask} = 1$ 
10.        If  $V_{Image} \geq VT_{low}$  and  $V_{Image} \leq VT_{high}$  then  $V_{mask} = 1$ 
11.        obtain the LP blob image ( $A$ )
12.        if  $HSV_{masks} = 1$ 
13.        end segmentation
14.        use mathematical morphology by Equations (4) and (5)
15.        analyze dimensions through aspect ratio and LP spatial area
16.        else
17.          use PCA and form a luminance vector
18.          calculate the luminance vector mean
19.          approximate  $threshold_{high}$  and  $threshold_{low}$ 
20.          If  $mean > threshold_{high} \rightarrow$  decrease the luminance,
21.          else If  $mean < threshold_{high} \rightarrow$  increase the luminance,
22.          Obtain improved output image ( $X'$ )
23.        end LP Localization
24. Output:  $I' =$  LP localization

```

Once the improved luminance image is obtained, the dimensions of the extracted regions are examined to locate the existence of a possible license plate. We analyze the dimensions of the license plate through its spatial area and aspect ratio. Finally, the LP module draws the green bounding box on connected regions, which outlines the existing LP in the image.

3.3. The LP Recognition

After a license plate is detected, normally the conventional LP identification methods segment the plate characters to recognize LP. These steps usually combine image processing techniques or video sequences, and their calculations depend on the true recognition rate and the error recognition rate. As discussed earlier, LP recognition is a difficult task due to the huge variety of plate formats and severely varying outdoor illuminations during the image acquisition phase. Many methods perform well in standard circumstances, for instance, controlled illuminations, restricted vehicle speeds, prespecified roads, and static backgrounds. Several algorithms have been designed to achieve LPR in images. In addition, issues such as processing time, computational complexity, and recognition rate are also important parts of the LPR algorithm. Algorithm 3 shows the pseudocode of the proposed LP recognition algorithm.

As can be seen in Algorithm 3, our developed method contains interconnected steps and performs miscellaneous operations after the LP bounding box is fed to the recognition module. Since the area contained by the LP is normally small, for better visibility, contrast is enhanced using contrast from basic image processing methods. The improved contrast image is binarized and segmented by applying the morphological operations using Equations (4) and (5), respectively.

Algorithm 3: The LP recognition pseudocode.

```

1.  Input: LP bounding box
2.      begin operations
3.          Enhance contrast and deblur the image for better visibility
4.          Binarize the image obtained in the above steps
5.          Obtain segmented image (S) through dilation and erosion using Equations (4) and (5)
6.          Get Pre-trained model
7.          do
8.              for S = 1:n
9.                  Perform prediction on S
10.                 Build output string
11.             end
12.         end operations
13.     end operations
14.  Output: Recognized LP characters

```

On the basic pretrained model, the LP characters are predicted to build the possible LP strings that may appear inside the LP bounding that was processed in the initial stage of the LP recognition module. Algorithm 3 generally depicts the core theme of the LP recognition scenario. All the operations used herein, such as contrast, deblurring, and binarizing the image, are essentially handy for the recognition task.

4. Simulation Results

To simulate, we use a workstation, which has one NVIDIA RTX 2070 GPU along with an Intel CPU-Corel i7-6700. Simulations are done in Python version 3.6.0. Below, we discuss in detail the performance of our proposed LP recognition algorithm.

4.1. Training Data Preparations and Model Training

Before our developed method is executed, we initially prepare the data and make some assumptions to train the model. Algorithm 4 shows the arrangements for preparing the training data. To extract the LP digits from the input image, basic data processing (DP) operations are performed from lines 2–13 of Algorithm 4. Most of these DP operations include desaturating the image through a Gaussian low-pass filter and binarizing the image. Moreover, the erosion and dilation operations described above are also performed. Meanwhile, the LP image is converted to 28×28 pixel image on which random spatial transformations are applied that ultimately result in a 28×28 dataset with prominent characters and their classes.

Since then, we have also performed experiments on the CCPD dataset, which has substantial license plate variations, such as tilted or blurred plates. For the tilted plates, spatial transformations are applied to the 28×28 pixels converted image. This operation essentially corrects the appearance of the license plate and ultimately makes the algorithm easier to process. Similarly, for poor image quality in which characters are not fully visible, characters touch each other due to blur or similar phenomena. In such conditions, mathematical morphological image processing techniques, such as erosion and dilation as described in Equations (4) and (5), respectively, become handy. All the operations listed in Algorithm 4 essentially prepare and result in well-managed, systematic data that is nicely processed by our developed algorithm during the recognition task.

Algorithm 4: Training data preparations.

```

1.  Input: Single digits extracted from LPs
2.      begin DP
3.          Extract Single Digit
4.          Use desaturate
5.          Use De-Blurring
6.          Binarize the image
7.          Erode Image
8.          Dilate Image
9.          Convert to  $28 \times 28$  image = (i)
10.         for s = 1:random (n)
11.             Perform Random Spatial Transform on (i)
12.             Save the image (i) and the character class to a dataset
13.         end
14.     end DP
15.  Output: Dataset of  $28 \times 28$  resolution with character classes

```

After LP character data is obtained, in the next step, training of the LP recognition model is performed as shown in Algorithm 5, which takes the 28×28 LP character image and yields the recognition model with weights. During the first part of the LP training, a 13-layer CNN is used to build a DNN. This DNN is then applied to a 3×3 Conv2D layer along with a 2×2 MaxPool layer. As shown in Algorithm 5, the next stages also apply a dense layer to perform the 50% dropout to obtain the appropriate model. During the model training, the LP characters are checked and predicted for a small batch of images. Meanwhile, to obtain good accuracy, weights are adjusted at regular intervals after each execution epoch. Once the training data and LP recognition model training are set, in the next section, we demonstrate our detailed observations and findings. Our LP recognition analysis and discussion are based on the PKU, AOLP, and CCPD datasets, which are well-known and widely used in research these days.

Algorithm 5: The LPR model training.

```

1.  Input: Dataset of  $28 \times 28$  Images with character classes
2.      begin LPR Training
3.          begin Model Design
4.              Apply the DNN with 13 Layers of the CNN
5.              for i = 1:3
6.                  Conv2D  $3 \times 3$ 
7.                  MaxPool  $2 \times 2$ 
8.              end
9.              Flatten the LP with a dropout of 50%
10.             for i = 1:3
11.                 Use a dense Layer with a dropout of 50%
12.             end
13.          end Model Design
14.          begin Training
15.              for epoch = 1:n, get batch of images (i)
16.                  for i = 1:n
17.                      Provide image to the model and check predicted characters adjust weights
18.                  end
19.              end
20.          end LPR Training
21.  Output: Model with set weights

```

4.2. Analysis of the PKU Dataset

During our study, we initiated our experiments on the PKU dataset, which is a well-known publicly available vehicle dataset. Table 1 briefly describes the various vehicle categories in the PKU dataset. Generally, the PKU dataset is a collection of diverse vehicle images that are captured under diverse conditions [31]. As shown in Table 1, this dataset contains a total of 3977 diverse vehicle images. The developers of the PKU dataset divided the vehicles into five distinct categories, which they refer to as G1, G2, G3, G4, and G5. Out of 3977 vehicle images, the PKU dataset also contains a total of 4263 visible license plates, whose pixel resolution varies from 20 to 62 pixels.

Table 1. The PKU dataset description.

Category	Vehicle Conditions	Input Image Resolution (Pixels)	No. of Images	No. of Plates	Plate Height (Pixels)
G1	Cars on roads; ordinary environment at different daytimes; contains only one license plate per image	1082 × 728	810	810	35–57
G2	Cars/trucks on main roads at different daytimes with sunshine; only one license plate in each image	1082 × 728	700	700	30–62
G3	Cars/trucks on highways during the night; one license plate per image	1082 × 728	743	743	29–53
G4	Cars/trucks on main roads; daytimes with reflective glare; one license plate in input images	1600 × 1236	572	572	30–58
G5	Cars/trucks at roads junctions with crosswalks with several plates per image	1600 × 1200	1152	1438	20–60
Complete PKU dataset			3977	4263	20–62

In Figure 3, we demonstrate a few detection results for both vehicles and license plates for each category of the PKU dataset. We show different vehicles from each category to demonstrate a fair understanding.

Vehicle+LP detection: G1-category: The first row in Figure 3 demonstrates a few images from this category. It is evident for this category that for different-shaped vehicles, the detection module performs well by drawing a red rectangle around the object of interest, which is a vehicle in this case. The detected vehicle image is then analyzed by the LP localization module. In all four of the sample images in Figure 3 from the G1 category, the visible LP is accurately localized by our developed method.

Vehicle+LP detection: G2-category: The second row in Figure 3 demonstrates a few images from the G2 category. As indicated in Table 1, this category mostly contains vehicle images that are captured during different times of the day. In all four images shown for this category, both the vehicle and the LP localization module are in the correct position, thereby indicating the correct position of both of these objects. The first image shown for this category is of the truck, and the rest are the cars. However, the detectors applied to capture these objects are intelligent enough to discriminate between these shapes.

Vehicle+LP detection: G3-category: The third row in Figure 3 demonstrates a few images from the G3 category. Most of the images in this category are nighttime captures of small cars and trucks. It can be observed in the third row of Figure 3 that both objects are accurately located. To fairly discriminate the vehicle and the LP detection for nighttime captured images, we draw the white color bounding box around both the detected vehicle and the LP area.

Vehicle+LP detection: G4-category: The 4th row in Figure 3 demonstrates a few images from the G4 category. This category also contains one license plate in an image, but those are captured in a difficult situation of reflective glare that affects the image quality and the LP area appearance. However, in this case, our applied object detectors handle them efficiently. For each of the different images shown in the fourth row of Figure 3, the good performance of the applied detectors to localize both vehicles and the LP of that vehicle is evident.

Vehicle+LP detection: G5-category: The last row in Figure 3 demonstrates a few images from the G5 category. This category contains a few LPs in an image. As shown in the last row of Figure 3, all the instances of object detection are completely achieved. In particular, the first image in the fifth row shows the object from an angle, which is also correctly spotted by the applied detectors. The rest of the images in this row contain at least two vehicles along with two LPs that are accurately detected.



Figure 3. Cont.

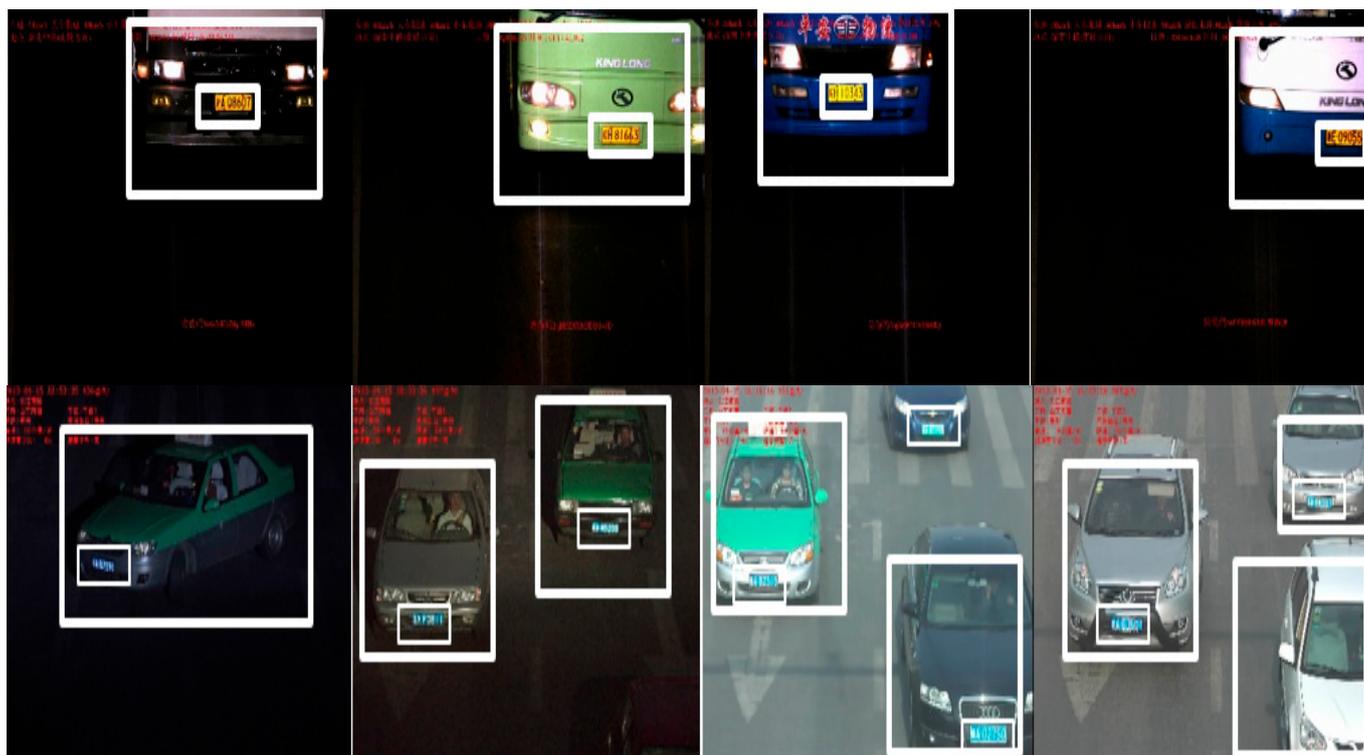


Figure 3. Vehicle+LP detection on PKU dataset.

Table 2 lists the comparison of each category of the PKU dataset for various methods mentioned therein. A few of the important findings from Table 2 are summarized below.

- Each of the compared methods along with our developed method yields 100% vehicle detection accuracy in the G1 and G2 categories, except the work developed in [33], whose vehicle detection accuracy is 99%. Similarly, for the G3 category, the method developed in [33,34] yields 98.20% and 99% vehicle detection accuracy. The remaining approaches all produce 100% vehicle detection results.
- In the G4 category, all of the methods compared can find vehicles with an accuracy of at least 99%. In this category, YOLO-v7-based methods [46,47] yield the highest vehicle detection accuracy of 99.74% and 99.72%, respectively. While for the G5 category, an improved YOLO-v7-based method ranks first, yielding 99.22% vehicle detection accuracy. Our developed method ranks 3rd and yields a vehicle detection result at par with [46] by delivering 99.10% detection accuracy.
- On the PKU dataset to locate vehicles, an improved YOLO-v7-based method ranks first and yields a mean vehicle detection accuracy of 99.79%, followed by standard YOLO-v7 [46], whose accuracy is 99.76%. Our developed method also yields approximately similar results as compared with [46]. Vehicle detection is a prototype in our developed system. Therefore, an accuracy of slightly over 99.75% is very encouraging in the later stages of the algorithm.
- Table 2 also lists the LP detection comparisons for several methods. As can be seen, the improved YOLO-v7 [46] ranks first in all five categories of the PKU dataset in terms of LP detection. The standard YOLO-v7 method [45] ranks 2nd in terms of license plate localization on this dataset. For the G1 and G2 categories, all of the methods compared had a LP detection accuracy of at least 97%, whereas, for the G3 category, the methods listed in Table 2 yielded at least 98% LP detection. For the G4 category, approximately 99% LP detection is achieved. The G5, which is the most challenging category in the PKU dataset, is also addressed nicely. In this category, the methods listed in Table 2 yield at least 98% accurate license plate detection. In addition, our method yields at least 99% LP detection for G1, G2, G3, and G4 categories. In the G5 category,

our finely tuned version of the Faster RCNN achieves 97.30% accurate license plate detection accuracy.

- Our analysis indicates that the mean LP detection accuracies of the works [32–34,45–47] are found to be 98.47%, 98.06%, 98.47%, 99.09%, 99.05%, and 99.13%, respectively. The aforementioned LP detection accuracies are a good indicator that all the compared methods yield at least 98% license plate detection accuracy. YOLO-based methods [45–47] perform well to locate an object, such as a vehicle or license plate. However, from Table 2, we find that our method, which employs a fine-tuned version of the Faster RCNN, yields a mean license plate detection accuracy of 99.04%. The aforesaid analysis is a good indicator of the application of the various methods to achieve objects, such as vehicles and license plates, in various real-life applications. Vehicle and license plate detection is a prototype of our developed system. Therefore, our deployed detectors also yields at par results with the recently published works.

Table 2. Category-wise Vehicle + License Plate detection comparison (%) on PKU dataset.

Object	Ref	PKU Dataset Categories				
		G1	G2	G3	G4	G5
Vehicle	[32]	100	100	100	99	98.50
	[33]	99	98	98.20	99.10	98
	[34]	100	100	99	99.10	98
	[45]	100	100	100	98.96	99.13
	[46]	100	100	100	99.72	99.10
	[47]	100	100	100	99.74	99.22
	Proposed	100	100	100	100	99.70
License Plate	[32]	99	97.05	98.80	99	98.50
	[33]	97	98.01	98.20	99.10	98
	[34]	98.50	98.22	98.55	99.10	98
	[45]	98.80	99.45	99.15	98.96	99.13
	[46]	99.85	99.50	99.22	99.35	97.35
	[47]	99.87	99.65	99.40	99.40	97.35
	Proposed	99.81	99.50	99.20	99.40	97.30

With the state-of-the-art method listed in Table 2, detection accuracy is almost at par with that of conventional methods. After the objects, which in our case are vehicles and LPs, are located, in the next phase we process the detected LP area for recognition. It is important to state that in the PKU dataset, all the visible license plate labels are not annotated. Therefore, in the PKU dataset, we labeled the 2250 images. The 1355 images are randomly selected for training, and the other 901 are used for testing. To evaluate license plate recognition accuracy, the license plate was localized by a bounding box as shown in Figure 3 for each category of the PKU dataset. The detected license plate is now fed to our newly developed recognition module.

As shown in Figure 4, the proposed LP recognition technique correctly understands different LPs that appear in each of the five categories of the PKU dataset. The important points noted during the LP recognition task are discussed further below.

LP recognition: G1-category: As shown in the first row in Figure 4, the proposed recognition algorithm precisely identifies the LPs shown therein. Our obtained correct recognition result is shown on top of the original LP on the input vehicle images. The third image in the first row of Figure 4 has a relatively complex background. However, it does not pose any threat to the proposed method of achieving the correct identification result.

LP recognition: G2-category: As shown in the second row in Figure 4, the first three images have different car colors with their own installed LPs. Our method correctly identifies all such cases. However, the fourth image of the bus with visible LP has a relatively complex background. Nevertheless, the proposed method handles this scenario well and achieves the correct result on top of the original LP shown therein.

LP recognition: G3-category: As shown in the third row in Figure 4, the proposed LPR method reliably handles the high-glare images. The LPs on the vehicles in the first two pictures in this row are clear enough to be correctly identified. Similarly, it is clear from this row that our developed method handles low-contrast images in which both the vehicles and the background have blackish appearances. Generally, it is observed in the third row of Figure 4 that our developed method has barely any effect on its recognition performance with blackish objects against a black background.

LP recognition: G4-category: As shown in the fourth row in Figure 4, the area around the vehicles is highly dark. There also appear to be glare and high beams from vehicles. However, in all four images shown for this category in Figure 4, our developed method accurately identifies all the LP numbers and successfully handles the glare situations.

LP recognition: G5-category: As shown in the fifth row in Figure 4, there appear to be multiple vehicles and LPs in the images. For all the images shown, our developed method identifies all the LP that appear in the images. In the second and fourth images, there appear to be three LPs. In the fourth image, our method identifies all three LPs, whereas, in the second image, only two LPs are detected out of three. One reason is the red text that appears in the input image around the LP area, which created a hurdle for our developed method.



Figure 4. Cont.

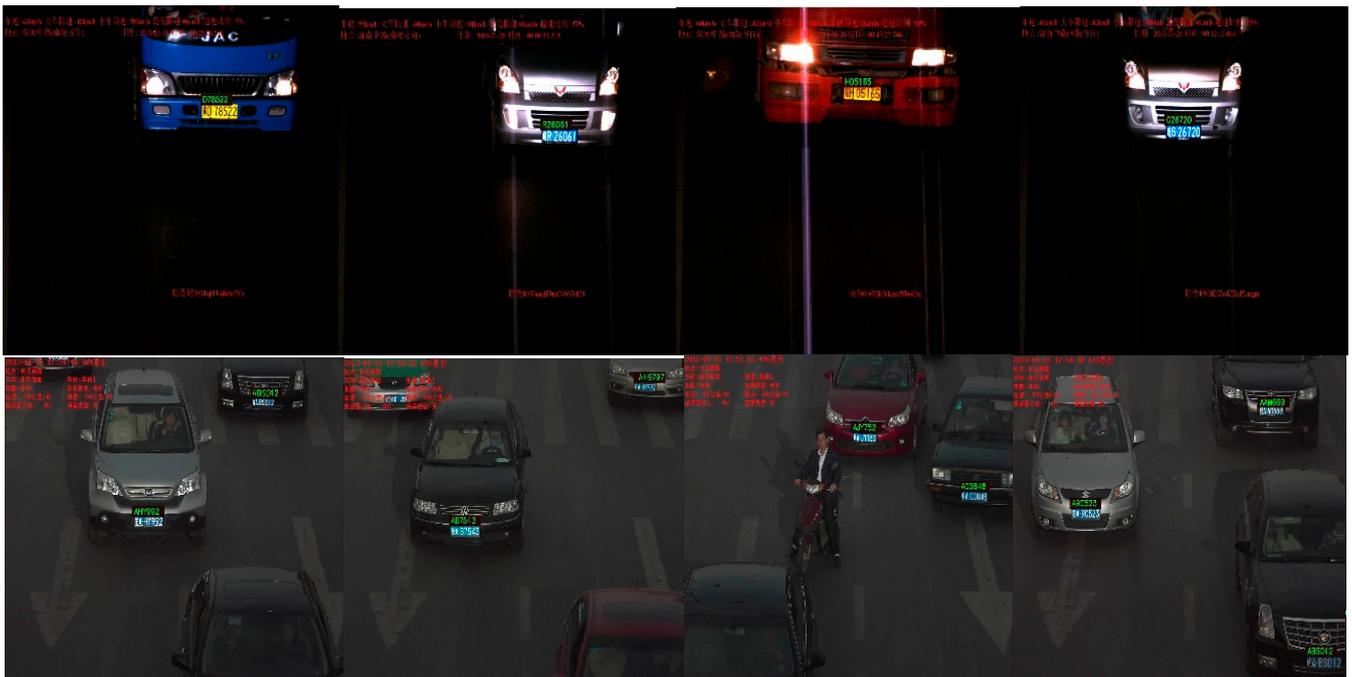


Figure 4. The LP recognition on different images of the PKU dataset.

Table 3 lists the LP recognition rate for each of the PKU categories for works developed in [32–34]. It is important to state here that these methods were chosen for comparison on the PKU dataset because their standard implementation is publicly available. This makes it logical to train these models on the PKU dataset along with our developed method.

Table 3. Category-wise LP recognition accuracy comparison (%) on PKU dataset.

Ref	G1	G2	G3	G4	G5
[32]	96	97.80	92.60	80.00	72.00
[33]	92.00	90.50	90.10	89.60	86.40
[34]	98.00	98.50	90.00	86.01	81.10
Proposed	100	100	100	99	99.63

From Table 3, it is evident that all the compared methods yield over 90% recognition accuracy for the G1 category. Our developed method yields 100% LP recognition accuracy in this category. The work developed by Zhang et al. [34] ranks second, yielding 98% accurate recognition. For the G2 category, the work developed by Masood et al. [33] yields the lowest LP recognition accuracy of 90.50%. In this category, our developed method ranks first, followed by the work reported in [34]. For the G3 category, the works on [33] and [34] yield almost similar results by producing at least 90% recognition accuracy. In this category, the work developed by Xu et al. [32] also reports 92.60% LP recognition. For the G4 category, we observe that works in [32–34] yield below 90% LP recognition. In this category, our developed method comprehensively outperforms the compared methods. For the G5, which is the most difficult category of the PKU dataset, the work in [32] produces the least accuracy of 72%, followed by [34], whose accuracy is a bit over 80%. In this category, our developed method yields 99.63% recognition accuracy.

In Figure 5, we report the mean license plate recognition accuracy on the PKU dataset. Our proposed LPR method comprehensively beats the compared methods in terms of mean recognition accuracy. As shown in Figure 5, our developed license plate recognition method yields 99.63% accuracy on the PKU dataset. Similarly, the work proposed in [34]

ranks second, yielding 90.72% accuracy. On the aforementioned dataset, the work proposed in [32] yields the lowest license plate recognition accuracy of 79.28%. To the best of our knowledge, on the PKU vehicle dataset, the proposed method has almost solved the LPR accuracy problem.

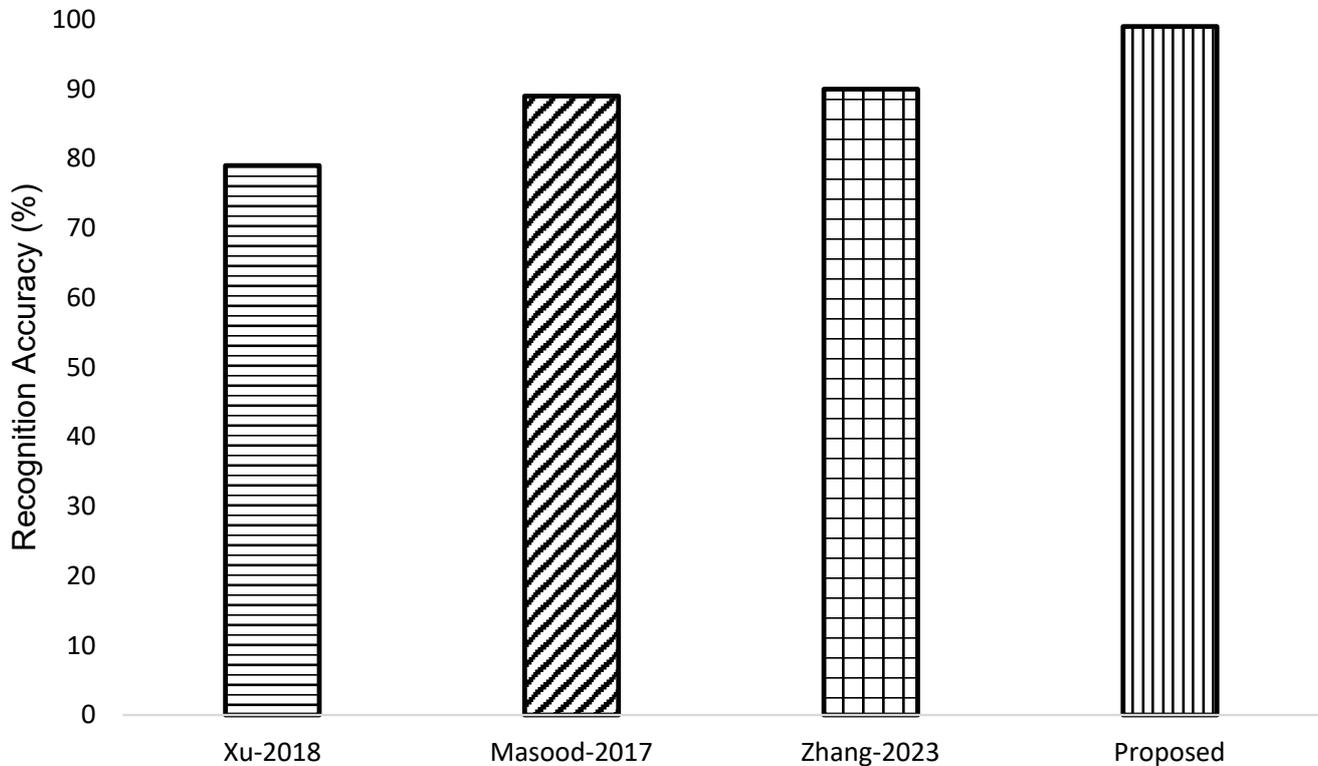


Figure 5. Mean LP recognition comparison on the PKU dataset.

4.3. Analysis of the AOLP Dataset

The application-oriented license plate (AOLP) [35] database consists of 2049 images of a Taiwan license plate. This dataset is categorized into three subsets according to complexity levels and photographing conditions, which are access control (AC), traffic law enforcement (TLE), and road patrol (RP). For the readers' information, below we briefly describe the categories contained in the AOLP dataset.

Access Control (AC): The AC refers to the cases in which a vehicle passes a fixed passage at a reduced speed or with a full stop, such as at a toll station or the entrance/exit of a region.

Traffic Law Enforcement (TLE): The TLE refers to cases where a vehicle travels at a regular or higher speed but violates traffic laws, such as a traffic signal or speed limit, and is captured by a roadside camera. Here, 757 images were collected for this application category.

Road Patrol (RP): The RP refers to the cases where the camera is installed or handheld on a patrolling vehicle and takes images of the vehicles from arbitrary viewpoints and distances. Since we do not have any other images with Taiwan license plates, we use any two of these subsets for training and the remaining one for testing, similar to previous practices. Figure 6 shows our obtained results on the AOLP dataset.



Figure 6. The LPR on the AOLP dataset is organized under (a) Access control, (b) Traffic law enforcement, and (c). Road patrol conditions.

Figure 6 shows the license plate recognition results for each of the aforementioned categories of the AOLP dataset. The proposed LPR method works well for scenarios where half of the vehicle bonnet is visible along with the license plate location, which is much lower on the horizontal axis. In each of the images in Figure 6a for the AC category, the proposed method accurately identifies the license plates. Similarly, for the TLE category, as shown in Figure 6b, where the license plates appear in the angular view, the proposed method accurately handles this angle variation by correctly identifying all the license plates. The third image in Figure 6b is especially interesting, as here a yellow vehicle appears at the back side of the license plate, which ultimately results in the partial occlusion of the license plate. Although it does not affect the digits of the plate area, the proposed LPR algorithm handles this partial occlusion and accurately identifies the license plate. Figure 6c shows the RP conditions. Clearly, this is a challenging category as there appears to be a large angle deviation of the viewpoint of the license plate, which makes this scenario challenging for most of the machine learning algorithms. However, as can be seen in Figure 6c, the proposed LPR method reliably handles this issue by indicating the correct number on the license plate.

Table 4 lists the LP recognition rate on different classes of the AOLP dataset for works developed in [36–39]. It is important to state here that these methods were chosen for comparison on the AOLP dataset because their evaluations on this dataset, along with standard implementation, are publicly available. This makes a fair reason for us to train these methods on the AOLP dataset along with our developed method. Table 4 also lists the comparison of the proposed LPR with a few recent methods on the AOLP dataset. As can be seen in Table 4, for the AC category, the proposed method yields the highest recognition rate of the license plates in this category.

Table 4. Comparison of the AOLP dataset.

Method	Accuracy % on Each Category			
	AC: No of Images = 681	TLE: No of Images = 757	RP: No of Images = 611	Mean Recognition Accuracy %
[36]	94.9000	94.2000	88.4000	92.5000
[37]	95.3000	96.6000	83.7000	91.8666
[38]	96.6000	97.8000	91.0000	95.1333
[39]	97.3000	98.3000	91.9000	95.8333
Proposed	97.8970	98.2719	91.9006	96.0231

Moreover, in the AC category, the work proposed in [39] ranks second among the compared methods. Similarly, for the TLE category, the proposed LPR method ranks second on the AOLP dataset. In this category, the work reported in [39] yields the highest recognition accuracy. However, the work in [36] ranks fourth among all compared methods, yielding slightly over 94% identification accuracy. For the RP category, the method reported in [39] and the proposed method yield almost similar identification accuracy of slightly over 91%, despite the fact that the proposed method is a bit higher. As indicated by the last column in Table 4, the proposed license plate recognition method yields the highest license plate recognition accuracy of 96.0231% on the AOLP dataset. The work listed in [39] ranks second in achieving overall identification accuracy, followed by [38]. In general, and across the whole AOLP dataset, all of the methods compared correctly identify license plates over 91% of the time.

4.4. Analysis of the CCPD Dataset

The CCPD dataset [40] is the largest publicly available LP dataset and has a collection of over 290,000 Chinese LP images. This dataset is separated into several categories according to the difficulty of identification, for instance, the illuminations on the LP area, the distance from the license plate when photographing, and the degree of horizontal and vertical tilts. The CCPD dataset also contains images in different weather conditions, such as rainy, snowy, or foggy. Each category includes 10,000 to 20,000 images. The CCPD-base consists of approximately 200,000 images, of which 100,000 are used for training and the other half are for testing. As listed in Table 5, the other sub-datasets, such as the CCPD-DB, the CCPD-FN, the CCPD-rotate, the CCPD-weather, and the CCPD-challenge, are also used during the test phase.

Table 5. Comparison of the CCPD dataset.

Model	CCPD-Base (100 k)	CCPD-DB (20 k)	CCPD-FN (20 k)	CCPD-Rotate (10 k)	CCPD-Tilt (10 k)	CCPD-Weather (10 k)	CCPD-Challenge (10 k)	Overall Accuracy (%)
[40]	98.5000	96.9000	94.3000	90.8000	92.5000	87.9000	85.1000	95.5000
[41]	99.1000	96.3000	97.3000	95.1000	96.4000	97.1000	83.2000	93.0000
[42]	99.5000	98.1000	98.6000	98.1000	98.6000	97.6000	86.5000	98.3000
[43]	98.9000	96.1000	96.4000	91.9000	93.7000	95.4000	83.1000	96.6000
[44]	99.6000	98.8000	98.8000	96.4000	97.6000	98.5000	88.9000	98.5000
Proposed	99.8500	98.7800	98.8000	98.1100	98.8000	98.9000	88.8000	98.7000

Figure 7a shows a few samples of the output images on the CCPD-base images. Clearly, the proposed method performs well on all images. Particularly, the left-most image has huge illumination variations with very limited visible contrast in the license plate area. The proposed method handles that scenario well and correctly identifies the license plate. Similarly, the second, third, and fourth images in the top row of Figure 7a are the cases

where the license plate appears in the angular view. However, our proposed method handles this scenario and identifies all the license plates. Figure 7b shows the CCPD-blur image output of our developed method. Most of these blurred images were captured in outdoor conditions with strong sunlight and complex backgrounds. Since these images appear blurry, the license plate area has a low resolution. However, it can be seen in the second row of Figure 7b that our developed method performs significantly well and identifies all the license plates shown therein in the second row. Particularly, the first and third images in the second row of Figure 7b are indicative of the good performance of our developed method where the background is complex along with various other objects. Moreover, the third row in Figure 7c is the sample output of our proposed method for the CCPD-FN cases. Clearly, in this case, our developed method is quite accurate and reliably identifies all the license plates shown therein. It is to be noted that the third row in Figure 7 also contains complex backgrounds. However, the good performance of our developed method is unaffected by these factors.

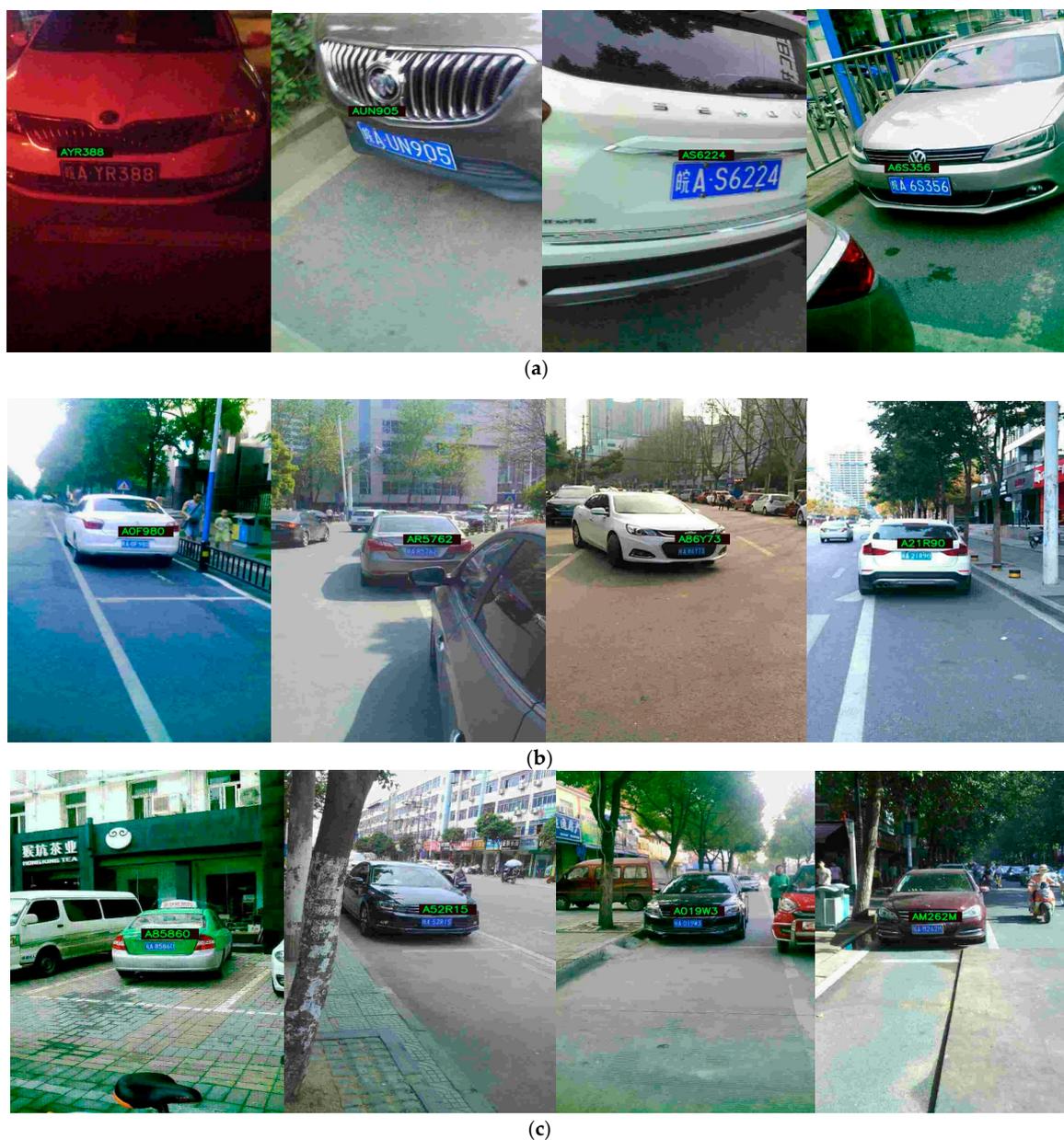


Figure 7. The LPR on the CCPD dataset for (a) base, (b) blur, and (c) FN scenarios.

A more detailed analysis of our developed method is shown in Figure 8. As can be seen, the outputs in the first row of Figure 8 are from the CCPD-rotate category. Particularly, the first image has a rotated license plate along with an overly whitish appearance due to the presence of very strong sunlight. Clearly, the developed method handles such a scenario and accurately identifies the license plate. The fourth image in the first row of Figure 8 has a relative combination of dark and bright contrast. Overall, the proposed method performs well in the CCPD-rotate category and, as seen in Table 5, produces encouraging results. The second row in Figure 8 shows the license plate identification resultant images from the CCPD-tilt category. The first image in the third row of Figure 8 is a low-contrast image example that has severe black contrast. It can be seen that our developed method is unaffected by this situation and accurately identifies the license plate. Similarly, the last image in the third row of Figure 8, which has a slightly misplaced license plate, is highly challenging in the tilt category. However, our developed method also handles this case intelligently and produces accurate output.

More output resultant images from the CCPD dataset are shown in the third row of Figure 8, where a few cases are shown for the different weather conditions. The first three images in the third row of Figure 8 correspond to the snowy weather where our developed method reports accurate results, whereas the fourth image is for the rainy day in which our developed method performs at par and yields accurate results. The fourth row in Figure 8 is for the outputs generated by the algorithm for the CCPD challenge category.

During simulations, we find that this is the most challenging category in the dataset, and it is not easy for every algorithm to handle this. The first image shown in the last row of Figure 8 indicates that both the vehicles and the outside environment are severely dark. However, our developed method handles this scenario and yields accurate recognition results. The same is true for the third image in the last row, where our approach accurately identifies and identifies the license plate. Similarly, for the 2nd image in the last row of Figure 8, there appears to be a shadow on the road and the vehicle, and there is also a bright light in the center of the license plate. However, our developed method passes through this hurdle and yields the correct result. Likewise, the rightmost bottom image in Figure 8 is the case where there are back lights turned on, and half of the license plate has a blue background with white color text on it while the other half has a light grey background with yellow text over it. Consequently, our established approach delivers accurate and encouraging results in this case.

Table 5 lists the LP recognition rate on different classes of the CCPD dataset for works developed in [40–44]. It is important to state here that these methods were chosen for comparison on the AOLP dataset because their evaluations on this dataset, along with standard implementation, are publicly available. This makes a fair reason for us to train these methods on the AOLP dataset along with our developed method. In Table 5, we show the comparison of our developed method with these five techniques on the CCPD dataset for all the categories. It can be seen that, for the CCPD-Base category, our method ranks second out of all the compared methods. In this category, the work reported by [44] has the highest accuracy. In this category, the work conducted in [40] has the least recognition accuracy. For the CCPD-DB category, our method follows [44] and lies in the second position. Here, the work in [41] has the lowest accuracy. For the CCPD-FN category, our method and [44] have the highest license plate recognition accuracy, followed by the work done in [42].

For the CCPD-rotate category, our developed technique beats the compared works and yields the highest identification accuracy of license plates. In this category, the work done in [43] yields the lowest identification rates. Moreover, for the CCPD-Tilt category, our method has the highest recognition accuracy, followed by the work in [40], which has the lowest reported license plate identification rate. For the CCPD-weather category, our method again beats the compared works. Here, the work in [40] has the lowest recognition rates. Furthermore, for the CCPD challenge category, the work presented in [44] has the highest license plate identification rate and [41] has the lowest. In this category, we again

rank second out of the compared methods. Our developed method yields the highest overall license plate recognition accuracy, with a 98.7000% correct recognition rate. The work performed in [44] ranks second, and the work reported in [40] lies in the third spot. Overall, the work reported in [41] has the lowest license plate identification rate of 93%.



Figure 8. The LPR on the CCPD includes (a) rotate, (b) tilt, (c) weather, and (d) challenge scenarios.

4.5. Computational Complexity

To perform the computational analysis of our developed method and the compared methods, we manually cropped the image resolutions. In our experiments, we selected different image resolutions, which are 700×1100 , 500×800 , 400×600 , 320×240 , and 300×280 pixels. From the compared works in this manuscript, we choose ten methods and executed them on the aforesaid image resolutions. Complete results are detailed in Figure 9. It is evident that the work reported by Yu et al. [41] and Li et al. [36] is computationally complex and consumes more than 3 ms to process the image resolution of 700×1100 to yield the final recognition result. Moreover, the works of Yuan et al. [34], Luo et al [42], and Wang et al. [43] also consume more than 2.5 ms to process the aforesaid image resolution to generate the final resultant image. The works reported in Masood et al. [33] and Wu et al. [38] are computationally efficient and consume nearly 0.5 ms to process the test image for various image resolutions. Therefore, we observe that our developed technique takes slightly over 2 ms to deliver the final result. In terms of the execution time ranking, our developed method ranks fourth out of all of the compared methods. We observe that all the compared methods are near real-time for processing various image resolutions. Once an algorithm is trained on every dataset, our developed method along with other methods can be used in a resource-constrained environment, as we see that all the methods explored in this study work in near real-time in actual living environments with high accuracy.

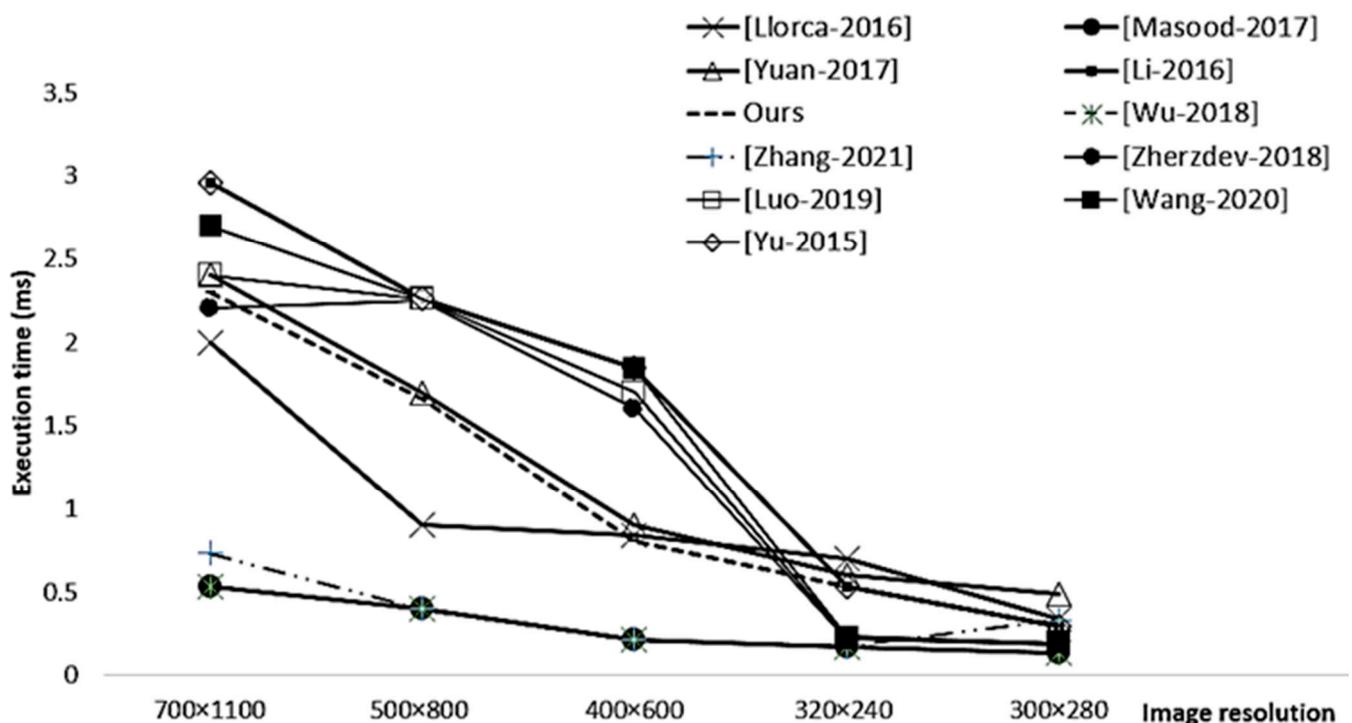


Figure 9. Computational complexity [32–44].

4.6. Discussion

Detailed simulations shown in this paper indicate that object detection, such as vehicle or license plate detection, has been an active research field in recent years. This paper presented a detailed analysis of license plate recognition on three publicly available datasets. For the task of vehicle detection, a Faster RCNN architecture was used. The license plate was located and recognized through our own developed methods. Our findings are indicative of superior outputs on challenging datasets. Moreover, a detailed comparison of our developed method was carried out with several state-of-the-art license plate recognition approaches. We are optimistic that this study will be a fair guideline for

beginners and practitioners to modify or use any detector or recognizer for their desired tasks or applications. The outcomes of our developed system for recognizing license plates are summarized below.

PKU Dataset:

On this dataset, our developed method yielded 100% recognition accuracy in the G1, G2, and G3 categories. In the G4 category, our developed method was 99% successful at accurately recognizing the license plate. Finally, in the G5 category, our developed method yielded 99.63% recognition accuracy. Overall, on the PKU dataset, our developed method ranks first out of the three compared methods in terms of license plate recognition accuracy.

AOLP Dataset:

This dataset contains three challenging categories, which are access control, traffic law enforcement, and road patrol. On access control, our developed method yielded 97.8970% accurate recognition accuracy and ranked first herein. On traffic law enforcement, our developed method yielded 98.2719% license plate recognition accuracy and ranked second among the compared methods. On the road patrol category, our developed method generated a mean recognition accuracy of 91.9006% and ranked first among the four compared methods. The whole-mean accuracy on the AOLP dataset by our developed method is 96.0231%.

CCPD Dataset:

This is the largest publicly available license plate dataset and contains challenging scenarios, such as blur, rotation, tilt, and varying weather. On this dataset, our developed method yielded a mean recognition accuracy of 98.7000% and ranked first among all compared methods. In general, for all the other aforementioned categories, our developed method yielded over 98% recognition accuracy. However, for the CCPD challenge category, our developed method yielded slightly over 88% recognition accuracy and ranked second among the five compared methods.

4.7. Limitations

As with any other algorithm for machine learning, we discovered several shortcomings and failures in our method. Figure 10 depicts a handful of these instances with the following observations:



Figure 10. Few failures cases examples.

- It is clear from the rightmost image in the first row of Figure 10 that the input image is extremely blurry with a non-clear license plate. In such a case, our developed method struggles to distinguish the actual words and reads “A” from the license plate as “0”.
- Similar is the case for the next two images in the first row of Figure 10. We also observe that there is no specific rule for license plate fonts. Therefore, such cases are very hard to identify correctly. As shown in the first image in the second row of Figure 10, the extreme blur is also a very challenging situation for any algorithm to deal with.
- We observe that occlusion, either partial or full, is also a challenging factor for the machine learning-based license plate identification method. One such case is shown in the third image in the second row of Figure 10, where high intensity light beams have created occlusion in the license plate area and thereby a hurdle for the algorithm to handle with. Therefore, before processing the license plate, such factors should be carefully analyzed.
- We also note that light that falls on the license plate area due to reflection from the vehicle’s surface also reduces the recognition ability of the algorithm. One such case is seen in the middle image of the second row in Figure 10. Therefore, before a test license plate is fed to the recognition algorithm, this issue should also be noted. In such cases, an image enhancement or contrast rectification method might be useful to improve the quality of the appearance of the license plate.

5. Conclusions

Accurate detection and recognition of vehicle license plates in natural scene images is an important task to be performed by machine learning algorithms. Nowadays, it is an integral part of modern intelligent traffic control systems. However, this task is quite challenging due to various factors, for instance, the non-uniform patterns of the plates, variations in view angles, such as blurriness, and the occlusions. With such factors, it is always difficult for a single algorithm to handle the aforesaid issues. This paper discussed methods to detect and identify license plates that appear in an image. The proposed method is composed of three distinct but interconnected steps: (i) vehicle detection, (ii) license plate detection, and (iii) license plate recognition. To locate the vehicles, a fine-tuned version of the Faster RCNN was used, while the license plate area was located through our own developed plate localization module. Finally, the recognition task is achieved using the deep learning network. Simulations were performed on three databases, which are the PKU, the AOLP, and the CCPD license plate dataset. Our proposed method achieves competitive performance and yields 99%, 96.0231%, and 98.7000% recognition rates on the aforesaid datasets. We are optimistic that our findings are promising and will be applicable to a variety of real-world applications, including surveillance and the monitoring of suspicious vehicles.

In the future, the proposed method could be modified to handle extreme blurriness. Similarly, the proposed method could also be improved to handle occlusion. Moreover, the developed algorithm could also be made intelligent by being trained in parallel over various time intervals.

Author Contributions: Conceptualization, F.S. and Z.M.; methodology, F.S.; software, F.S., Y.A.S., K.K., M.S., U.K. and Z.M.; validation, F.S.; formal analysis, F.S., Y.A.S., K.K., M.S., U.K. and Z.M.; investigation, F.S. and Z.M.; resources, F.S., Y.A.S., K.K., M.S., U.K. and Z.M.; data curation, F.S.; writing—original draft preparation, F.S., Y.A.S., K.K., M.S., U.K. and Z.M.; writing—review and editing, F.S., Y.A.S., K.K., M.S., U.K. and Z.M.; visualization, F.S. and Z.M.; supervision, Z.M.; project administration, Z.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Fan, X.; Zhao, W. Improving robustness of license plates automatic recognition in natural scenes. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 18845–18854. [[CrossRef](#)]
2. Xie, L.; Ahmad, T.; Jin, L.; Liu, Y.; Zhang, S. A new CNN-based method for multi-directional car license plate detection. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 507–517. [[CrossRef](#)]
3. Soon, F.C.; Khaw, H.Y.; Chuah, J.H.; Kanesan, J. PCANetbased convolutional neural network architecture for a vehicle model recognition system. *IEEE Trans. Intell. Transp. Syst.* **2019**, *20*, 749–759. [[CrossRef](#)]
4. Lu, L.; Huang, H. A hierarchical scheme for vehicle make and model recognition from frontal images of vehicles. *IEEE Trans. Intell. Transp. Syst.* **2019**, *20*, 1774–1786. [[CrossRef](#)]
5. Farid, A.; Hussain, F.; Khan, K.; Shahzad, M.; Khan, U.; Mahmood, Z. A Fast and Accurate Real-time Vehicle Detection Method Using Deep Learning for Unconstrained Environments. *Appl. Sci.* **2023**, *13*, 3059. [[CrossRef](#)]
6. Selmi, Z.; Halima, M.B.; Alimi, A.M. Deep learning system for automatic license plate detection and recognition. In Proceedings of the 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), Kyoto, Japan, 9–15 November 2017; pp. 1132–1138.
7. Shi, J.W.; Zhang, Y. License plate recognition system based on 476 improved YOLOv3 and BGRU. *Comput. Eng. Des.* **2020**, *41*, 2345–2351.
8. Silva, S.M.; Jung, C.R. Real-time license plate detection and recognition using deep convolutional neural networks. *J. Vis. Commun. Image Represent.* **2020**, *71*, 102773. [[CrossRef](#)]
9. Lee, Y.; Yun, J.; Hong, Y.; Lee, J.; Jeon, M. Accurate license plate recognition and super-resolution using a generative adversarial networks on traffic surveillance video. In Proceedings of the International Conference on Consumer Electronics-Asia (ICCE-Asia), Jeju, Republic of Korea, 24–26 June 2018; pp. 1–4.
10. Mahmood, Z.; Khan, K.; Khan, U.; Adil, S.H.; Ali, S.S.A.; Shahzad, M. Towards Automatic License Detection. *Sensors* **2022**, *22*, 1245. [[CrossRef](#)]
11. Jia, W.; Zhang, H.; He, X. Region-based license plate detection. *J. Netw. Comput. Appl.* **2007**, *30*, 1324–1333. [[CrossRef](#)]
12. Silva, S.M.; Jung, C.R. License Plate Detection and Recognition in Unconstrained Scenarios. In *Computer Vision—ECCV 2018 Lecture Notes in Computer Science*; Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y., Eds.; Springer: Cham, Switzerland; Volume 11216, pp. 593–609. [[CrossRef](#)]
13. Laroca, R.; Severo, E.; Zanlorensi, L.A.; Oliveira, L.S.; Goncalves, G.R.; Schwartz, W.R.; Menotti, D. A Robust Real-time Automatic License Plate Recognition based on the YOLO Detector. In Proceedings of the International Joint Conference on Neural Network (IJCNN), Rio de Janeiro, Brazil, 8–13 July 2018.
14. Hsu, G.S.; Ambikapathi, A.; Chung, S.L.; Su, C.P. Robust license plate detection in the wild. In Proceedings of the 2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), Lecce, Italy, 29 August–1 October 2017; pp. 1–6.
15. Selmi, Z.; Halima, M.B.; Pal, U.; Alimi, M.A. DELP-DAR system for license plate detection and recognition. *Pattern Recognit. Lett.* **2020**, *129*, 213–223. [[CrossRef](#)]
16. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. In Proceedings of the International Conference on Learning Representations (ICLR), San Diego, CA, USA, 7–9 May 2015.
17. Bulan, O.; Kozitsky, V.; Ramesh, P.; Shreve, M. Segmentation and annotation-free license plate recognition with deep localization and failure identification. *IEEE Trans. Intell. Transp. Syst.* **2017**, *18*, 2351–2363. [[CrossRef](#)]
18. Zhu, J.Y.; Park, T.; Isola, P.; Efros, A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 2223–2232.
19. Li, H.; Wang, P.; Shen, C.; Zhang, G. Show, attend and read: A simple and strong baseline for irregular text recognition. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; pp. 8610–8617.
20. Wang, X.; Man, Z.; You, M.; Shen, C. Adversarial generation of training examples: Applications to moving vehicle license plate recognition. *arXiv* **2017**, arXiv:1707.03124.
21. Xu, H.; Zhou, X.-D.; Li, Z.; Liu, L.; Li, C.; Shi, Y. EILPR: Toward End-to-End Irregular License Plate Recognition Based on Automatic Perspective Alignment. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 1–10. [[CrossRef](#)]
22. Wang, Y.; Bian, Z.; Zhou, Y.; Chau, L.-P. Rethinking and designing a high-performing automatic license plate recognition approach. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 8868–8880. [[CrossRef](#)]
23. Qiuying, H.; Cai, Z.; Lan, T. A single neural network for mixed style license plate detection and recognition. *IEEE Access* **2021**, *9*, 21777–21785.
24. Yogheedha, K.; Nasir, A.; Jaafar, H.; Mamduh, S. Automatic vehicle license plate recognition system based on image processing and template matching approach. In Proceedings of the International Conference on Computational Approach in Smart Systems Design and Applications (ICASSDA), Serawak, Malaysia, 15–17 August 2018; pp. 1–8.
25. Chris, H.; Ahn, S.Y.; Lee, S.W. Multinational license plate recognition using generalized character sequence detection. *IEEE Access* **2020**, *8*, 35185–35199.

26. Khan, A.M.; Awan, S.M.; Arif, M.; Mahmood, Z.; Khan, G.Z. A Robust Segmentation Free License Plate Recognition Method. In Proceedings of the 1st International Conference on Electrical, Communication and Computer Engineering (ICECCE), Swat, Pakistan, 24–25 July 2019; pp. 1–6.
27. Tourani, A.; Shahbahrani, A.; Soroori, S.; Khazaee, S.; Suen, C.Y. A robust deep learning approach for automatic iranian vehicle license plate detection and recognition for surveillance systems. *IEEE Access* **2020**, *8*, 201317–201330. [[CrossRef](#)]
28. Shashirangana, J.; Padmasiri, H.; Meedeniya, D.; Perera, C. Automated license plate recognition: A survey on methods and techniques. *IEEE Access* **2020**, *9*, 11203–11225. [[CrossRef](#)]
29. Hassaballah, M.; Kenk, M.; Muhammad, K.; Minaee, S. Vehicle detection and tracking in adverse weather using a deep learning framework. *IEEE Trans. Intell. Transp. Syst.* **2020**, *22*, 4230–4242. [[CrossRef](#)]
30. Ren, S.; He, K.; Ross, G. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1135–1149. [[CrossRef](#)]
31. Hsieh, J.-W.; Yu, S.-H.; Chen, Y.-S. Morphology-based license plate detection from complex scenes. In Proceedings of the 2002 International Conference on Pattern Recognition, Quebec City, QC, Canada, 11–15 August 2002; Volume 3.
32. Llorca, D.F.; Salinas, C.; Jimenez, M.; Parra, I.; Morcillo, A.G.; Izquierdo, R.; Lorenzo, J.; Sotelo, M.A. Two-camera based accurate vehicle speed measurement using average speed at a fixed point. In Proceedings of the 2016 IEEE 19th International Conference on Intelligent Transportation System (ITSC), Rio de Janeiro, Brazil, 1–4 November 2016; pp. 2533–2538.
33. Masood, S.Z.; Shu, G.; Dehghan, A.; Ortiz, E.G. License plate detection and recognition using deeply learned convolutional neural networks. *arXiv* **2017**, arXiv:1703.07330.
34. Yuan, Y.; Zou, W.; Zhao, Y.; Wang, X.; Hu, X.; Komodakis, N. A robust and efficient approach to license plate detection. *IEEE Trans. Image Process.* **2017**, *26*, 1102–1114. [[CrossRef](#)] [[PubMed](#)]
35. Hsu, G.S.; Chen, J.C.; Chung, Y.Z. Application-Oriented License Plate Recognition. *IEEE Trans. Veh. Technol.* **2013**, *62*, 552–561. [[CrossRef](#)]
36. Li, H.; Shen, C. Reading car license plates using deep convolutional neural networks and LSTMs. *arXiv* **2016**, arXiv:1601.05610.
37. Li, H.; Wang, P.; Shen, C. Toward end-to-end car license plate detection and recognition with deep neural networks. *IEEE Trans. Intell. Transp. Syst.* **2019**, *20*, 1126–1136. [[CrossRef](#)]
38. Wu, C.; Xu, S.; Song, G.; Zhang, S. How many labeled license plates are needed. In Proceedings of the Chinese Conference on Pattern Recognition and Computer Vision (PRCV), Guangzhou, China, 23–26 November 2018; Springer: Cham, Switzerland, 2018; pp. 334–346.
39. Zhang, L.; Wang, P.; Li, H.; Li, Z.; Shen, C.; Zhang, Y. A robust attentional framework for license plate recognition in the wild. *IEEE Trans. Intell. Transp. Syst.* **2021**, *22*, 6967–6976. [[CrossRef](#)]
40. Xu, Z.; Yang, W.; Meng, A.; Lu, N.; Huang, H.; Ying, C.; Huang, L. Towards end-to-end license plate detection and recognition: A large dataset and baseline. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 255–271.
41. Zherzdev, S.; Gruzdev, A. LPRNet: License plate recognition via deep neural networks. *arXiv* **2018**, arXiv:1806.10447.
42. Luo, C.; Jin, L.; Sun, Z. MORAN: A multi-object rectified attention network for scene text recognition. *Pattern Recognit.* **2019**, *90*, 109–118. [[CrossRef](#)]
43. Wang, T.; Zhu, Y.; Jin, L.; Luo, C.; Chen, X.; Wu, Y.; Wang, Q.; Cai, M. Decoupled attention network for text recognition. *arXiv* **2020**, arXiv:1912.10205. [[CrossRef](#)]
44. Yu, S.; Li, B.; Zhang, Q.; Liu, C.; Meng, M.-Q.H. A novel license plate location method based on wavelet transform and EMD analysis. *Pattern Recognit.* **2015**, *48*, 114–125. [[CrossRef](#)]
45. Zhao, J.; Hao, S.; Dai, C.; Zhang, H.; Zhao, L. Improved Vision-Based Vehicle Detection and Classification by Optimized YOLOv4. *IEEE Access* **2022**, *10*, 8590–8603. [[CrossRef](#)]
46. Tran, D.N.N.; Pham, L.H.; Nguyen, H.H.; Jeon, J.W. City-Scale Multi-Camera Vehicle Tracking of Vehicles based on YOLOv7. In Proceedings of the International Conference on Consumer Electronics-Asia (ICCE-Asia), Yeosu, Republic of Korea, 26–28 October 2022; pp. 1–4.
47. Zhao, H.; Zhang, H.; Zhao, Y. Yolov7-sea: Object detection of maritime uav images based on improved yolov7. In Proceedings of the Winter Conference on Applications of Computer Vision, Wailoloa, HI, USA, 3–7 January 2023; pp. 233–238.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.