Journal of Theoretical and Applied Information Technology
15th February 2020. Vol.98. No 03

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

REAL-TIME LICENSE PLATE DETECTION BASED ON VEHICLE REGION AND TEXT DETECTION

HOANH NGUYEN
Faculty of Electrical Engineering Technology, Industrial University of Ho Chi Minh City, Ho Chi Minh City, Vietnam
E-mail: nguyenhoanh@iuh.edu.vn

ABSTRACT

This paper presents a new approach for real-time license plate detection based on vehicle and text regions. Firstly, vehicle regions are extracted by single shot multibox detector (SSD) framework. Secondly, multi-channel maximally stable extremal regions (MSER) algorithm is used to generate character candidates in the vehicle regions. Using properties of vehicle regions, this paper filters out false character candidates and then constructs license plate candidates with remaining character candidates. Then, false license plate candidates are eliminated by exploiting the correlation between dimension of vehicle and license plate. Finally, remaining license plate candidates are passed to a word/no-word classifier to keep the final license plate. To run in real time on embedded systems, this paper chooses the MobileNets architecture for deep CNN configurations. Experimental results on the public test dataset and new collected dataset show that the proposed approach can apply to different types of license plates with better performance than current state-of-the-art methods.

Keywords: *License Plate Detection, Convolutional Neural Network, Intelligent Transportation Systems, Object Detection, Deep Learning*

1. INTRODUCTION

Vehicle license plate recognition is widely used in intelligent transport systems, traffic control, vehicle parking and so on. An automatic license plate recognition normally includes two stages: license plate detection and license plate recognition. License plate detection takes images captured from camera to extract exactly license plate region, while license plate recognition segments and recognizes each character on license plate. License plate detection has dramatic effect on the accuracy of the whole system. Thus, many approaches have been proposed to detect license plate. Traditional methods [5, 6, 18, 19, 20, 21] are usually based on morphological features of license plate such as colour, dimension, character and so on. These methods perform well under certain limited conditions such as the same type/region/country, fixed illumination, fixed viewpoint and simple background. If these conditions change, these methods show poor performance. Recently, with fast development of deep learning, a certain number of methods for detecting license plate based on deep learning [1, 2, 3, 4, 10] have been proposed. These methods show better accuracy than traditional methods. Deep CNN-based methods firstly create license plate candidates. Then, a deep CNN-based plate/non-plate classifier is used to reject non-plate candidates. Although these methods perform well in complex conditions, they cannot run in real time on low-spec machines. Furthermore, a deep CNN-based plate/non-plate classifier requires numerous license plate images for learning process. Because license plate has a wide variety of types, dimensions, colours, characters in different countries or regions, it is impossible to collect a large number of license plate images including many types. Therefore, these methods focus on specific license plate such as American license plate and Taiwan license plate. In addition, with privacy concerns, there are not any standard datasets available for training, so license plate detection is rarely considered as a part of object detection problem in the existing literature. To tackle these issues, this paper proposes a new approach for detecting license plate in real time. To be widely used with different license plate types, this paper considers not to use a plate/non-plate classifier, which requires a large number of license plate images. With a fixed position on vehicle, a change of size of vehicle in an image will lead to change the size of license plate with the same ratio. Thus, this paper firstly extracts vehicle regions in image by using a deep CNN-based object detector. Then,

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

multi-channel MSER is used to detect character candidates for each channel. With each channel, this paper filters out false candidates based on the dimension of vehicle region. Next license plate candidates are formed by connecting remaining character candidates. False license plate candidates on each channel are eliminated by exploiting the correlation between dimension of vehicle and license plate, and then remaining license plate candidates on different channels are merged by non-maximum suppression algorithm. Because a license plate always includes several characters, this paper constructs a word/no-word classifier based on MobileNets architecture to eliminate no-word candidates and keep final license plate. Finally, this paper uses vertical edges projection to refine final license plate and get better intersection over union (IoU). The main contributions of this paper are summarized as follows:

- This paper proposes a new approach to detect different types of vehicle license plates in diverse outdoor scenes. Moreover, the proposed approach performs well in a situation where license plates are small in complex background.
- This paper uses two state-of-the-art deep CNN frameworks: SSD framework for vehicle detection and MobileNets architecture for building classifier. With these frameworks, the proposed method can run in real time on low-spec machines.
- This paper collects a new dataset with license plate images in outdoor scenes to test the proposed method. Experimental results show that the proposed method performs well on new type of license plate without retraining.
- Comparing with other results from the state-of-the-art methods on the same test dataset, the proposed method achieves better results on both detection ratio and precision/recall.

This paper is organized as follows: an overview of previous methods is presented in Section 2. Section 3 describes detail the proposed method. Section 4 demonstrates experimental results. Finally, the conclusion is made in Section 5.

2. RELATED WORK

In this section, this study introduces previous approaches, which are related to license plate detection, including traditional methods and recently proposed methods based on deep CNN. Besides, methods for vehicle detection and text detection will also be discussed.

2.1 Vehicle Detection

Vehicle detection system firstly locates vehicle candidate regions. Then, a classifier is constructed to eliminate false vehicle candidate regions. Traditional methods usually apply a sliding-window with different scales and ratios to create candidate regions. These approaches will create numerous candidate windows, so the overall performance of system will be limited. To overcome this problem, Yuan et al. [33] proposed a flexible bounding-box generating algorithm to locate the vehicle proposal regions. A graph-based algorithm is then used to compute a vehicle proposal score for each bounding box. Recently, deep CNN-based methods have become the leading method for high quality general object detection. Faster region-based convolutional neural network (Faster R-CNN) [26] defined a region proposal network (RPN) for generating region proposals and a network using these proposals to detect objects. RPN shares full-image convolutional features with the detection network, thus enabling nearly cost-free region proposals. This method has achieved state-of-the-art detection performance and become a commonly employed paradigm for general object detection. SSD framework [27] predicted category scores and box offsets for a fixed set of default bounding boxes using small convolutional filters applied to different scales from feature maps of different scales, and explicitly separate predictions by aspect ratio. This framework showed much faster and comparably performance with other methods. Most of these deep learning models target general object detection including vehicle. To better handle the detection problem of vehicles in complex conditions, Chu et al. [32] proposed a vehicle detection scheme based on multi-task deep CNN which learning is trained on four tasks: category classification, bounding box regression, overlap prediction and subcategory classification. A region-of-interest voting scheme and multi-level localization are then used to further improve detection accuracy and reliability. Experimental results on standard test dataset showed better performance than other methods.

2.2 License Plate Detection

2.2.1 Traditional Methods

Traditional methods are usually based on morphological features of license plate such as colour, edge, texture, character and so on to detect license plate [22], [34]. Recently, a certain number of methods combining these morphological features for detecting license plate have been proposed. In [5], a new Riesz fractional model was used to improve low quality license plate images affected by

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

multiple factors. Since the proposed model involves differential operation by performing convolution operation over each input image with the Riesz fractional derivative window, it enhances edges irrespective of distortions created by multiple factors. After improving the quality of the input license plate images, the authors modified MSER algorithm used for character candidate detection by adding stroke width information and then used a classifier to eliminate non-character. In [6], Yuan et al. proposed a novel image downscaling method for license plate detection which can substantially reduce the size of the image without sacrificing detection performance. For generating license plate candidates, the authors proposed a novel line density filter for extracting license plate candidates. Then, these candidates will be refined by connected-component labelling algorithm. Finally, the real license plate is obtained from the detected candidate regions by cascaded license plate classifier based on linear SVMs and colour saliency features. Gou et al. [18] used morphological operations, various filters, different contours and validations for detecting coarse license plate. Then character-specific ERs are selected as character regions through a Real AdaBoost classifier with decision trees. Accurate character segmentation and license plate location are achieved based on the geometrical attributes of characters in standard license plates. Finally, characters are extracted and recognized using an off-line trained classifier based on HDRBM. In [19], Li et al. extracted MSER as character candidates. The authors then introduced a CRF model to describe the contextual relationship among the candidates. Finally, license plates are located through CRF inference. In [20], Ashtari et al. proposed an Iranian vehicle license plate recognition system based on a modified template-matching technique by the analysis of target colour pixels to detect the location of a license plate, along with a hybrid classifier that recognizes license plate characters. Zhou et al. [23] proposed a scheme to automatically locate license plates with principal visual word, discovery and local feature matching algorithm. The authors bring in the idea of using the bag-of-words model popularly applied in partial-duplicate image search. Traditional methods can be run in real time on low-spec machines, and these approaches achieve good performance with limited conditions such as license plates with the same country, simple background, simple environment and so on. However, if these conditions change, the performance of these methods will be significantly reduced.

2.2.2 Deep CNN-based methods

Recently, a certain number of methods for detecting license plate based on deep CNN have been proposed. In [1], Li et al. used a 4-layer with 37-class CNN classifier in a sliding-window fashion across the entire image at first stage to detect the presence of text and then create a text saliency map. At second stage, a deep CNN-based 4-layer plate/non-plate classifier is adopted to reject the false negative. In [2], Kim et al. proposed a method for vehicle detection instead of detecting license plate at first. The Faster R-CNN framework is adopted for detecting vehicle region. The hierarchical sampling method is used for generating license plate candidates. Finally, non-plate candidates are filtered out by a deep CNN-based plate/non-plate classifier. Rafique et al. [3] used the Faster R-CNN framework with VGG16/ZF architecture to directly detect license plates. There is not any standard license plate dataset available for training Faster R-CNN, so the authors extended Pascal VOC2007 data by combining a publicly available license plate dataset. Bulan et al. [4] proposed a two-stage approach, where a set of candidate regions are firstly extracted by a weak sparse network of winnows classifier trained with successive mean quantization transform features and then scrutinized by a strong readable/unreadable CNN classifier in the second stage. Images which fail a primary confidence test for plate localization are further classified to identify reasons for failure such as license plate not present, license plate too bright, license plate too dark and no vehicle found. Xie et al. [10] introduced a new MD-YOLO model for multi-directional car license plate detection. The proposed model could elegantly solve the problem of multi-directional car license plate detection and could also be deployed easily in real time circumstances because of its reduced computational complexity compared with previous CNN-based methods. The authors used a prepositive CNN, which plays an important role in removing redundant information, because the image area proportion of the car license plate is usually very small, and small image portions such as the car license plate may deem to introduce some redundant information.

In this paper, vehicle regions are detected at first step. This is similar to the approach used in [2]. However, different from [2], this paper uses information from extracted vehicle regions to construct and refine license plate candidates effectively.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

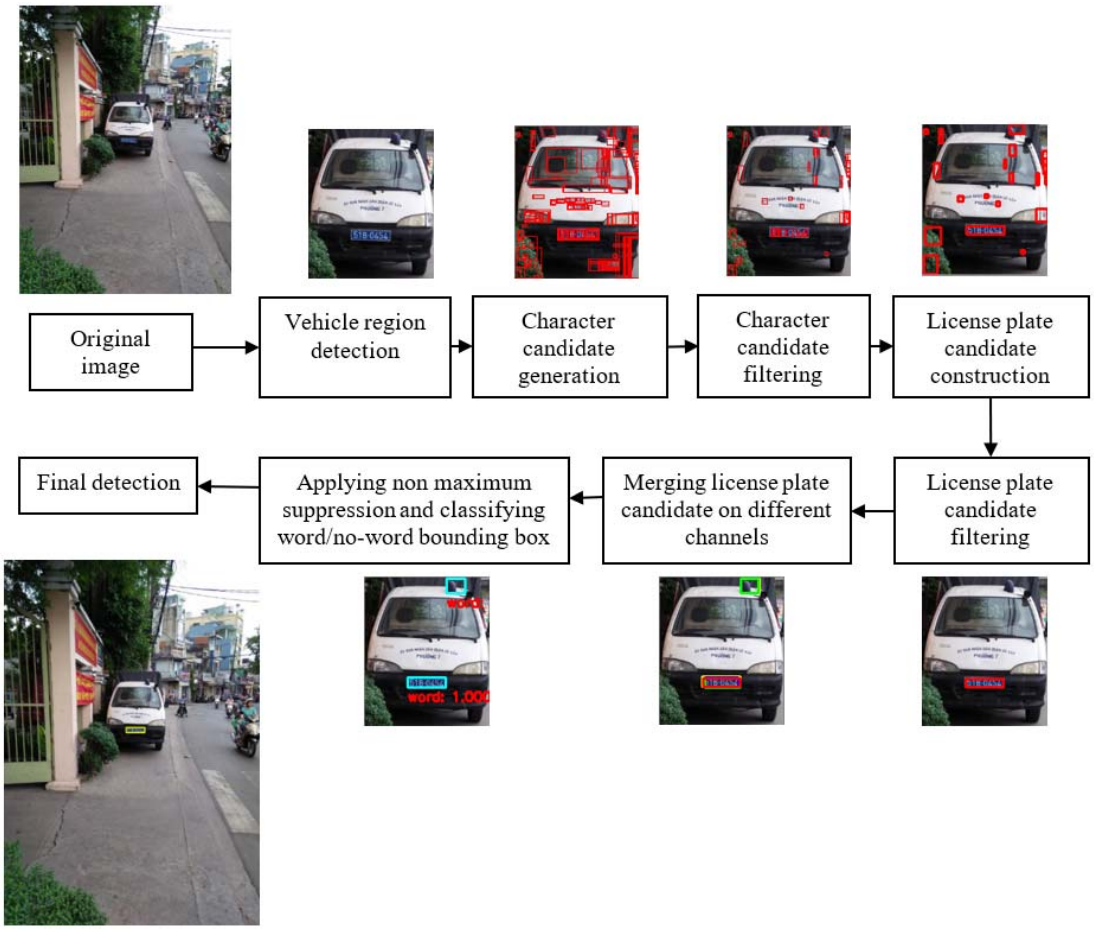


Figure 1: The Overall Framework of The Proposed Method.

2.3 Scene text detection

With the important role in real world, a large number of scene text detection methods have been proposed [17]. Basically, they can be divided in two groups: sliding window-based methods and connected component-based methods. Sliding window methods slide a large number of windows with different scales and ratios through all possible positions of the image and extract features at first step. Then, a strong classifier is used to classify text and non-text regions. Tang et al. [7] proposed a new approach which firstly use a CNN-based text-aware candidate text region extraction model for creating coarse text candidate regions. To refine coarse text candidates, the authors then used a CNN-based refinement model to precisely segment the coarse regions into text to get the refined text regions. The refined text regions are finally classified by a CNN-based text region classification model to get the final text regions. In [9], Jaderberg et al. used a combination of Edge Box proposals and a trained aggregate channel features detector to generate word bounding box candidates. Then, a random forest classifier is used to filter the number of proposals to a manageable size. Finally, a CNN classifier is trained for regression. The advantage of sliding-window methods is that they create candidates with high recall, but they also create numerous candidate regions, which need significant effort to classify.

Connected component methods are based on the properties of pixel such as intensity, colour and stroke width to extract features from the connected components. Then, a classifier is used to filter out non-text regions. He et al. [8] developed a novel Text-CNN classifier and an improved CE-MSERs detector, which together lead to a significant performance boost. Text-CNN model is specially designed to compute discriminative text features from general image components by leveraging more informative supervised information that facilitates text feature computing. In [11], text components are generated by applying the MSER detector on the input image. Then, each MSER component is assigned a confident value by using a trained CNN

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

a *b*

Figure 2: Examples of Vehicle Detection on Two Test Datasets, (a) Caltech Cars Dataset, (b) Collected Dataset. Vehicle Regions with Blue Rectangles are Extracted for Next Step.

classifier. Finally, the text components with high confident score are employed for constructing the final text-lines. Tian et al. [12] proposed a multi-level MSER technique to extract text candidates from scene images by using a range of the controlling parameter deltas. A segmentation score, which is based on stroke width, boundary curvature, character confidence and colour constancy, is designed to measure the segmentation performance and select best-quality text candidates based on the segmentation score. Qin and Manduchi [13] used multi-channel MSER segmentation at the first stage. At second stage, the number of MSERs is reduced by selecting one representative MSER per cluster and specifically the one with the highest fullness in the cluster. Then representatives of region clusters are passed to a deep network, and a final line grouping stage forms word-level segment. Inspired by the original Canny edge detector, Cho et al. [14] presented a novel algorithm using double threshold and hysteresis tracking to detect texts of low confidence. Liu et al. [15] presented a variant MSER algorithm to extract character candidates from channels of G, H, S, O1, and O2. Non-text components are identified and removed with a two-layers filtering scheme. Then, text candidates are constructed by grouping the component pairs and then linked them by the single link clustering algorithm. In [16], Yin et al. proposed a robust and accurate MSER-based scene text detection method. Firstly, the authors proposed a novel pruning algorithm to remove the repeating components extracted by MSERs algorithm. Secondly, text candidates are constructed by clustering character candidates based on the single-link algorithm using the learned parameters. Finally, a character classifier is used to estimate the posterior probability of text candidate corresponding to non-text and eliminate text candidates with high non-text probability. The advantage of connected-component methods is that they can greatly reduce the number of candidate regions, but they also miss some true character regions. Especially, if characters are blurred or obscured, these methods show reduced performance. With the purpose of running in real time, this paper uses connected component-based method for generating character candidates.

3. PROPOSED APPROACH

This section presents the proposed approach to license plate detection. As illustrated in Figure 1, this paper firstly extracts all vehicle regions in image. With each vehicle region, MSER algorithm is used on different channels to find character candidates. Then, license plate candidates are constructed by using character candidates. Finally, a deep CNN-based classifier is used to locate the final license plate.

3.1 Deep CNN-based Object Detector for Vehicle Detection

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

a b

Figure 3: MSER Results on Grayscale Channel of Image, (a) Original MSER Results, (b) MSER Results after Filtering Out False Character Candidates.

a b

Figure 4: Grouping MSER Results on The Same Channel of The Image. Each Group with The Same Colour (Left) and Bounding Box of The Same Group (Right), (a) Small Characters and Small Distance Between Them, (b) Large Characters and Large Distance Between Them.

For the purpose of detecting different types of license plates in outdoor scenes, this study firstly detects vehicle [2] instead of directly detecting license plates. Deep CNN-based object detectors exhibit poor performance with small objects [25], while license plates are usually small on image. Thus, some methods directly detecting license plate with a deep CNN-based object detector [10, 3] showed a limited performance. Furthermore, because of the difference in size, colour and character of license plate and privacy concerns, it is impossible to collect a large number of license plate images covering all license plates in the world, while deep CNN-based methods require a plenty of data.

Currently, deep CNN-based methods are focusing on license plate of specific country [1, 2, 3, 4, 5, 6, 24], where a small public dataset is available. Otherwise, with a lot of available public datasets, vehicle is easy to detect with training-based method because of their similarity. Moreover, after detecting vehicle regions, the number of character candidates significantly reduces. This will help to speed up the process and eliminate false positive candidates effectively. In addition, the fixed position of license plate on vehicle also help to early reject detected character candidates, which are not belonged to license plate.

Recently, many different deep CNN-based object detectors have been proposed to improve results of object detection and reduce processing time. Recent state-of-the-art frameworks include Faster R-CNN [26], SSD [27] and Region-based Fully Convolutional Networks (R-FCN) [30]. These general object detectors achieve significantly improved performance compared to traditional methods, so they can handle vehicle detection very well. SSD combines region proposals and region classifications in a ‘single shot’. The core of SSD is predicting category scores and box offsets for a fixed set of default bounding boxes using small

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

convolutional filters applied to different scales from feature maps of different scales, and explicitly separate predictions by aspect ratio. Because SSD does everything in one shot, it is the fastest model. With the same network architecture and input resolution, Faster R-CNN achieves better accuracy, but SSD is much faster with competitive accuracy [25, 27].

There are some deep CNN architectures that showed state-of-the-art performance on many competitions such as VGG-16 won the ILSVRC challenge, Resnet-101 won COCO 2015 challenge, Inception v2 won ILSVRC 2014 classification and detection challenge and so on. Google recently released an efficient model called MobileNet [28] for mobile and embedded vision applications. MobileNets splits the convolution into a 3x3 depthwise convolution and a 1x1 pointwise convolution, effectively reducing both computational cost and number of parameters. It introduces two parameters that we can tune to fit the resource/accuracy trade-off: width multiplier and resolution multiplier. The width multiplier allows us to thin the network, while the resolution multiplier changes the input dimensions of the image, reducing the internal representation at every layer. For the purpose of real time processing, this paper chooses MobileNets as a based network for SSD framework.

To improve performance, this study uses SSD framework with MobileNets architecture pre-trained model on COCO dataset [29] and then further fine-tunes on the VehicleDataset [43] to obtain the final model. VehicleDataset contains 9,850 vehicle images with six categories: bus, microbus, minivan, sedan, SUV, and truck. The number of images in each type is 558, 883, 476, 5,922, 1,392 and 822 respectively. These images are different from illumination condition, scale, surface colour and viewpoint. Because almost images in this dataset contain front view or real view of the vehicle, they will help to improve the overall performance of license plate detection since only real view or front view contains license plate. For fine-tuning, due to the constraints of GPU memory on low-spec machines, this study uses learning rate at 0.0001 with batch size of 8. Random horizontal flip and random crop are used for data augmentation. Other parameters are the same as setup in [25]. Training process stops after 100,000 mini-batch iterations. This paper then uses the final model for detecting vehicle on test datasets. Figure 2 shows some examples of vehicle detection on two test datasets respectively. After detecting vehicle on image, this study uses a threshold score at 0.5 for true vehicle regions and then rejects all true vehicle regions with area below 1/120 the area of the image. All true vehicle regions are extracted for next step.

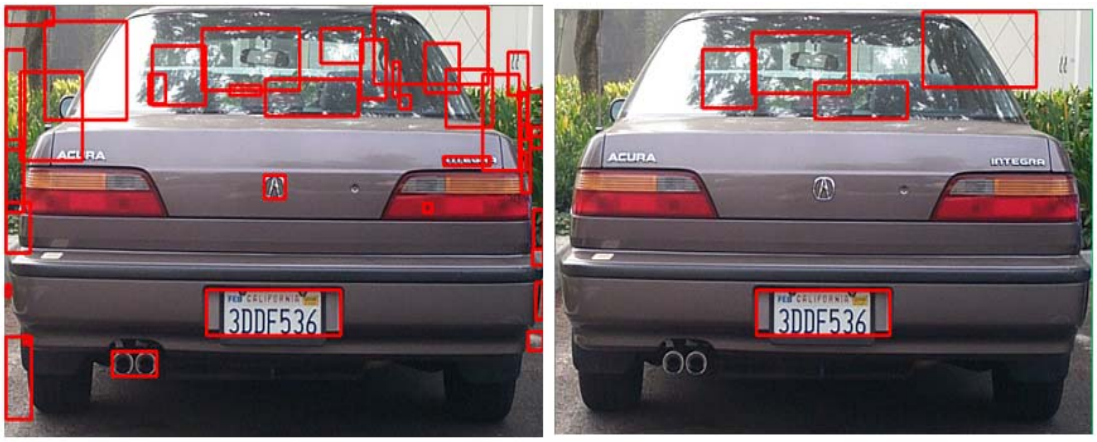
3.2 Character Candidate Generation

Each vehicle region detected from previous step is cropped for generating character candidates. Although license plates in different countries have different types of sizes, colours and fonts, they have the same feature that they are made up of several consecutive characters. These characters are arranged horizontally with the same size. To detect license plates in different countries without retraining, this paper considers license plate as a word consisting of multiple characters. A deep CNN-based word/no-word classifier [7, 8, 11] shows high performance in many cases. Moreover, it is feasible to collect a dataset consisting of different types of words.

To generate character candidates, many methods have been proposed. These methods can be divided into three groups: based on sliding windows [35, 36], based on edges information [9, 11, 12, 37, 38] and based on deep CNN [1, 4, 7]. Sliding window methods create candidates with high recall, but they take long time to finish process. Edges-based methods detect every character in regions with more precise. Thus, edges-based methods show more efficient performance compared to sliding window methods. The weakness of edges-based methods is that they cannot detect characters which are reflected, blurred or obscured. With advantages of deep CNN classifier, deep CNN-based methods usually create a good result in complicated situations, but they still take long time for generating candidates.

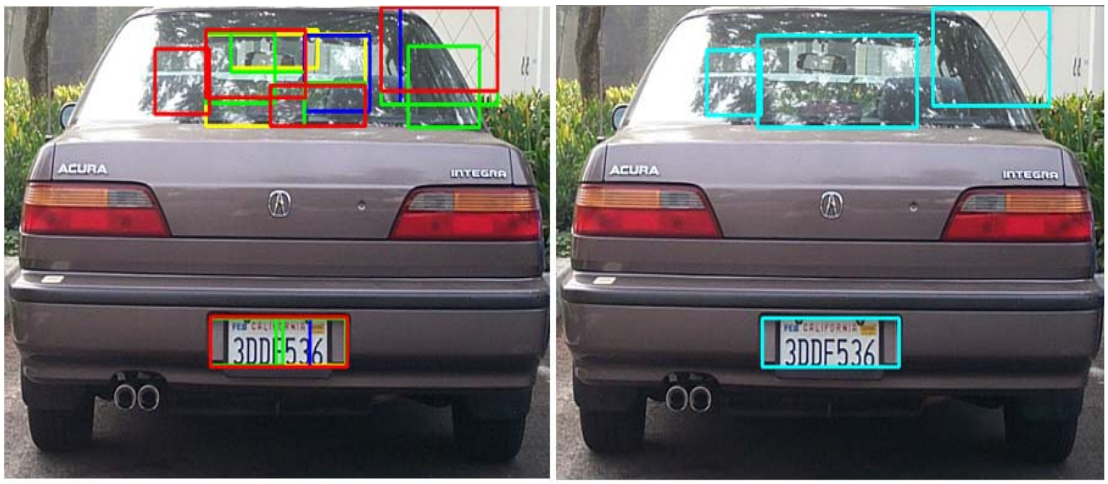
In this study, multi-channel MSER algorithm is used for generating character candidates. MSER algorithm is very capable of detecting characters by considering each of them as a stable extremal region. A certain number of methods for text detection based on MSER have been proposed and showed an impressive performance on text detection [8, 11, 13, 39]. This paper chooses MSER algorithm based on the following facts. Firstly, MSER algorithm works perfectly if characters have high contrast to the background, while characters on license plate always have opposite colour to background. Secondly, MSER detector is computationally fast and can be computed in linear time of the number of pixels in image [11]. Finally, the performance of MSER detector is reduced if characters are reflected, blurred or obscured. This may happen when characters on license plate are blurred and cannot consider as a stable extremal region. This paper

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195



a b

Figure 5: Filtering Coarse License Plate Candidates on Grayscale Channel, (a) Before Filtering, (b) After Filtering.



a b

Figure 6: License Plate Candidates on Different Channels, (a) Before NMS. Yellow Rectangles: Candidates on Grayscale Channel; Red Rectangles: Candidates on R Channel; Blue Rectangles: Candidates on B Channel; Green Rectangle: Candidates on R Channel, (b) After NMS.

overcomes this situation by a refinement step showing at following section. More specific, if some characters cannot be detected, this paper still can cluster remaining characters to a license plate based on their geometry shape. To improve performance of MSER, this paper computes MSER on four image channels: grayscale, R, G and B. These MSER results are used as character candidate components.

3.3 License Plate Construction Based on Character Candidates and Vehicle

Since characters on license plate are quite small compared to the size of the vehicle, and the height of characters is always greater than the width, this paper first removes all MSER results that meet the following conditions:

$$h_c > \frac{h_v}{10} \cup w_c > \frac{w_v}{10} \cup \frac{h_c}{w_c} < 1 \cup \frac{h_v}{w_c} > 8 \quad (1)$$

Here h_v, w_v, h_c and w_c represent the height, width of the vehicle region and the MSER component respectively. The conditions (1) help to eliminate all false character candidates in case of too long, too high and too big. Otherwise, because of the effect of shadows or brightness, MSER results may be missing on some parts of a character leading to small detected component. Thus, this paper keeps all small MSER components. Figure 3 shows MSER results with original and filtered MSER results on grayscale

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Figure 7: Examples of Word/No-Word Classification, (a) Classification Results, (b) Final Detection Results.

channel of original image. As shown in this figure, a large number of false character candidates, which are too big/high/long, have been eliminated.

Next, this paper constructs license plate candidates based on remaining character candidates. For each channel, this paper checks for bounding boxes of every remaining character candidate and groups them to the same group if they are 'nearby' each other. Because the spatial distance between characters on the same license plate is usually smaller than outside characters, character candidates are considered as 'nearby' if they have spatial distance as follows:

$$VD < \frac{w_c}{22} \cap HD < \frac{h_c}{10} \quad (2)$$

Here, *VD* and *HD* represent the vertical distance and horizontal distance of two nearby character candidates respectively. Since license plate is always fixed on vehicle, a change of size of vehicle on image will lead to change at the same ratio for characters on license plate. Thus, condition (2) is always true for any size of vehicle in an outdoor scene image. Some methods [12, 14, 15, 16] considered colour, stroke width, spatial distance and scale ratio to check for 'nearby' characters. These features may be affected by MSER results and other conditions such as illumination and background. Using the dimension of the vehicle, this paper can eliminate these effects. Figure 4 illustrates the efficiency of spatial distance condition. As shown in this figure, using the dimension of the vehicle, this

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

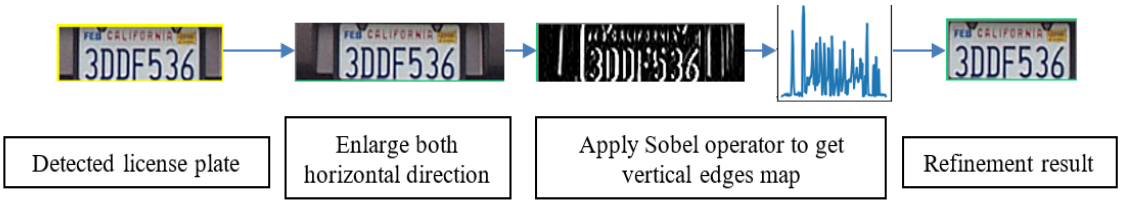


Figure 8: License Plate Refinement.

paper can exactly group nearby characters to the same group if the distance between them becomes larger as the size of the vehicle in image becomes bigger.

After grouping nearby characters, this paper finds the convex hull and then creates a bounding box for each group in each channel. This paper then rejects bounding boxes which are fully inside others. Remaining bounding boxes are considered as coarse license plate candidates. For each coarse license plate candidate, this paper filters out false license plate candidates by comparing the dimension of coarse license plate candidates with the dimension of vehicle as follows:

$$\frac{w_{lp}}{h_{lp}} < 0.8 \ \cup \ \frac{w_{lp}}{h_{lp}} > 6 \quad (3)$$

$$\frac{w_{lp}.h_{lp}}{w_v.h_v} < 0.01 \ \cup \ \frac{w_{lp}.h_{lp}}{w_v.h_v} > 0.1 \quad (4)$$

Here, w_{lp} and h_{lp} represent the width and height of coarse license plate candidates; w_v and h_v represent the width and height of vehicle region. Condition (3) is established based on the fact that license plates have rectangular or square shape. Thus, the width of license plate is always equal or greater than the height of license plate. This study chooses threshold at 0.8 to ensure that all license plate candidates are kept in case of missing some characters after generating character candidates. Condition (4) will reject all coarse license plate candidates which are too small or too big compared to the area of vehicle. Figure 5 shows results of license plate candidate construction on grayscale channel before and after filtering out false candidates. This study finds that conditions (1), (2) and (4) are important in this approach. Because license plate is always fixed on vehicle, a change of dimension of vehicle in image will lead to change at the same ratio for license plate. Thus, these conditions are true in any case and any type of license plate. Moreover, these conditions help to eliminate other text regions in vehicle.

Finally, instead of merging MSER results from each channel [15, 42], this study keeps MSER results on each channel and constructs license plate candidates on each channel. Because license plate candidates on different channels of the image usually overlap each other in position of true license plate, this study uses non-maximum suppression (NMS) algorithm on overlapped candidates from different channels and then keeps the candidates that have the largest width. Figure 6 shows license plate candidates on different channels and final license plate candidates after NMS. All final license plate candidates are cropped for next step.

3.4 Word/No-Word Classification

As shown in Figure 6, license plate candidates constructed from character candidates may contain false license plates. Thus, this paper builds a classifier to eliminate these false results. Recently proposed methods for classifying license plate candidate often use a plate/non-plate classifier to filter out non-plate candidates among license plate candidates [1, 2, 4, 5, 18]. These approaches require many license plate images. Since there are not any standard license plate datasets available for training, some approaches generate synthetic license plate images and then use them to train classifiers [4].

Because license plate always contains some characters, this paper builds a classifier to classify word and no-word candidates instead of classifying plate and non-plate candidates. With fast development of deep learning recently, a deep CNN-based word/no-word classifier has achieved significantly better results than traditional methods [9]. These deep CNN-based classifiers can be divided into character, word and text classifier. In this paper, a deep CNN-based word/no-word classifier is built by using MobileNets as feature extractor. MobileNets has achieved state-of-the-art performance on both latency and accuracy, and this architecture can be run in real time for different tasks on low-spec machines. Word/no-word classification is a two-class problem in image classification, so a thinner or shallower CNN architecture than the original MobileNets is enough to get good performance. In [28], thinner models using small width multiplier achieved better results than shallower models using fewer layers. Thus, this paper reduces the width multiplier to 0.5 and the

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Figure 9: Some Images in New Collected Dataset.

resolution multiplier to 128 instead of making the network shallower to achieve both good performance and real-time running. To train the network, this paper firstly collects 5000 images of word (2000 train samples and 3000 test samples) from IIIT 5K-word dataset [40]. Because license plate has at least three characters, this study keeps only 2500 images of word which have vertical layout, at most 10 characters and at least 3 characters in this dataset. This paper also crops 2500 images of word from the synthetic scene text dataset [41]. For no-word images, this paper randomly crops 5000 images from background of synthetic scene text dataset and VehicleDataset [42]. The original MobileNets model trained on the ImageNet dataset is used as the pre-trained model. This paper trains the network with learning rate at 0.01, batch size at 32, and the training is stopped after 250,000 training steps. Figure 7 shows some example results of word/no-word classification. After getting confident score for each license plate candidate, because each vehicle region has at most one license plate, this paper rejects all license plate candidates with confident score below 0.5. Then, license plate candidate with the highest confident score among remaining license plate candidates is selected as a true license plate.

3.5 License Plate Refinement

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 1: Details of New Collected Dataset.

Case	Description	Number of images
Case 1	Daytime, complex background, normal sunshine, one license plate, variety of sizes	200 (200 license plates)
Case 2	Nighttime, complex background, reflective glare, one license plate, variety of sizes	200 (200 license plates)
Case 3	Nighttime and daytime, complex background, variety of illumination conditions, multiple vehicle and license plate	100 (150 license plates)

Table 2: Performance Comparison on Caltech Cars Dataset (%).

Year	Method	Precision	Recall
2012	Zhou et al. [23]	95.5	84.8
2016	Li et al. [1]	97.56	95.24
2017	Kim et al. [2]	98.39	96.83
-	The proposed method with IoU at 0.5	98.43	99.21
-	The proposed method with IoU at 0.7	96.88	98.41

Because license plate candidates on different channels of the image usually overlap each other in position of true license plate, NMS algorithm was used to keep candidates with the largest width as explained in previous section. As shown in Figure 6, this approach may create an over-sized license plate at horizontal direction. To tackle this problem, this paper uses a simple transformation to refine detected license plate. Because characters on license plate are printed on vertical direction, this study firstly extends the boundary of detected license plate on both left and right of horizontal direction. Then, extended Sobel operator is used on extended license plate to get vertical edges map. Finally, vertical projection is applied on this edges map to get left and right boundary of license plate. Figure 8 illustrates the proposed approach on Caltech Cars license plate dataset.

4. EXPERIMENTAL RESULTS

The proposed method is implemented on a low-spec machine with Core i5 6400 processor, NVIDIA GTX 1050Ti gpu and 8 Gb of RAM. TensorFlow is used for implementing deep CNN models, and OPENCV library is used for real time processing.

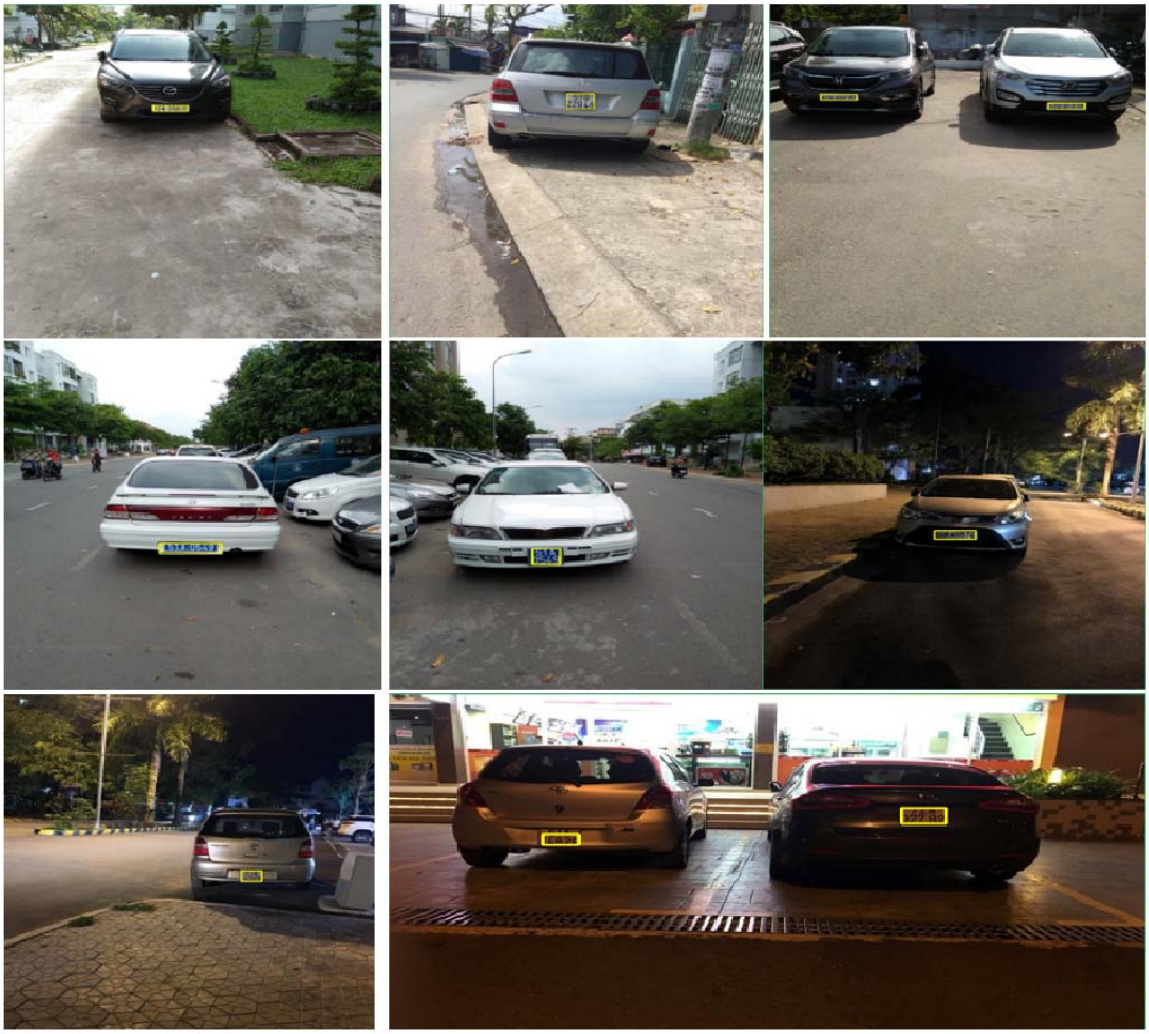
4.1 Datasets for Evaluation

Since there is no uniform dataset for evaluating license plate detection methods, this paper checks some publicly available datasets. These datasets are often used for evaluating in recent approaches such as Caltech Cars (Real) 1999 dataset [31], Application Oriented License Plate (AOLP) dataset [21] and PKU (Peking University) vehicle dataset [6]. Caltech Cars dataset includes 126 images of cars from the rear. The images are taken in the Caltech parking lots with resolution of 896 x 592 in different conditions, backgrounds and distances. There is one USA license plate in each image. AOLP dataset includes 2049 images of Taiwan car license plate and is categorized into three subsets: AC with 681 images, traffic LE with 757 images and RP with 611 images. PKU vehicle dataset includes 3828 vehicle images captured from various scenes under diverse conditions and is divided into five groups (G1-G5) corresponding to different configurations. Almost images in PKU and AOLP dataset were taken from near front view or real view of vehicle without fully containing vehicle. Because the proposed method is based on vehicle detection at first stage and focuses on detecting license plates in diverse outdoor scenes, this paper chooses the Caltech Cars dataset to evaluate the performance and compare to other state-of-the-art methods. Furthermore, this paper collects a new dataset with 500 images of vehicle in diverse outdoor scenes to show the effectiveness of the proposed method. These images have been collected in different conditions, illuminations and backgrounds. Table 1 shows detail of new collected dataset. Figure 9 shows some images in this new dataset.

4.2 Criteria for Evaluation

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

a



b

Figure 10: Detection Results of The Proposed Method, (a) Caltech Cars Dataset, (b) New Collected Dataset.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Figure 11: Examples of Failed Cases with MSER Results.

Table 3: Performance of The Proposed Method on New Collected Dataset (%).

Case	Detection ratio
Case 1	98.33
Case 2	92
Case 3	84

Table 4: Average Processing Time of Each Step.

Step	Processing time
Vehicle detection	0.15 s
License plate candidate construction	0.12 s
License plate classification	0.35 s
Total processing time	0.62 s

Since there is no uniform criterion for evaluating the performance of different license plate detection methods, most studies follow the criteria used in text detection, i.e. precision, recall and detection ratio. With the purpose of comparing results of the proposed method with other state-of-the-art methods, this paper also uses these criteria for evaluating the performance. Detection ratio [18] is defined as ratio between the number of correctly detected license plates and the number of all ground truth license plates. A license plate is correctly detected only if the overlap between the detected bounding box and ground truth bounding box is above a threshold. Precision is defined as ratio between the number of correctly detected license plates and the number of detected bounding boxes, while recall is defined as ratio between the number of correctly detected license plates and the number of ground truths. IoU is a threshold which measures the quality of detection. This paper also evaluates processing time of the proposed method on test datasets.

4.3 Detection Results

Because ground truths of the Caltech Cars dataset are not available, this paper manually labels all images in this dataset and new collected dataset. Figure 10 shows some results of the proposed method on the Caltech Cars dataset and new collected dataset. As shown in this figure, the proposed method can locate exactly position of license plate in complex outdoor scenes. Figure 11 presents some failed cases. The proposed method misses only one license plate on Caltech Cars dataset because MSER algorithm exhibits limited performance in this case. With new collected dataset, the proposed method fails in some cases where license plates are too small. Table 2 and Table 3 show results of the proposed method on Caltech Cars dataset and new collected dataset respectively. With Caltech Cars dataset, this paper compares the results with recently state-of-the-art methods proposed by Li et al. [1], Kim et al. [2] and Zhou et al. [23]. As shown in Table 2, the proposed method achieves the best results. More specific, the proposed method achieves precision at 98.43% and recall at 99.21%

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

with IoU at 0.5. Furthermore, the proposed method shows more effective performance when IoU threshold is increased to 0.7. Higher value of IoU means better quality of system. With IoU at 0.7, the proposed method achieves at 96.88% of precision and 98.41% of recall. With new collected dataset, detection ratio is used to evaluate the performance. Table 3 shows detection ratio results with IoU threshold at 0.7. As shown in this table, the proposed method performs well in case of daytime and complex background. In case of nighttime with variety of illumination conditions and small license plate, the proposed method exhibits limited performance because MSER algorithm is sensitive to noise in these conditions. Table 4 shows average processing time of each step in the proposed method. As shown in this table, the proposed method meets the requirement for real-time processing on low-spec machines.

5. CONCLUSIONS

This paper presents a new effective approach for detecting license plate in complex outdoor scenes. The proposed method consists of four steps: vehicle region extraction, character candidate generation, license plate candidate construction and word/no-word classification. SSD framework and MobileNets architecture are used to detect vehicle regions and classify word/no-word candidates. To create license plate candidates, this paper first uses multi-channel MSER algorithm to generate character candidates. Then, false character candidates are eliminated based on the dimension of vehicle regions and then license plate candidates are generated based on remaining character candidates. This paper evaluates the performance on widely used dataset and new collected dataset. Experimental results show that the proposed method achieves better results than other state-of-the-art methods in terms of detection accuracy and run-time efficiency. The proposed method can be implemented in real-time on low-spec machines. However, the proposed method fails when license plate is too small because MSER algorithm is sensitive to noise in this case. Thus, this paper will further improve MSER results by enhancing contrast of image in the future.

REFERENCES:

[1] Li H., Shen C., "Reading car license plates using deep convolutional neural networks and LSTMs", *arXiv preprint* arXiv:1601.05610, 2016.

[2] Kim S.G., Jeon H.G., Koo H.I., "Deep-learning-based license plate detection method using vehicle region extraction", *Electronics Letters*, 2017, pp. 1034-1036.

[3] Rafique M.A., Pedrycz W., Jeon M., "Vehicle license plate detection using region based convolutional neural networks", *Soft Computing*, 2017.

[4] Bulan O., Kozitsky V., Ramesh P., Shreve M., "Segmentation- and Annotation-Free License Plate Recognition With Deep Localization and Failure Identification", *IEEE Transactions on Intelligent Transportation Systems*, 2017, pp. 2351-2363.

[5] Raghunandan K.S., Shivakumara P., Jalab H.A. *et al.*, "Riesz Fractional Based Model for Enhancing License Plate Detection and Recognition", *IEEE Transactions on Circuits and Systems for Video Technology*, 2017.

[6] Yuan Y., Zou W., Zhao Y. *et al.*, "A Robust and Efficient Approach to License Plate Detection", *IEEE Transactions on Image Processing*, 2017, pp. 1102-1114.

[7] Tang Y., Wu X., "Scene Text Detection and Segmentation Based on Cascaded Convolution Neural Networks", *IEEE Transactions on Image Processing*, 2017, pp. 1509-1520.

[8] He T., Huang W., Qiao Y. *et al.*, "Text-Attentional Convolutional Neural Network for Scene Text Detection", *IEEE Transactions on Image Processing*, 2016, 25, (6), pp. 2529-2541.

[9] Jaderberg M., Simonyan K., Vedaldi A. *et al.*, "Reading Text in the Wild with Convolutional Neural Networks", *International Journal of Computer Vision*, 2015, pp. 1-20.

[10] Xie L., Ahmad T., Jin L. *et al.*, "A New CNN-Based Method for Multi-Directional Car License Plate Detection", *IEEE Transactions on Intelligent Transportation Systems*, 2018, pp. 507-517.

[11] Huang W., Qiao Y., Tang X., "Robust scene text detection with convolution neural network induced MSER trees", *European Conf. on Computer Vision (ECCV)*, 2014, pp. 497–511.

[12] Tian S., Lu S., Su B. *et al.*, "Scene Text Segmentation with Multi-level Maximally Stable Extremal Regions", *22nd International Conference on Pattern Recognition*, Stockholm, 2014, pp. 2703-

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

2708.

[13] Qin S., Manduchi R., "A fast and robust text spotter", *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2016 pp. 1-8.

[14] Cho H., Sung M., Bongjin J., "Canny text detector: Fast and robust scene text localization algorithm". *Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 3566–3573.

[15] Liu Z., Li Y., Qi X. *et al.*, "Method for unconstrained text detection in natural scene image", *IET Computer Vision*, 2017, pp. 596-604.

[16] Yin X.-C., Yin X., Huang K. *et al.*, "Robust text detection in natural scene images", *IEEE Trans. Pattern Anal. Mach. Intell.*, 2014, 36, (5), pp. 970–983.

[17] Ye Q., Doermann D., "Text detection and recognition in imagery: a survey", *IEEE Trans. Pattern Anal. Mach. Intell.*, 2015, 37, (7), pp. 1480–1500.

[18] Gou C., Wang K., Yao Y. *et al.*, "Vehicle License Plate Recognition Based on Extremal Regions and Restricted Boltzmann Machines", *IEEE Transactions on Intelligent Transportation Systems*, 2016, pp. 1096-1107.

[19] Li B., Tian B., Li Y. *et al.*, "Component-Based License Plate Detection Using Conditional Random Field Model", *IEEE Transactions on Intelligent Transportation Systems*, 2013, pp. 1690-1699.

[20] Ashtari A.H., Nordin M.J., Fathy M., "An Iranian License Plate Recognition System Based on Color Features", *IEEE Transactions on Intelligent Transportation Systems*, 2014, pp. 1690-1705.

[21] Hsu G.S., Chen J.C., Chung Y.Z., "Application-Oriented License Plate Recognition", *IEEE Transactions on Vehicular Technology*, 2013, pp. 552-561.

[22] Anagnostopoulos C.N.E., Anagnostopoulos I.E., Psoroulas I.D. *et al.*, "License Plate Recognition From Still Images and Video Sequences: A Survey", *IEEE Transactions on Intelligent Transportation Systems*, 2008, pp. 377-391.

[23] Zhou W., Li H., Lu Y. *et al.*, "Principal Visual Word Discovery for Automatic License Plate Detection", *IEEE Transactions on Image Processing*, 2012, pp. 4269-4279.

[24] Sarfraz M.S., Shahzad A., Elahi M.A. *et al.*, "Real-time automatic license plate recognition for CCTV forensic applications", *Journal of Real-Time Image Processing*, 2011, pp. 285-295.

[25] Huang J., Rathod V., Sun C. *et al.*, "Speed/accuracy trade-offs for modern convolutional object detectors", *CVPR*, 2017.

[26] Ren S., He K., Girshick R. *et al.*, "Faster r-cnn: Towards real-time object detection with region proposal networks", *Advances in neural information processing systems*, 2015, pp. 91–99.

[27] Liu W., Anguelov D., Erhan D. *et al.*, "SSD: Single shot multibox detector", *arXiv preprint* arXiv:1512.02325, 2015.

[28] Howard A.G., Zhu M., Chen B. *et al.*, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications", *CoRR*, 2017.

[29] Lin T.-Y., Maire M., Belongie S. *et al.*, "Microsoft COCO: Common objects in context", *ECCV*, 2014.

[30] Dai J., Li Y., He K. *et al.*, "R-FCN: Object detection via region-based fully convolutional networks", *arXiv preprint* arXiv:1605.06409, 2016.

[31] "Caltech Cars 1999 (Rear) 2 dataset", http://www.vision.caltech.edu/Image_Datasets/cars_markus/cars_markus.tar, accessed 2003.

[32] Chu W., Liu Y., Shen C. *et al.*, "Multi-Task Vehicle Detection With Region-of Interest Voting", *IEEE Transactions on Image Processing*, 2018, pp. 432-441.

[33] Yuan X., Su S., Chen H., "A Graph-Based Vehicle Proposal Location and Detection Algorithm", *IEEE Transactions on Intelligent Transportation Systems*, 2017, pp. 3282-3289.

[34] Du S., Ibrahim M., Shehata M. *et al.*, "Automatic License Plate Recognition (ALPR): A State-of-the-Art Review", *IEEE Transactions on Circuits and Systems for Video Technology*, 2013, pp. 311-325.

[35] Kim K., Jung K., Kim J.: "Texture-base approach for text detection in images using support vector machines and continuously adaptive mean shift algorithm", *IEEE Trans. Pattern Anal. Mach. Intell.*, 2003, pp. 1631–1639.

[36] Chen X., Yuille A., "Detecting and reading text in natural scenes", *Proc. IEEE Conf. CVPR*, 2004, pp. 366–373.

[37] Epshtein B., Ofek E., Wexler Y.,

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

"Detecting text in natural scenes with stroke width transform", *Proc. IEEE Conf. CVPR*, 2010, pp. 2963–2970.

[38] Yao C., Bai X., Liu W. *et al.*, "Detecting texts of arbitrary orientations in natural images", *Proc. IEEE Conf. CVPR*, 2012, pp. 1083–1090.

[39] Neumann L., Matas J., "Real-time scene text localization and recognition", *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 3538-3545.

[40] Mishra A., Alahari K., Jawahar C., "Scene text recognition using higher order language priors", *Proceedings of the British Machine Vision Conference*, 2012.

[41] Gupta A., Vedaldi A., Zisserman A., "Synthetic Data for Text Localisation in Natural Images", *CVPR*, 2016.

[42] Wang Q., Lu Y., Sun S., "Text detection in nature scene images using two-stage nontext filtering", *International Conference on Document Analysis and Recognition (ICDAR)*, 2015, pp. 106-110.

[43] Zhen D., Wu Y., Pei M. *et al*., "Vehicle Type Classification Using a Semisupervised Convolutional Neural Network", *IEEE Transcations on Intelligent Transportation Systems*, 2015.