International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-9 Issue-6, April 2020

Real-time License Plate Recognition in Overweight Vehicle Balance System

Huong-Giang Doan

Abstract: ***Recently, license plate recognition has been become an attractive field in computer vision. Which consists some main steps such as: data collection, plate detection, character separation, character segmentation, characters recognition and character series connection. Many state-of-the-art methods have been proposed while almost these approaches utilize complex algorithm. That spends a large time cost to obtain competitive accuracy; and/or high equipment performance such as CPU, GPU, cameras and so on. In addition, almost recent methods have been not deployed and evaluated for an end-to-end real application. Such system still has to face with many challenges due to the time cost, accuracy of system, complex background, light condition, motion blur and so on. In this paper, we propose a new framework for deeply evaluate efficient of license plate recognition system. Then a real application is deployed in the overweight balance system. In this application, the license plate recognition system is integrated as a middle step in order to reduce not only labor but also automatic for an industrial balance system.***

Keywords: ***License Plate Recognition, Deep Learning, Character Recognition, License Plate Detection, Number Plate Detection, Optimize Tree.***

I. INTRODUCTION

In recent years, license plate recognition (LPR) has become a great attention of researchers thanks to its potential applications such as check in/out, car park, automatic ticket vehicle control, traffic, automatic electronic balance. In many last proposed methods, community researchers are concentrated on counting car/vehicle [11][20] on the way, identification [1] and tracking [12], and so on. Which are sensitive with light condition, objects are immobile as well as motion blur. Such methods have been proposed for LPR recognition such as [2][3][4][19][20]. Corresponding with the rapidly increasing in number of car/vehicle may be appeared more and more over weight on the street. This lead to not only require too labors in order to manual check and import license at overweight station, but also reduce accident for workers. There has been some previous researches about LPR [13][14][15][19]. Moreover, some cases concentrated to resolve separation steps (e.g. Plate detection [13], Character spotting [14], Number recognition [18]).

Revised Manuscript Received on April 30, 2020.
***** Correspondence Author
Huong-Giang Doan*, Control and Automation Faculty, Electric Power University, Ha Noi, Viet. Email: giangdth@epu.edu.vn

Which is difficult to integrate into complete system, others is high computation time cost to obtain high accuracy, or the proposed method require high performance of pre-processing system [1][19] (e.g. high quality camera, speed CPU/GPU,…), and others cases are not suitable with some special constrains of overweight systems. In [20], authors used CNN in two steps in LPR system as plate detection and character recognition. Moreover, they utilized thresholds as pre-detection step of plate, character separation. Although some license plate dataset is published such as [3][16]. In [16] authors proposed method to recognize license plate in Myanmar, [19] recognized number license of Indian. Furthermore, it has not a public license plate dataset of car/vehicle in Vietnam. That has three main types as illustrated in Fig. 5 (a), (b) and (c). In this paper, a solution is proposed to integrate into an overweight system. That obtains not only high accuracy but also real-time license plate recognition system. In addition, the system is deployed in normal PC (no GPU requirement) as well as spends low time cost. The results must be tried to compare with some state-of-the-art methods. Moreover, a license plate dataset is also captured and published for research community.

The remaining of this paper is organized as follows: Section II describes our proposed approach. The experiments and results are analyzed in Section III. Section IV concludes this paper and recommends some future works.

II. PROPOSE METHOD

The main flow-work for license plate recognition from RGB image consists of a series of the cascaded steps as shown in Fig. 1. Given RGB images, license plate region are detected, extracted, spotted digits and recognized. The steps are presented in detail as the next sections following:

A. Collection and Pre-processing data

RGB images (I) are captured by the camera of IPhone 5 (640x480 pixels) as showed in Fig. 2. Distance between camera to license plate of car about 1 meter. Plates are face to face with camera. 1100 images of car/vehicle in Vietnam are captured. That 1000 RGB images are manually labeled the ground true and named LicenseCAR_EPU1 dataset. This dataset is utilized to evaluate our propose method as showed in the middle and bottom panels of Fig. 1. Additionally, 100 images are chosen to prepare the training data for the top part of Fig. 1. The training data is manual segmented as presented in Fig. 3 following. Firstly, given plate image (Fig. 3(a)), RGB images of digits are cut out with different sizes (Fig. 3(b)). Then, they are converted to binary images resized to the same size (25x50pixels). It is labeled as showed in Fig. 3(c).

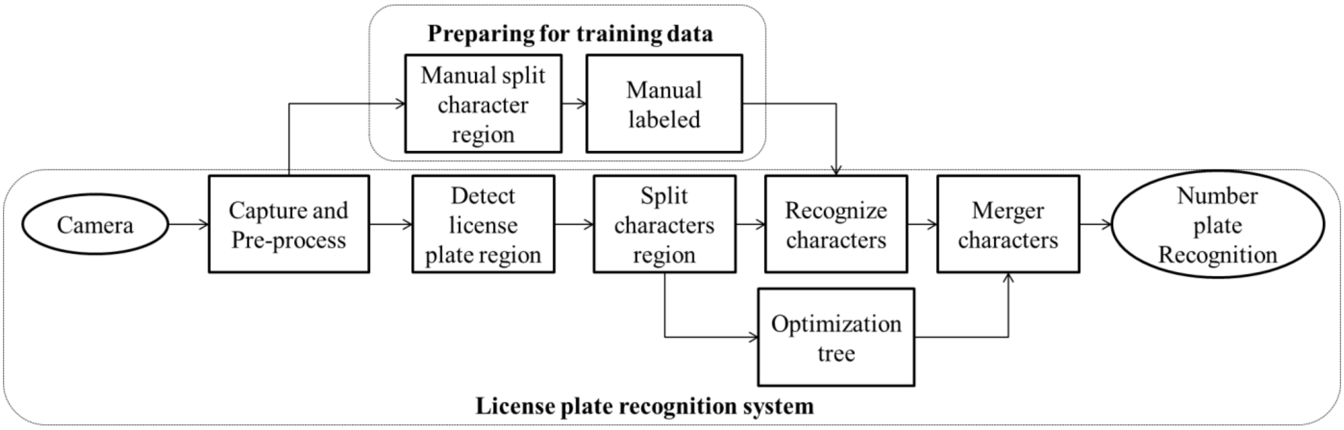


Fig. 1.Propose framework for license plate recognition

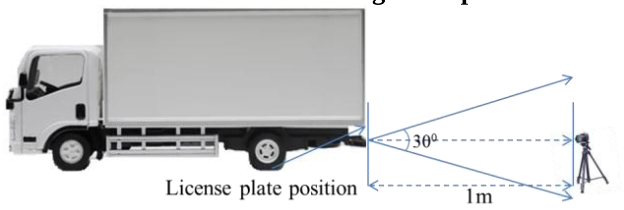


Fig. 2.Setup capture environment

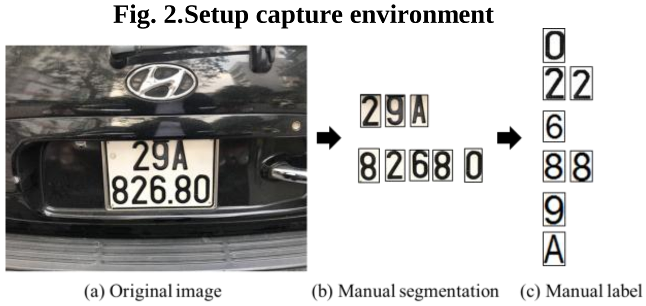


Fig. 3.Manual preparing data for trainning

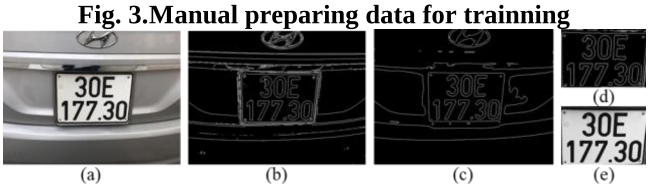


Fig. 4.License plate region detection

B. Detect license plate and spotting character region

Detect license plate region: Given an input image of behind any car $I(x,y)$ (Fig. 4(a)), a Laplace mask is applied to detect edges of objects in image $I\,(I_E(x,y))$ (Fig. 4(b)). Then, connected arcwises in $I_E(x,y)$ are remained while opened arcwises are removed out of $I_E(x,y)$ (Fig. 4(c)). Next, candidate license plate regions ($I_B^i(x,y); (i=1\div n)$) are created from connected arc-wises (Fig. 4(b,c)).

In license plate region consists three categories as showed in Fig. 5: First type in Fig. 5(a) with background is white and foreground is black. Second type as showed in Fig. 5(b) with background is blue and foreground is black and at Fig. 5 (c), background is red and foreground is black. Those types are checked and converted into the same binary image as show in Fig. 5(d), (e) and (f). Then, four heuristics are utilized to choose license plate region exactly, such as: (1) Ratio between height and width of regions ($S_{I_i^B}=\dfrac{H_{I_i^B}}{W_{I_i^B}}<1$); (2) The numbers of edges in region ($N_{I_i^B}=4$); (3) The ratio between background and foreground ($0,3<P_{I_i^B}=\dfrac{Fo_{I_i^B}}{Ba_{I_i^B}}<0.8$); and (4) histogram of the gray license plate region in x and y dimension.

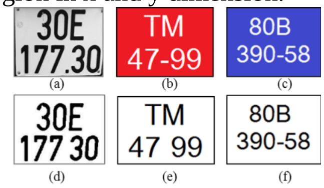


Fig. 5. Normalization for license plate types

Spotting character region: Binary license plate (HxW pixels) is existed the sub-connected regions $(I_j^D;\ j=(1\div m))$. Moreover, a sub-connected area is chosen when: (1) Density of black dots in region belongs (0.3 ÷ 0.7). It is that mean its envelope has neither a large circumference nor too small; (2) Height and width of sub-connected regions j (h_j^{sub}, w_j^{sub}) is satisfied following condition:

$$I_j^D = Digit\ if \begin{cases} w_j^{sub} < \alpha * W \\ \beta * H < h_j^{sub} < \gamma * H \end{cases} \quad (1)$$

(α,β,γ) are thresholds which are chosen with $(\alpha=\frac{1}{8};\beta\frac{1}{3},\gamma=\frac{4}{5})$. After all digits are spotted which are different size as illustrated in Fig. 6 (b). These images (I_j^D) are then resized to the same dimension ($(wxh)=$ 25x50 pixels) as showed in Fig. 6 (c).

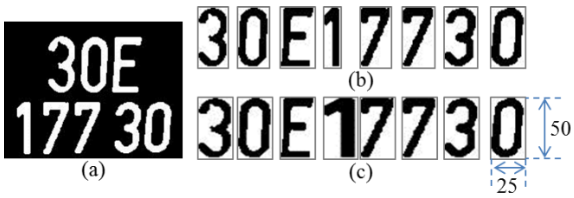


Fig. 6.Detect and separate digit regions

C. Character recognition

In this section, series of digit images $I_k^D (k = 1 \div v)$ with $(v \le m)$ are then represented by three methods: All pixels of a digit; feature extractor and using Convolutional Newron Network as presented Sec. 2.3.1. Then, digits are recognized as in Sec. 2.3.2. In the next section, optimization tree is ultilized in order to recognize digits as showed detail in Sec. 2.3.3.

1) Feature representation

Raw data representation: Given $I_k^D (w,h)$; $(w = 25; h = 50)$, all pixels of a digit image are utilized to present feature vectors as showed in equation (2) following:

$$F_{I_k^D}^{(1)} = [1, wxh]^T = [1, 25x50]^T = [1,1250]^T \quad (2)$$

Hand craft feature extraction: In this part, some state-of-the-art descriptors are used to extract features of digit images as: SIFT [7], SURF [8], HOG [9] and KDES [10]. By using those corresponding descriptors, the number of most important key points of a spotted digit image (Fig. 6. (c)) are detected with $F^{(2)} = F_{Sift}$; $F^{(3)} = F_{Surf}$; $F^{(4)} = F_{HOG}$ and $F^{(5)} = F_{KDES}$ as presented in equations (3), (4), (5) and (6) following:

$$F_{I_k^D}^{(2)} = F_{SIRF} = [K_1^{(2)} \quad ... \quad K_{100}^{(2)}]^T \quad (3)$$

$$F_{I_k^D}^{(3)} = F_{SURF} = [K_1^{(3)} \quad ... \quad K_{100}^{(3)}]^T \quad (4)$$

$$F_{I_k^D}^{(4)} = F_{HOG} = [K_1^{(4)} \quad ... \quad K_{100}^{(4)}]^T \quad (5)$$

$$F_{I_k^D}^{(5)} = F_{KDES} = [K_1^{(5)} \quad ... \quad K_{128}^{(5)}]^T \quad (6)$$

Deep learning feature extraction: Recently, deep learning (CNN) has been widely used in computer vision in various tasks such as feature extraction, recognition, identification. E.g. deep neural networks for license plate recognition [1][2][3][19][20]. In [619], although CNN network is used to extract feature of both spatial and temporal information but the whole video sequence are compacted into three Depth maps. Therefore, input of this deep learning network is a single image. In this paper, this network is firstly employed to train parameters of model. This model is then utilized to extract feature of digit images. This convolutional neural network composes of 8 convolutional layers, 4 max pooling and 2 fully connected layer followed by a soft-max output layer. The architecture of designed network is illustrated in Fig. 7 following:

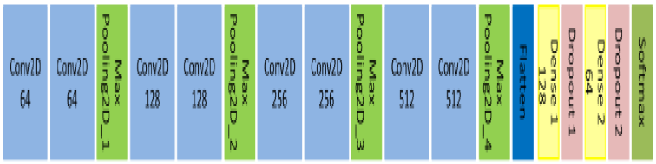


Fig. 7.The CNN architecture for digit extractor.

The images $I_k^D (w,h); k = (1 \div v)$ where w, h are width and height of normalized digits. Which images are utilized as inputs of this convolutional neuron network. The images will be passed to CNN to extract consequence digit features. In this research, this network is used as feature extractor. The dimension of output feature is taken at last layer of network and the feature size is presented by $F^{(6)} = F_{CNN}$ as equation (7) following:

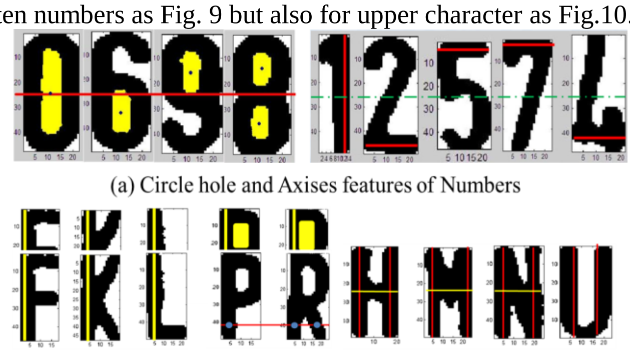
$$F_{I_k^D}^{(6)} = F_{CNN} = [K_1^{(6)} \quad ... \quad K_{64}^{(6)}]^T \quad (7)$$

2) Digit classification

The features $F^{(1)};...;F^{(6)}$ in Sec. C. (1) are utilized as the input of the SVM classifier [6]. That is utilized as input of multi-class SVM classifier. The output of multi-class SVM will be one number value among {0,1,2,…,9} and upper character format {A,B,C,…,Z} which contains total 37 classes of digits. Furthermore, $F^{(1)}$ also is passed over others classifier (KNN and Naïve Bayes). The results are presented detail in Sec. III following.

3) Optimization tree

Firstly, numbers have difference features such as illustrated in Fig. 8 following. Numbers: 0, 6, 9 and 8 contain circle hole that inverse with 1, 2, 5, 7 (axis position of numbers). In addition, hole and axes position are not the same together. Seminar to upper character as showed in Fig. 8 (b). In this paper, optimization tree is proposed to recognize not only for ten numbers as Fig. 9 but also for upper character as Fig.10.



(a) Circle hole and Axises features of Numbers

(b) Circle hole and axises features of upper digits

Fig. 8.Features of Number and Upper Digits in License Plate

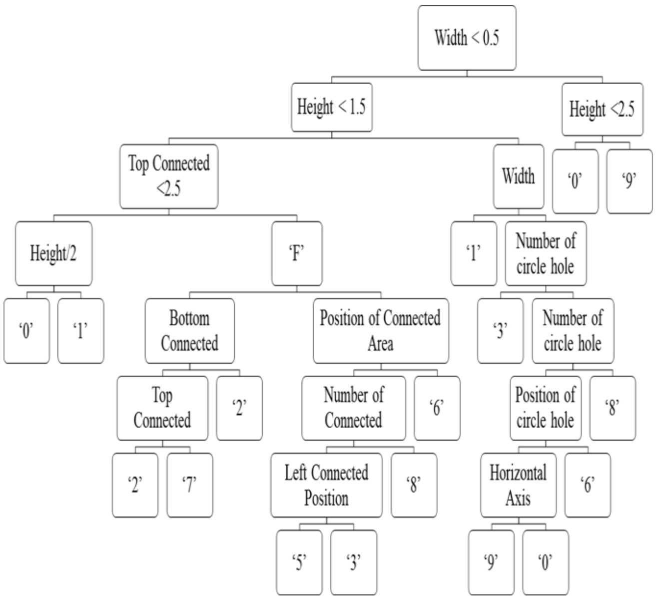


Fig. 9.Optimization Tree of Number

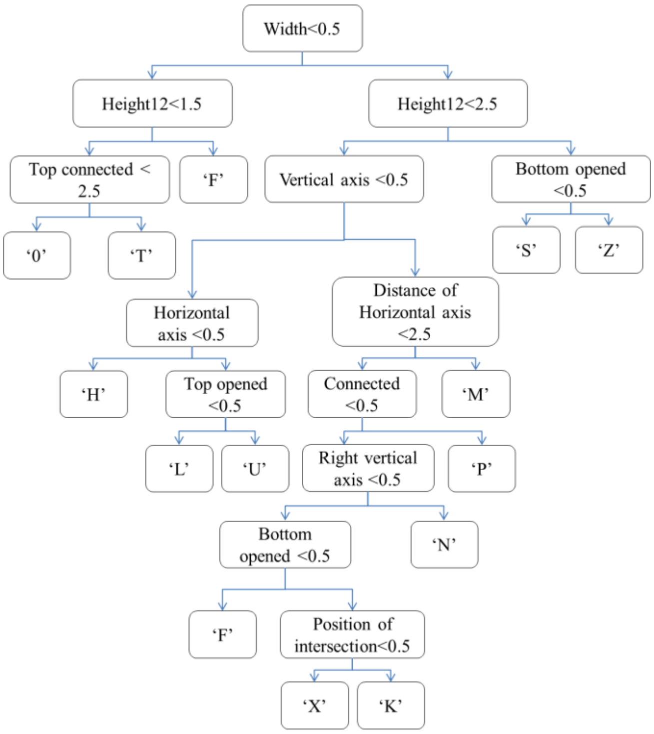


Fig. 10. Optimization Tree of Upper Character

III. RESULT AND DISCUSSION

The proposed framework is warped by Matlab program and Python program on a PC Core i6 4.2 GHz CPU, 8GB RAM. We evaluate performance of the license plate recognition on LicenseCAR_epu1 dataset. Four evaluations are considered such as: (1) How is the better feature representation; (2) Comparison of accuracy recognition rate between hand-craft feature method, deep learning feature method and Optimization method; (3) Compare time cost of strategies; (4) Real application. The detail evaluations are presented as following sub-sections:

A. Evaluation of plate recognition rate on different feature representations

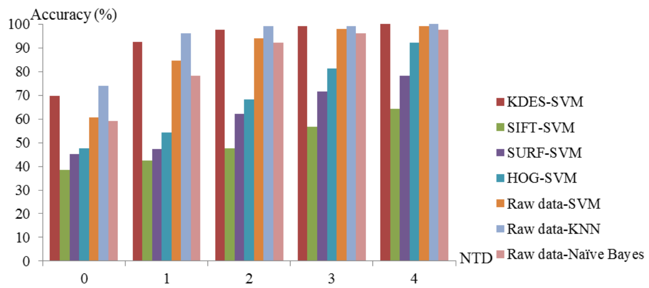


Fig. 11. Accuracy with the different feature representations

In this evaluation, we test the accuracy rate of various hand craft feature representation with SVM classifier. LicenseCAR_epu1 dataset is used in this work. The accuracy is evaluated at different NTD value (Number True Digit) of each license plate (while each number plate may be has 7 to 9 digits). In this test, NTD is only considered from 0 (all digits is true) to 4 (four digits are fail). A glance at the Fig. 8, it is apparent that the KDES-SVM and Raw data – KNN obtain the best percentage while SIRF-SVM is lowest accuracy in overall. Look at the figure, the highest values stood at 69.88% and 74.09% for KDES-SVM and Raw data – KNN. While accuracy of SIRF-SVM method is only 38.58% with NTD equals 0. Furthermore, the license plate recognition of the Raw data obtains higher then feature extractors in all cases.

B. Comparison of hand-craft feature representations and deep learning feature

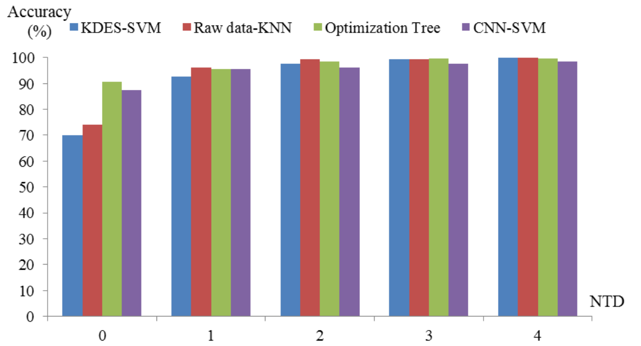


Fig. 12. Evaluation with the different classifiers

Given the Fig. 9 that shows a comparison of recognition accuracy of five various methods at four NTD values. It is evident from the information provided that Optimization Tree has by far the highest percentage in almost cases except at NTD value equals 1. That Optimization Tree accuracy is slightly lower CNN method at 95.56% and 96,09% respectively. This evaluation illustrates that, the deep learning method is not efficient at small training data over the period show. In addition, the two highest hand-craft feature methods (KDES-SVM and Raw data - KNN as show in Sec. III. A) are far lower than CNN and Optimization Tree methods at NTD = 0. Also note worth is fact that Optimization Tree method is more likely to deploy a real application.

C. Comparison of time cost

Table 1. Evaluate time cost on different methods

Methods	Time cost (s)	Accuracy (%)
KDES - SVM	3.56	69.88
Raw data - SVM	5.55	50.78
Raw data - KNN	4.84	74.09
Raw data - Naïve bayes	3.21	59.07
Optimization Tree	1.86	90.58
CNN	6.71	87.51

A glance at the table provided reveals time cost average and recognition accuracy of methods when all digits in a license plate are detected and recognized during the period shown. It could be seen from the Tab. 1 that the proposed method obtains the best hand gesture recognition accuracy, with the highest value at 90.58% on LicenseCARepu1 dataset dataset and time cost for entire proposed recognition system require the lowest value at 1.86 (sec). The remain methods accounted by far with the highest time cost belongs CNN method up to 6.71 sec and accuracy is 87.51%. The deep learning method needs high CPU performance (or GPU card) as well as a large training dataset. It is quite difficult to deploy a real application.

D. Real application

Given best result of optimization tree method as presented in previous Sec. III, in this part, a real application is deployed as showed in Fig. 13 following. That integrates license plate recognition to manage in the overweight vehicle balance system. The license plate recognition result is taken exactly and automatically that is become one field of data management. Four fields of database (e.g. License plate, Name of Driver, Date and Time, Weight Value) are connected directly with SQL server. While license plate recognition system is return result after controller push into "Capture Data" button and then "License Plate Recognition" button. Weight value is automatically send from sensors to processing equipment and then transfer to PC through RS232 cable. When all four fields of data are inserted, "Add Violate Vehicle" button is enable. In addition, statistic function could be also implemented for any fields of this data by "Statistic Function" button of software.

Fig. 13. Interface of License Plate Recognition Integration Aplication

IV. CONCLUSION

In this paper, an approach for entire license plate recognition system that has integrated into a real overweight vehicle balance application. This paper has deeply investigated the results of with some state-of-the-art methods such as combination between feature extractors (SIRF, SURF, KDES, HOG and CNN) and classifiers (SVM, KNN, Naïve Bayes). Experiments were conducted on our captured dataset. The evaluations lead to some following conclusions: i) Concerning optimization tree issue, the proposed method has obtained highest performance. It is simple approach and obtains real-time system. So one of recommendation is to combine between features of characters as well as create larger training dataset to obtain the higher accuracy of license plate recognition; ii) The proposed method has only experimented on Vietnam vehicle that has been not tested on other published datasets.

REFERENCES

1. Prajwal M. J., Tejas K. B., et.al. *Detection of Non-Helmet Riders and Extraction of License Plate Number using Yolo v2 and OCR Method*, International Journal of Innovative Technology and Exploring Engineering (IJITEE), Vol. 9, Issue-2, pp. 5167-5172. 2019.
2. Jain, Vishal, et al. *Deep automatic license plate recognition system*. The Proceedings of the 10th Indian Conference on Computer Vision, Graphics and Image Processing. ACM, 2016.
3. Caltech dataset: http://www.vision.caltech.edu/html-files/archive.html
4. M. Jaderberg, A. Vedaldi, and A. Zisserman. *Deep features for text spotting*. In Procceding of ECCV, pp. 512-528,2019.
5. Huong-Giang Doan and Van-Toi Nguyen, *Improving Dynamic Hand Gesture Recognition on Multi-views with Multi-modalities*, International Journal of Machine Learning and Computing, Vol. 9, No. 6, pp. 795-800, 2019.
6. C. 1. C. Burges, *A Tutorial on Support Vector Machines for Pattern Recognition*, Vol. 43, pp. 1-43, 1997.
7. David G Lowe, *Method and apparatus for identifying scale invariant features in an image and use of same for locating an object in an image*, David Lowe's patent for the SIFT algorithm, 2004.
8. Bay, Herbert & Tuytelaars, Tinne & Van Gool, Luc. *SURF: Speeded up robust features*. The Proceedings of the 9th European conference on Computer Vision-ECCV, pp. 404-417. 2006.
9. Navneet Dalal, Bill Triggs. *Histograms of Oriented Gradients for Human Detection*. International Conference on Computer Vision & Pattern Recognition (CVPR '05), United States. pp.886--893, 2005.
10. Bo, Liefeng, Xiaofeng Ren, and Dieter Fox. *Kernel descriptors for visual recognition.* In Advances in neural information processing systems, pp. 244-252. 2010.
11. Crouzil, Alain & Khoudour, Louahdi & Valiere, Paul & Truong, Nghi. *Automatic Vehicle Counting System for Traffic Monitoring*. Journal of Electronic Imaging (JEI),pp. 051207(1) - 051207(12). 2016.
12. Pena-Gonzalez, Raul & Nuno, Marco. (2014). *Computer vision based real-time vehicle tracking and classification system*. Midwest Symposium on Circuits and Systems. pp. 679-682. 10.1109/MWSCAS.2014.6908506.
13. Y. Qiu, M. Sun, and W. Zhou. (2009). *License plate extraction based on vertical edge detection and mathematical morphology*. In The Proceeding of International Conference Computer Intelligence Software Engineer pp. 1-5. 2009.
14. M. Jaderberg, A. Vedaldi, and A. Zisserman. (2014). *Deep features for text spotting*. In The Proceeding of ECCV. pp. 512-528. 2014.
15. W. Zhou, H. Li, Y. Lu and Q. Tian. (2012). *Principal Visual Word Discovery for Automatic License Plate Detection*. In IEEE Transactions on Image Processing, vol. 21, no. 9, pp. 4269-4279. 2012.
16. Kyaw, Nwe & Sinha, Professor G & Mon, Khin. *License Plate recognition of Myanmar Vehicle Number Plates A Critical Review*. pp. 771-774.2018.
17. de Campos, Teofilo & Babu, Bodla & Varma, Manik. *Character Recognition in Natural Images*. VISAPP 2009 - Proceedings of the 4th International Conference on Computer Vision Theory and Applications. pp. 273-280.
18. O. Akbani, A. Gokrani, M. Quresh, et.al. Character Recognition in Natural Scene Images. *2015 International Conference on Information and Communication Technologies*. pp. 1-6. 2015.
19. C.C.Chowdary, U.Kiran. *An Approach towards Detection of Indian Number Plate from Vehicle,* International Journal of Innovative Technology and Exploring Engineering (IJITEE), Vol. 8, Issue-9S3, pp. 1432-1436. 2019.
20. Selmi, Zied & Ben Halima, et.al. *Deep Learning System for Automatic License Plate Detection and Recognition*. The 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), pp.1132-1138, 2017.

AUTHOR PROFILE

Huong-Giang Doan, received B.E. degree in Instrumentation and Industrial Informatics in 2003, M.E. in Instrumentation and Automatic Control System in 2006 and Ph.D. in Control engineering and Automation in 2017, all from Hanoi University of Science and Technology, Vietnam. She is a lecturer at Control and Automation faculty, Electric Power University, Ha Noi, Viet Nam. Her current research centers on human-machine interaction using image information, action recognition, manifold space representation for human action, computer vision