

GPU Accelerated Number Plate Localization in Crowded Situation

Adhi Prahara^{a,1,*}, Andri Pranolo^{a,2}, Rafał Dreżewski^{b,3}

^aInformatics Department, Universitas Ahmad Dahlan, Umbul Harjo, Yogyakarta 55164, Indonesia

^bDepartment of Computer Science, AGH University of Science and Technology, Kraków, Poland¹

¹adhi.prahara@tif.uad.ac.id, ²andri.pranolo@tif.uad.ac.id, ³drezew@agh.edu.pl

ARTICLE INFO

Article history:

Received September 15, 2015

Revised October 07, 2015

Accepted November 30, 2015

Keywords:

Number Plate

Plate Localization

Vertical Edge Detection

HOG

SVM

GPU

ABSTRACT

Number Plate Localization (NPL) has been widely used as part of Automatic Number Plate Recognition (ANPR) system. NPL method determines the accuracy of ANPR system. Although it is a mature research, the challenge still persists especially in crowded situation where many vehicles are present. Therefore, a method is proposed to localize number plate in a crowded situation. The proposed NPL method uses vertical edge density to extract potential region of number plate and then detects the number plate using combination of Histogram of Oriented Gradients (HOG) and Support Vector Machine (SVM). The method employs GPU to deal with multiple number plate detection, to handle multi-scale detection window, and to perform real time detection. The experimental results are very promising, 0.9883 value of AUC (Area Under Curve), and 0.9362 of BAC (Balance Accuracy). Moreover, potential real time detection is foreseen because total process is executed in less than 50ms. Errors are mainly caused by background that contain letters, non-standard number plate and highly covered number plate.

Copyright © 2015 International Journal of Advances in Intelligent Informatics.

All rights reserved.

I. Introduction

Number Plate Localization (NPL) is very important in Automatic Number Plate Recognition (ANPR) system. ANPR system is used for reading a vehicle registration plates on image or video. ANPR system can be found in toll gate payment system and traffic monitoring system. In ANPR system, NPL stage determines the accuracy of number plate detection. In research presented at [1-5], localization accuracy of number plate has been achieved quite well especially by using edge detection method combined with morphological operations. Edge detection method is fast in the term of processing time. However, it is mostly used to detect single or multiple number plates in normal condition. In a crowded situation such as in the traffic or parking area, many edges will be found. It will make the edge detection algorithm not sufficient to be used as a primary detector.

Some problems that can be found in a crowded situation are multiple number plates with variety of size, variations in lighting, non-standard / modified number plates, tilted number plates, and the traffic density that make a part of number plate covered with other objects. Moreover, number plate localization requires a fast and accurate method to detect number plate. Therefore, a method is proposed to localize number plate in crowded situation by using vertical edge density (VED) and a combination of histogram of oriented gradients (HOG) and support vector machine (SVM) running on Graphical Processing Unit (GPU). GPU is used to accelerate the localization method with its parallel processing. With GPU support, number plate localization is expected to be processed in real time.

II. Related Works

Many researchers had developed NPL methods and achieved good result. Zahedi and Salehi used SIFT (Scale Invariant Feature Transform) for license plate recognition (LPR) [1]. They extracted SIFT on sample image and target image then compared the key points to detect license plate. Test result

showed that SIFT alone could detect license plate with accuracy up to 84%. The error was mainly caused by key points coming from the background. To increase the detection rate, they used preprocessing step by calculating density of vertical edge to crop the license plate candidate. By using preprocessing step before matching with SIFT, the recognition rate increased to 89%.

Mai *et. al* used edge based and morphological operation to locate license plate [2]. The system combined morphological and image subtraction operation on grayscale image to reduce license plate candidate area. They did image binarization based on threshold, edge detection using Canny, and morphological operation on binary image to remove small objects. They calculated the angle of license plate (LP) and then rotating license plate based on Radon transform and bilinear interpolation. They cut the license plate region based on measuring properties of Vietnam license plate regions. The test result showed 97.27% accuracy.

Tarabek presented a robust algorithm for license plate detection that could detect multiple license plates with various sizes in complex backgrounds [3]. The system used vertical edges and edge density features to find candidate license plate regions. The candidates were selected based on geometrical and textural properties. The efficiency of the method was improved using the integral edge image and two-stage candidate window detection. The results showed robustness and efficiency of proposed method.

Al-Ghaili *et. al* proposed a fast method of a car license plate detection [4]. They proposed a fast vertical edge detection algorithm (VEDA) based on the contrast between the grayscale pixels. VEDA was applied to detect vertical edges on previously enhanced image by using an unwanted-line elimination algorithm (ULEA) after binarizing the input image using adaptive thresholding (AT). They detected license plate by extracting the candidate region based on statistical and logical operations. The results showed accurate edge detection performance and faster processing than Sobel by five to nine times.

Dehshibi and Allahverdi used multiclass AdaBoost to detect Persian license plate [5]. The system consisted of plate localization, normalization, and license plate recognition. The highlighted step was multiclass AdaBoost to perform license plate localization. They used morphological operations before applying AdaBoost with haar-like features. The classifier was a ten-layer cascade of AdaBoost classifiers which included a total of 1800 features. The plate localization method had 96.93% accuracy on type 1 of Persian license plate.

The proposed method uses vertical edge density (VED) to extract candidate area of number plate and eliminate most of the background. Histogram of Oriented Gradients (HOG) is extracted on the candidate area and then used as input for Linear-Support Vector Machine (Linear-SVM) to be classified into number plate and non-number plate category. Edge-based detection technique is used because it is easy, straightforward [6] and resulted in good performance as mentioned in [1-4]. The combination of HOG and SVM is widely used in detection system and has a good performance [7] [8]. HOG is one of strong texture descriptor and has its own mechanism to deal with lighting as mentioned in [7]. Machine learning classifier gives good accuracy to deal with various condition. SVM is chosen because it has high tolerance level to new data sample, easy to tune the parameter, and need small data training.

III. Methodology

The GPU accelerated number plate localization in crowded situation system uses combination of two localization methods and uses GPU as computing unit as shown in Fig. 1. In order to perform calculation in GPU, input image needs to be uploaded from CPU to GPU. The first stage of the method is localization of number plate region using vertical edge density. Input image is converted to grayscale then vertical edges are extracted using Sobel filter. To calculate the edges density, a box filter is used. Vertical edge density localized the region of number plate by removing most of the background with less density of vertical edges. Number plate region is determined by selecting area that has edge density above the given threshold.

At the second localization stage, HOG descriptors are extracted from the number plate region. Multi scale window is set up to perform multi window detection on number plate image. Detection is done using Linear-SVM with HOG features as an input and number plate category as an output.

The proposed method was built using Visual Studio 2010 with C++ language in addition of OpenCV 2.4.10 library with GPU support and NVidia CUDA Toolkit 6.5. Method runs on laptop with Intel Core i5 processor, NVidia GT 450M 2GB, and 8GB of RAM. All the methods use OpenCV library.

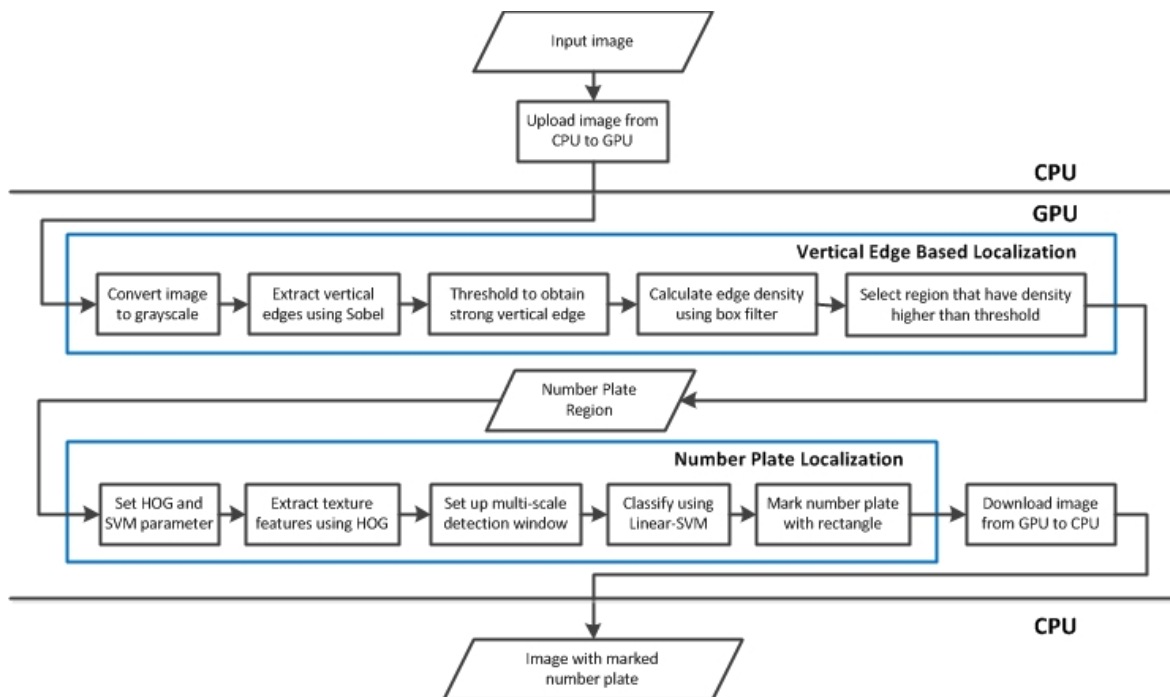


Fig. 1 Number Plate Localization (NPL) method

A. Dataset

The dataset is a collection of traffic and parking area images from the internet and observation taken in daytime. The samples contain many vehicles with reasonable size and orientation of number plate that can be seen clearly by human. Training data are number plate cropped manually from training images and collection of negative data obtained from traffic and parking area environment in various size and orientation. Test data are images of vehicles in the street or parking area.

B. General Purpose-Graphic Processing Unit (GP-GPU)

Most of graphics hardware is reinforced with Graphic Processing Unit (GPU). GPU is a high-level parallel architecture used to do fast operation in computer graphics. Usually GPU is used to convert numbers to image (computer graphics) and now can be used to convert image to numbers (computer vision). That process is known as GP-GPU (General Purpose-GPU) which used graphics card to do computation other than graphics [9]. GPU approach 2D image processing as 3D processing. Simply put, a quadrilateral polygon aligned to the area box on the screen and the image is rendered into the screen or as a buffer in memory. GPU executes many pixel shaders on floating-point image each time in parallel. This allows flexibility and precision pixel operations to be executed on the GPU without any lost in image quality.

C. Vertical Edge Density

Number plate area contains abundant of edges that is why edge-based technique is very effective to localize number plate. Compared to horizontal edge, vertical edge on number plate is stronger. Vertical edge is extracted using Sobel filter with vertical kernel operator [6]. By using vertical edge extracted by Sobel filter, a box filter is applied to calculate the local edge density. The region with edge density value greater than threshold is considered as region of number plate.

D. Feature Extraction

Histogram of Oriented Gradients (HOG) is used for feature extraction on number plate region. According to [7] in order to extract HOG descriptors on image, first we normalize gamma and color from input image then calculate the gradient. For color image, gradients are computed for each color channel, then the one with the largest norm is taken as the pixel's gradient vector. Image is divided

into small spatial regions called cells. For each cell, accumulating a local 1-D histogram of gradient directions or edge orientations over the pixel of the cell. Orientation bins are evenly spaced over 0-180 degree (unsigned gradient) with 20 degree for each space. The nine orientation bins are filled with the weighted magnitude for each orientation. The cells is organized into overlapping blocks and perform contrast normalization on the overlapping blocks for better invariance to illumination and shadowing. A set of features is obtained by collecting features on all blocks.

E. Classification

Support Vector Machine is used as classifier to detect number plate. SVM is a classifier that determined by separator called hyperplane. Hyperplane can be calculated by maximizing margin or distance from two set of object of two different class. Classification with SVM consist of training and classification. In the training step [10], if there are given training data $x_i \in R^n, i = 1, \dots, l$ in two classes and label $y \in R^l$ such that $y_i \in \{1, -1\}$, it can be solved with (1).

$$\min_{w,b,\xi} \frac{1}{2} w^T w + C \sum_{i=1}^l \xi_i \quad (1)$$

$$\text{subject to } y_i(w^T \phi(x_i) + b) \geq 1 - \xi_i, \xi_i \geq 0, i = 1, \dots, l$$

where w is weight vector, b is bias, ξ_i is slack variables, $\phi(x_i)$ maps x_i into higher dimensional space, $C > 0$ is the regularization parameter and the decision function is shown in (2).

$$\text{sgn}(w^T \phi(x) + b) = \text{sgn} \left(\sum_{i=1}^l y_i \alpha_i K(x_i, x) + b \right) \quad (2)$$

where $K(x_i, x_j)$ is the kernel function. After training process, parameter $y_i \alpha_i \forall i$, b , label names, support vectors, and kernel parameter saved as output model from training SVM.

In classification, voting strategy is performed for each data x which will be designated to be in a class with the maximum votes [10]. Optimal parameter is selected using k -fold cross validation which is a method to do cross validation by dividing training data into k set which has $(k-1)$ as training data and the rest will be the test data. After training process we will get variable w , x , and b for each class, then the classification process can be done with these steps:

1. Calculate kernel.
2. Calculate decision function with (2).
3. Repeat step 1 and 2 for other classes.
4. Determine the class by function which gives the most maximum votes (the member of class is determined by voting).

IV. Results and Discussion

The GPU accelerated number plate localization in crowded situation method is tested using 15 images of vehicles in traffic or parking area. The training data consists of 500 positive images and 1570 negative images. Positive training data are enriched with six orientations, 5°, -5°, 10°, -10°, 15°, and -15° and negative training data are enriched with three orientations, flip horizontal, flip vertical and flip both vertical and horizontal. The total training data will contain 3500 positive images and 6280 negative images.

A. Vertical Edge Based Localization

Number plate has strong vertical edge coming from the number printed on the plate. However, in a crowded situation, vertical edge density cannot become the primary detector because many vertical edges will be found and the localization method will fail to locate number plate accurately. Therefore, vertical edge density is only used to localize region of number plate by removing most of the background that does not have enough density of vertical edges.

Vertical edge, edge density, and localization of number plate are shown in Fig. 2, Fig. 3, and Fig. 4 respectively with two samples (a) and (b). Weak to strong vertical edge density in Fig. 2 is visualized by blue to yellow color in Fig. 3. Number plate region has strong edge as it is represented by yellow color. Although many strong edges are present, this step is still important because it can decrease the

false positive coming from classification by HOG+SVM as shown in Fig. 4. Blue rectangles marker in Fig. 4 indicate the predicted number plate by HOG+SVM and the red marker is the final marker obtained from the combination of vertical edge density and HOG+SVM.

GPU implementation of vertical edge based localization uses two filters, Sobel filter and box filter. Sobel filter with 3x3 kernel is used to extract vertical edge on image. Strong vertical edges are selected from vertical edges that have value greater than 128. A normalized box filter with 32x24 kernel is used to calculate density of vertical edge. Because of the large kernel, box filter is applied with separate kernel method to decrease the processing time in GPU. The high density of vertical edges is selected using threshold obtained from the sum of vertical edge values divided by total amount of non-zero values in the edge image.

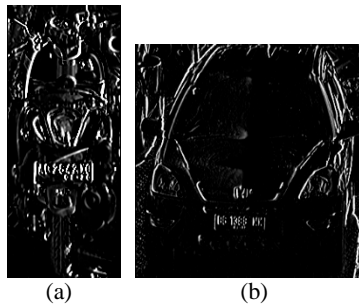


Fig. 2 Vertical edge

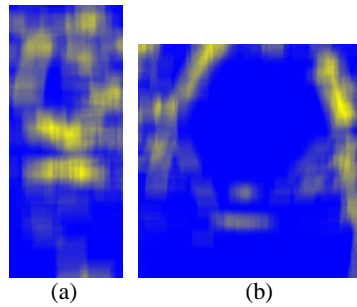


Fig. 3 Edge density

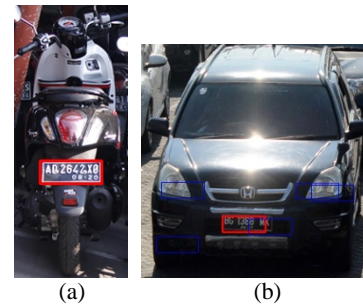


Fig. 4 Number plate localization

B. Number Plate Localization

HOG is extracted from the candidate number plate region using 64x24 detection window, 8x8 cell size, 16x16 block size, and 9 bin size with gamma correction. The detection window is coming from the fact that traffic image is usually taken from far distance. The minimum size of detection window is 64x24 that satisfies 3:1 ratio of number plate size, can be divided by HOG's cell size and it is greater than HOG's block size. Number plate and its HOG descriptor visualization are shown in Fig. 5 and Fig. 6 respectively with two samples (a) and (b). Total 504 features are extracted from number plate image. A linear support vector machine is used to classify number plate with parameters C is 2.5, termination criteria is 10^5 , and epsilon is $9.99E^{-7}$. SVM is trained with 10-cross fold validation to obtain optimal parameter for number plate classification.



Fig. 5 Number plate

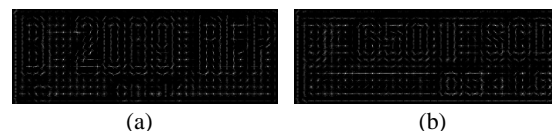


Fig. 6 HOG Descriptor visualization of number plate

To overcome the various problems of number plate localization, the proposed method offers some solutions to minimize the problems. Multi-scale windows are applied from the original detection window to deal with various size of number plate. Although the method is implemented in GPU, there is still limit on how many detection windows can be applied to achieve real time detection with the given image size. When using our laptop with the given specification and the described test data the limit is 1.05. For illumination problem, HOG has its own mechanism to deal with illumination as mentioned in [7]. SVM is used to deal with tilted number plate and part of the number plate that is covered by another objects by including some samples of tilted number plate and partially covered number plate as training data. To increase the accuracy of number plate localization, number plate region must pass a hit threshold. The hit threshold means that an object must be detected more than twice by the multi-window detector to be considered as license plate.

Localization result is marked by red rectangles as shown in Fig. 7 with four samples (a), (b), (c) and (d). The method can detect number plates with various size, tilted number plates, and partially covered number plates as shown in Fig. 7. The blue marker indicates region predicted by HOG+SVM that does not pass the VED. Many false detections are removed by VED method (a, b, c, d) but in sample (d), the VED method removed actual number plate because of the lighting on the plate. The failure detection is mainly caused by non-standard number plate (b) and more than half of the plate covered by another object (d). Custom number plate is very common in many countries and it is quite hard to include its model in the training data because of the large variety of possibilities. The false

positive is mainly caused by background texture that is similar to number plate such as background that contains letters (a).

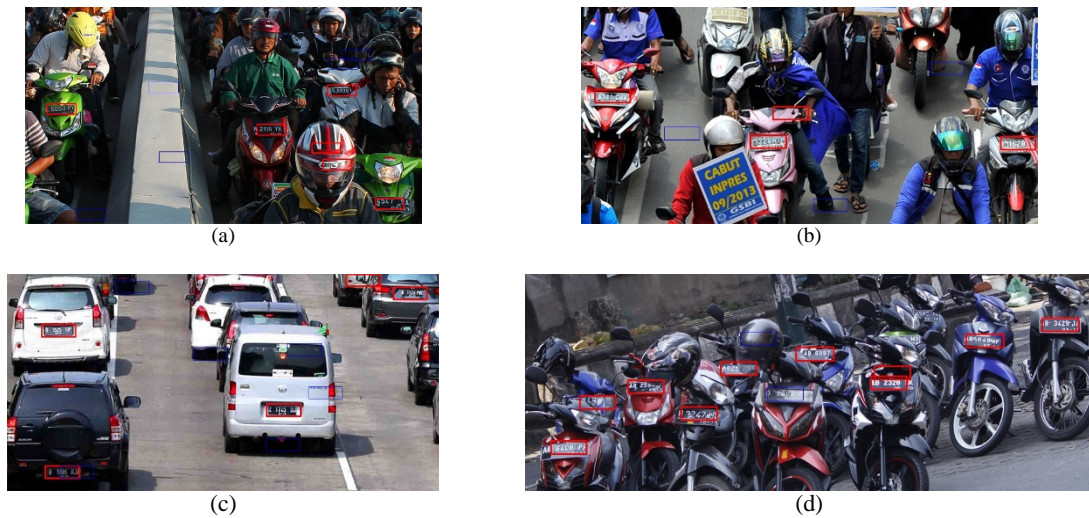


Fig. 7 Detected number plate in test image (red rectangle)

The test data is 15 images of traffic or parking area. A set of positive test data and negative test data is extracted from 15 test images and classified using HOG+SVM to obtain the score and label used for evaluation. Because of the imbalance between positive and negative data test, AUC (Area Under Curve) and BAC (Balance Accuracy) are used for evaluation [11]. The ROC (Receiver Operating Characteristic) curve of proposed method is shown in Fig. 8. Performance of the method is shown in Table I and Table II where TP is true positive, FN is false negative, TN is true negative, and FP is false positive. Sensitivity is $TP / (TP + FN)$, specificity is $TN / (TN + FP)$, BAC is $(sensitivity + specificity) / 2$, and AUC is obtained from ROC. True Positive Rate (sensitivity) and False Positive Rate ($1 - specificity$) are used to draw the ROC curve.

ROC is calculated from classification scores, actual labels, and predicted labels of all test data. By changing the threshold with selected classification score, the predicted labels and actual labels are compared to calculate the TPR and FPR values. This step is repeated for every classification score of all test data. The AUC from ROC curve in Fig. 5 is 0.9883 and BAC is 0.9362. The result of evaluation is good because all score are close to 1 which means that the method predicted right number plate with a high chance.

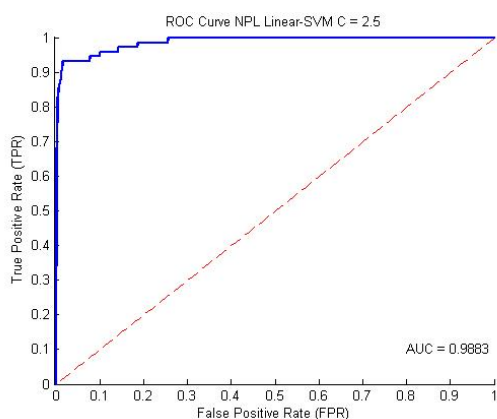


Fig. 8 ROC curve of proposed number plate localization method

Table I. Test data and evaluation result

Test Data	Positive Data	Negative Data	TP	FN	TN	FP
15	76	2639	67	9	2615	24

Table II. Performance evaluation of proposed NPL method

Sensitivity	Specificity	BAC	AUC
0.8815	0.9909	0.9362	0.9883

C. Processing Time

GPU is used to accelerate the proposed NPL method in order to achieve real time detection. The processing time is tested 10 times to ensure the system is stable when processing the data. Processing time is plotted according to each test image resolution with blue line for CPU and red line for GPU. The average processing time of vertical edge based localization using Sobel and box filters is shown in Fig. 9(a). Average processing time of HOG+SVM with multi detection window is shown in Fig.

9(b). The average total processing time of the proposed method is shown in Fig. 9(c). As shown in Fig. 9(c), GPU implementation of the proposed method is much faster (approximately 4x) than CPU implementation. The GPU performance is also more stable with increasing image resolution. On the largest test image with resolution 904 x 496, the total processing time of proposed NPL method is under 50ms or above 20 fps which can be considered as real time while total processing time of CPU is only 6 fps.

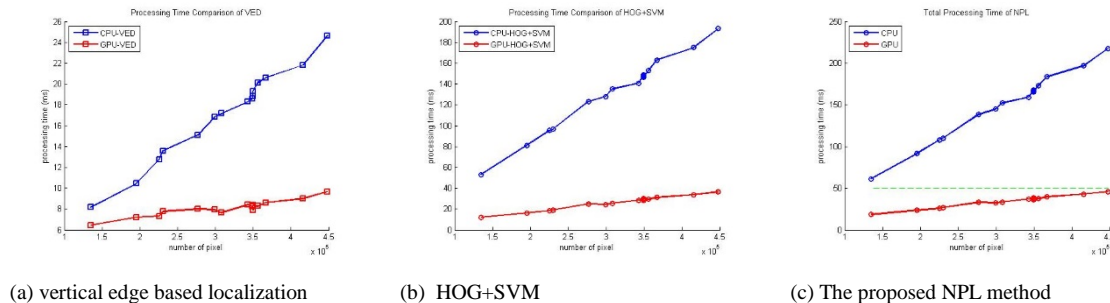


Fig. 9. Processing time comparison

V. Conclusions

The GPU accelerated number plate localization in crowded situation is proposed to localize number plate in a crowded situation. In a crowded situation, number plate can vary in size, illumination, orientation, and is often covered by other objects. The proposed method is designed to overcome these problems by using combination of vertical edge density and HOG (Histogram of Oriented Gradients) and SVM (Support Vector Machine) to localize the number plate. Vertical edge density alone cannot be used as primary detector because many edges would be found in a crowded situation. Therefore, combination of HOG+SVM is used to accurately localize the number plate from candidate number plate region extracted using vertical edge density method. The proposed method is evaluated using AUC (Area Under Curve) from ROC (Receiver Operating Characteristic) curve and BAC (Balance Accuracy). The evaluation shows good result with high AUC and BAC values. During future work, the proposed method can be further developed to localize custom number plate, highly covered number plate, and become part of the ANPR (Automatic Number Plate Recognition) system. The GPU performance also can be increased using more efficient algorithm for the filters and classifier.

References

- [1] M. Zahedi and S. M. Salehi, "License Plate Recognition System Based on SIFT Features," in *World Conference on Information Technology*, Istanbul, 2011.
- [2] V. Mai, D. Miao, R. Wang and H. Zhang, "An improved method for Vietnam License Plate location," in *2011 International Conference on Multimedia Technology (ICMT)*, Hangzhou, 2011.
- [3] P. Tarabek, "Fast license plate detection based on edge density and integral edge image," in *2012 IEEE 10th International Symposium on Applied Machine Intelligence and Informatics (SAMII)*, Herl'any, 2012.
- [4] A. M. Al-Ghaili, S. Mashohor, A. R. Ramli and A. Ismail, "Vertical-Edge-Based Car-License-Plate Detection Method," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 1, pp. 26 - 38, 03 October 2012.
- [5] M. M. Dehshibi and R. Allahverdi, "Persian vehicle license plate recognition using multiclass Adaboost," *International Journal of Computer and Electrical Engineering*, vol. 4, no. 3, pp. 355 - 358, June 2012.
- [6] B. B. Singh and V. H. Deepthi, "Survey on Automatic Vehicle Number Plate Localization," *International Journal of Computer Applications*, vol. 67, no. 23, pp. 7 - 12, 2013.

- [7] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, San Diego, CA, USA, 2005.
- [8] D. F. Llorca, R. Arroyo and M. A. Sotelo, "Vehicle logo recognition in traffic images using HOG features and SVM," in *2013 16th International IEEE Conference on Intelligent Transportation Systems - (ITSC)*, The Hague, 2013.
- [9] J. Fung and S. Mann, "Using graphics devices in reverse: GPU-based Image Processing and Computer Vision," in *2008 IEEE International Conference on Multimedia and Expo*, Hannover, 2008.
- [10] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2, no. 3, p. No. 27, 2011.
- [11] C. G. Weng and J. Poon, "A new evaluation measure for imbalanced datasets," in *Proceedings of the 7th Australasian Data Mining Conference (AusDM '08)*, Darlinghurst, Australia, 2008.