International Journal of Engineering and Technical Research (IJETR)
ISSN: 2321-0869 (O) 2454-4698 (P) Volume-7, Issue-10, October 2017

A Support Vector Machine (SVM) and Speeded Up Robust Features (SURF) for Indonesian Car License Plate Identification System

Andika Vebrina, Fitri Arnia, Yuwaldi Away

***Abstract*— The aim of this research is to design a car plate identification system by using the Speeded Up Robust Features (SURF) and the Support Vector Machine (SVM) methods. The system composes of two main stages including license plate localization and characters recognition. SURF can rapidly extract the interest points of an image because it uses the integral image technique to compute the determinant of the Hessian blob detector. SVM is a powerful machine learning technique to solve the classification problems which has been developed based on Structural Risk Minimization (SRM) principle. SVM has a good performance in character recognition. Therefore, it is very appropriate to be used in this system. By combining such two methods, this research has achieved 98.4% and 99.2% accuracy for plate detection and characters recognition, respectively. The plate verification and characters recognition have been performed in about 1.46 ms and 13.29 ms, respectively.**

***Index Terms*—Car plate identification, characters recognition, hessian matrix, integral image, SURF, support vector, SVM.**

I. INTRODUCTION

With the development of vision technology, research on car license plate identification system has also been done intensively. Nevertheless, this field of research still leaves a very interesting challenge to be scrutinized. This is because the development of car license plate identification system requires the integration of various problem-solving techniques [1] to obtain the reliable results. The most fundamental problems often encounter are the accuracy and the processing speed. In addition, the non-uniform of car license plates and also the quality of lighting at the time of image capturing also become the challenges in developing this identification system.

One of the solutions to meet the need of the above problems is, the authors has proposed a car license plate identification system by integrating SURF (Speeded-Up Robust Features) and SVM (Support Vector Machine) algorithms. SURF has been used because it can quickly extract the important points of an image by using the integral image technique on the Hessian matrix operation [2][3]. SVM is a powerful computer learning system for solving classification problems that has been developed based on Structural Risk Minimization (SRM) principle of statistical learning theory [4][5].

Andika Vebrina, Electrical Engineering, Syiah Kuala University, Banda Aceh, Indonesia
Fitri Arnia, Electrical Engineering, Syiah Kuala University, Banda Aceh, Indonesia
Yuwaldi Away, Electrical Engineering, Syiah Kuala University, Banda Aceh, Indonesia

Similar researches have been conducted as in [6] using Dual-Tree Complex Wavelet Transform (DTCWT) and Artificial Neural Network (ANN). This method has achieved 94% accuracy for plate characters recognition. References [7] used the Hamming distance approach in characters recognition method that has resulted 95% accuracy. References [8] used the Global Direction Contributivity Density (G-DCD), Local Direction Contributivity Density (L-DCD), and Peripheral Direction Contributivity (PDC). This study has yielded 93.54% total accuracy. References [9] used Neural Network (NN) that has resulted a good accuracy at 97.78%, but this study was carried out for license plates of Czechoslovakia where they have a standard European countries license plate with a standard characters as well, which can be more easily detected. References [10] used the SVM by-means method that yielded 96.7% accuracy with 57.4ms of time identification. References [11] used Probabilistic Neural Networks (PNNs), which achieved 89.1% accuracy. References [12] used the template matching method, and it has reached 91.1% accuracy of the characters recognition. Reference [3] used SURF and Bag-of-Word, and Histogram Similarity has resulted 90.69%, 90.32% and 98% accuracy for images, videos, and webcam, respectively. Reference [13] used adaptive template matching which is implemented by using correlation method. This method has resulted 96% accuracy for the identification of the vehicle license plate in India.

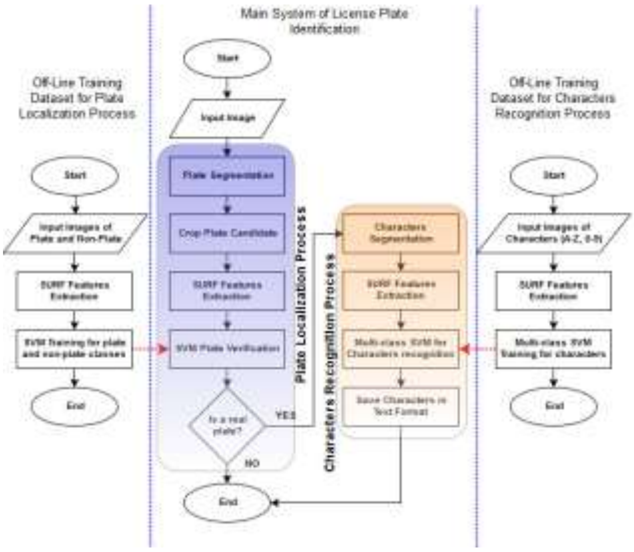


Figure 1. Flowchart of Indonesian car license plate identification system.

In this paper, we present a new robust Indonesian license plate recognition system as shown in Fig.1. The system composes of two main stages, namely *plate localization* and *characters recognition*. The plate localization stages is a stage to localize the region of license plate area, and the characters recognition stage is a stage to recognize the characters on the detected plate area. The processes of training data for both stages have been done in off-line mode, meaning that the data have been trained separately from the main system of car plate identification. The testing performance has been performed by using the car images were obtained with different backgrounds, illumination, license angles, distance from camera to cars, light conditions and different size and type of license plates.

The rest of the paper is organized as follows: section 2 presents the literatures study, section 3 explains the SVM training model, section 4 proposes the plate localization, section 5 proposes the characters segmentation and recognition, section 6 shows the result and discussion, and section 7 concludes the paper.

II. Literatures Study

A. Support Vector Machines (SVM)

Basically, the objective of SVM is to create a hyperplane as a separator between two classes (i.e. +1 class and -1 class) with the largest margin [14] [15]. SVM was proposed for binary classification of pattern recognition techniques.

Given a training set $T = \{x_i, y_i\}_{i=1}^{l}$ and $(x_i \in R^n, y_i \in \{-1,1\})$, where l is the number of data, n is the dimension of problem, and if the real function is determined by g(x), then the classification function can be obtained by [5]:

$$f(x) = sign(g(x)) \tag{1}$$

It can divide the points in the R^n space into two parts. If the function of g(x) is linear, then the hyperplane equation is [5]:

$$g(x) = w.x + b \tag{3}$$

where ***w*** is the weight of the vector and ***b*** is the bias or threshold. Fig. 2 shows the linear hyperplane that separates the two classes. The parts that become the support vectors are marked with the larger circles.

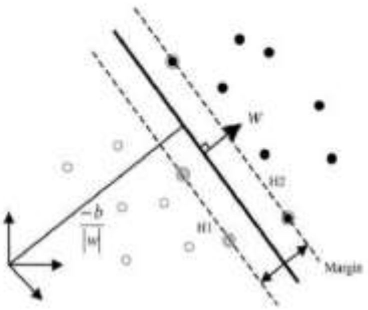


Figure 2. Linear hyperplane [14]

SVM is not only expected to split the two classes perfectly, but it is also expected to classify the maximum intervals called maximum margin. Such separation is called optimal hyperplane division, so data can be classified into two optimization types as given below: [14] [16].

$$\min_{w,b} \frac{1}{2}\|w\|^2 \tag{4}$$

s.t. $\quad y_i(w.x_i + b_i) \geq 1; i = 1,...l$

By introducing the Lagrange multiplier α_i, i = 1,…..,l, and according to the principle of duality, the problem of (4) can be converted into dual problem [5]:

$$\min_{\alpha} \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l} y_i y_j \alpha_i \alpha_j (x_i \cdot x_j) - \sum_{j=1}^{l} \alpha$$
$$s.t. \begin{cases} \sum_{i=1}^{l} y_i \alpha_i = 0 \\ \alpha_i \geq 0, i = 1, \cdots, l \end{cases} \tag{5}$$

Given the solution of $\alpha_1 \ldots \alpha_l$ to the dual problem, solution to the original problem for *w* is:

$$w = \sum_{i=1}^{l} \alpha_i y_i x_i \tag{6}$$

and ***b*** can be determined by:

$$b = y_j - \sum_{i=1}^{l} y_i \alpha_i (x_i^T . x_j) \tag{7}$$

Non Linear SVM

In practice, the data are not linearly separable, the hyperplane that maximizes margin can be done by minimizing misclassification error by introducing a slack variable ξ_i [14].

$$y_i(w.x_i + b_i) \geq 1 - \xi_i \tag{8}$$

If an error occurs, the corresponding ξ_i must exceeds unity, so $\sum_i \xi_i$ is an upper bound on the number of training errors. So that the objective function (4) can be rewritten as:

$$\min_{w,b} \left\{ \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l} \xi_i \right\} \tag{9}$$

where *C* is the parameter chosen by user to control the misclassification error.

Multi Class SVM

SVM has also been developed to solve multi-class cases. The most widely used method for multi-class SVM is to create N-SVM, where each SVM classifies one class from the other classes [16]. This method is called the one-versus-rest method [16]. In addition, the commonly used method is one-versus-one or called pairwise SVM [14], where it combines all possibilities of a two-class classification. For the N-class problem, the classifier training process should be done as much as $\frac{N(N-1)}{2}$ [16].

SVM Kernel

In this work, we used two types of SVM kernel namely Linear kernel and Radial Basis Function (RBF) kernel. Linear Kernel is used in license plate localization process whereas RBF kernel is used in characters recognition process. The RBF kernels can be expressed in feature vectors in the input space defined as [17]:

$$K(\boldsymbol{x}, \boldsymbol{x}') = \exp\left(-\frac{\|x - x'\|^2}{2\sigma^2}\right) \tag{10}$$

Where: $\|x - x'\|^2$ is the Euclidean square distance between two feature vectors and σ is the free parameter. By introducing $\gamma = -\frac{1}{2\sigma^2}$, RBF kernel is defined as [17]:

$$K(x, x') = \exp(\gamma\|x - x'\|^2) \quad (11)$$

LibSVM

LibSVM is an open source library developed by Chih-Chung Chang and Chih-Jen Lin since the year 2000. It can be used to classify two-class and multi-class cases. Based on its literature [19], libSVM uses the one-versus-one SVM.

B. *Speeded Up Robust Features (SURF)*

SURF is a method used to extract local features from an image. It consists of a detector and a descriptor. The detector is used to locate the interest points (keypoints) and descriptor is used to extract the features in every detected keypoint. SURF uses the Hessian matrix approach to detect interest points by finding the maximum determinant value [2]. The Hessian matrix is used because it has good performance in terms of accuracy. The operation of Hessian matrix involves the integral image that allows for fast computation of box type convolution filters, so that the computational time decrease significantly [15]. The entry of an integral image I_Σ (x) is generated by a square shape in image I at the location **x** = $(x,y)^T$. This square area is formed by the origin and **x** as follows [2]:

$$I_\Sigma(\mathbf{x}) = \sum_{i=0}^{i \leq x} \sum_{j=0}^{j \leq y} I(i, j) \quad (12)$$

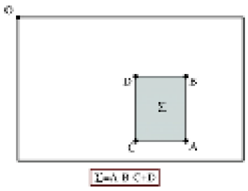


Figure 3.Integral Image [3]

If given a point **x** = (x, y) in an image I, the Hessian H (**x**, σ) matrix at point **x** with the scale σ is defined as follows [2]:

$$H(x,\sigma) = \begin{bmatrix} Lxx(x,\sigma) & Lxy(x,\sigma) \\ Lxy(x,\sigma) & Lyy(x,\sigma) \end{bmatrix} \quad (13)$$

where Lxx (**x**,σ) is the convolution of the Gaussian second order derivative $\frac{\partial^2}{\partial x^2} g(\sigma)$ with the image I in point **x**, and similarly for L*xy* (**x**,σ) and L*yy* (**x**,σ).

In order to use the integral image in computation of convolution of the Gaussian second order derivative, SURF uses an approximation of boxes filter as shown in Fig. 4.

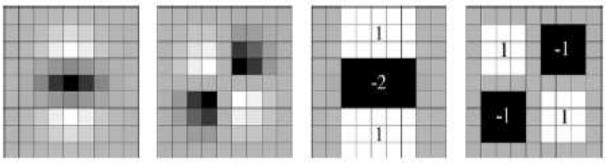


Figure 4. Left to right: Gaussian filter in *y*-direction (L*yy*) and *xy*-direction (L*xy*) following by their approximation in y-direction (Dyy) and xy-direction (Dxy) [2].

The approximation of convolutions of the Gaussian filter in *x*-direction, *y*-direction, and *xy*-direction are denoted to D*xx*, D*yy*, and D*xy*, respectively. Then, determinant of approximation of the Hessian matrix is written as follow [2]:

$$\det(\mathcal{H}_{approx}) = D_{xx}D_{yy} - (wD_{xy})^2 \quad (14)$$

where *w* is the relative weight of the filter responses used to balance the effect of using box filter approximation. More details about this explanation is can be found in [2].

SURF Descriptor

To determine the features, SURF uses Haar Wavelet responses in vertical direction *dx* and horizontal direction *dy* as shown in Fig. 5. The first step is to form a square area centered on the interest point. The size of this square is 20*s*, where *s* is the scale at which the interest point is detected, then it is divided into 4x4 sub-region squares [2] as described in Fig. 6.

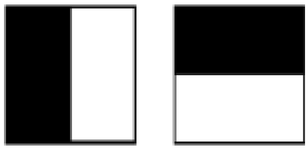


Figure 5. Haar wavelet filter used to get orientation changes in pixel intensity around the interest point. The black weight is -1 and white weight is +1.

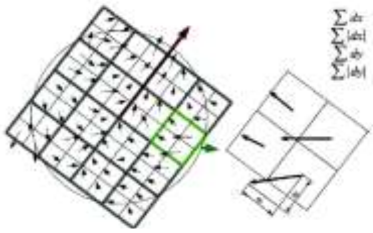


Figure 6. Square area with 4x4 sub-region [2]

For each of sub-region, convolution of pixels is performed with Haar wavelet response in direction of *dx* and *dy*. Then the orientation changes the intensity of the pixels. The obtained horizontal and vertical Haar Wavelet responses form a vector v = (Σ dx, Σ dy, Σ |dx|, Σ |dy|), which then produces a feature vector of 64 dimensions [2].

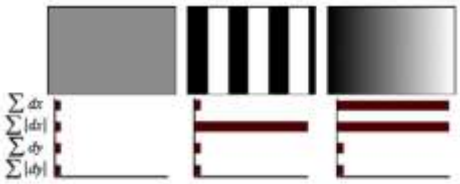


Figure 7. Illustration of SURF descriptor values for a sub-region [2]

III. Training SVM Model

Before plate localization and characters recognition can be performed, it is firstly required to conduct the training (learning) SVM data for both stages. In this work, the training data has been done in off-line mode, meaning that the data has been trained separately from the main system of car license plate identification. The objective of training SVM data is to classify the data based on given dataset. Then, the results of optimized training parameters, weight ***w*** and bias ***b*** can be used to classify the new unknown data.

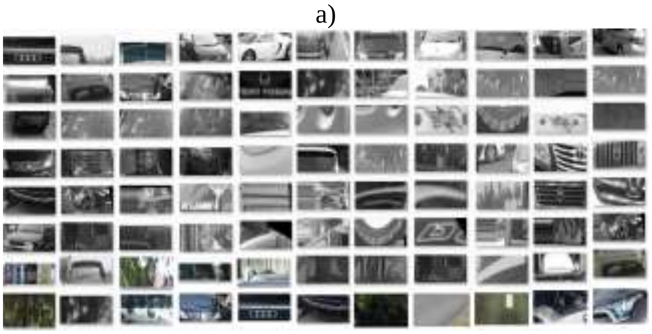
Dataset Training

The dataset for the process of license plate localization consist of two parts, including positive and negative dataset.

The positive dataset compose of license plate images, whereas the negative dataset compose of images which have totally no relation to the license plates, such as left/right car side, mirrors, lights indicators, tires, car doors, and so on. We have encountered difficulty to find the benchmarking dataset for Indonesian license plates, so we created our own dataset for both classes from the images captured by a camera. Fig. 8.a and 8.b show the used positive and negative classes dataset in license plate localization process, respectively.

For the characters recognition training dataset, we have used dataset from the *tesseract* [18] plus some of independent characters to fulfill the non-covered Indonesian style license plate characters. These independent characters are extracted from some images of Indonesian car license plates. The total of 36 classes have been labeled 0 - 9 and 10 - 35 corresponding to 10 numbers from 0 - 9 and 26 alphanumeric characters from A - Z, respectively.

a)



b)

Figure 8 Dataset training a) of class plate, and b) of class non-plate

Feature Extraction

The features extraction process has been performed by using the SURF method. For every image, SURF can result N strongest keypoints, and each keypoint consists of 64 features. Then, the total obtained features X_i from an image is a 2-D matrix (size N x 64) as described below:

$$X_i = \begin{bmatrix} f_{1,1} & f_{1,2} & f_{1,3} & \cdots\cdots & f_{1,63} & f_{1,64} \\ f_{2,1} & f_{2,2} & f_{2,3} & \cdots\cdots & f_{2,63} & f_{2,64} \\ f_{3,1} & f_{3,2} & f_{3,3} & \cdots\cdots & f_{3,63} & f_{3,64} \\ & & & . & & \\ & & & . & & \\ f_{N,1} & f_{N,2} & f_{N3} & \cdots\cdots & f_{N,63} & f_{N,64} \end{bmatrix} \tag{15}$$

where i is the ith image number. In order to be used with **LibSVM**, X_i must be transformed into 1-D features vector by reshaping it, so that their elements become a features vector as follow:

$$X_i = [f_{1,1} \; f_{1,2} \cdots \; f_{1,64} \; f_{2,1} \; f_{2,2} \cdots \; f_{2,64} \cdots\cdots\cdots f_{N,1} \; f_{N,2} \cdots \; f_{N,64}] \tag{16}$$

So, the features matrix **F** from all training images $\{X_i\}_{i=1}^{l}$, where ***l*** is the total of training data, can be written as follow:

$$\mathbf{F} = \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ . \\ . \\ X_l \end{bmatrix} = \begin{bmatrix} f_{1,1}|_{X_1} & f_{1,2}|_{X_1} \cdots & f_{1,64}|_{X_1} & f_{2,1}|_{X_1} & f_{2,2}|_{X_1} \cdots & f_{2,64}|_{X_1} & \cdots\cdots f_{N,1}|_{X_1} & f_{N,2}|_{X_1} \cdots & f_{N,64}|_{X_1} \\ f_{1,1}|_{X_2} & f_{1,2}|_{X_2} \cdots & f_{1,64}|_{X_2} & f_{2,1}|_{X_2} & f_{2,2}|_{X_2} \cdots & f_{2,64}|_{X_2} & \cdots\cdots f_{N,1}|_{X_2} & f_{N,2}|_{X_2} \cdots & f_{N,64}|_{X_2} \\ f_{1,1}|_{X_3} & f_{1,2}|_{X_3} \cdots & f_{1,64}|_{X_3} & f_{2,1}|_{X_3} & f_{2,2}|_{X_3} \cdots & f_{2,64}|_{X_3} & \cdots\cdots f_{N,1}|_{X_3} & f_{N,2}|_{X_3} \cdots & f_{N,64}|_{X_3} \\ & & & & . & & & & \\ & & & & . & & & & \\ & & & & . & & & & \\ f_{1,1}|_{X_l} & f_{1,2}|_{X_l} \cdots & f_{1,64}|_{X_l} & f_{2,1}|_{X_l} & f_{2,2}|_{X_l} \cdots & f_{2,64}|_{X_l} & \cdots\cdots f_{N,1}|_{X_l} & f_{N,2}|_{X_l} \cdots & f_{N,64}|_{X_l} \end{bmatrix} \tag{17}$$

Then, **F** can be directly trained by **LibSVM** along with their corresponding labels **L** as shown below:

$$\mathbf{L} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ . \\ . \\ y_l \end{bmatrix} \tag{18}$$

In the case of binary classification of the license plate detection, y_1 through y_i could be the number 0 or 1, where 0 represents the non-plate and 1 represent the plate. Whereas, in case of multi-class classification of characters recognition, we labeled y_1 through y_i with the numbers between 0 and 35, where 0 - 9 correspond to the number between 0 - 9, and 10 - 35 correspond to 26 alphanumeric characters from A - Z.

Training data for plate detection is done by using the linear SVM, whereas training data for characters recognition is done by using the RBF kernel. The used of RBF kernel in characters recognition is based on the results of the cross-validation as shown in table 1, where it appears that the RBF kernel provides maximum accuracy compared to 3rd order Polynomial kernel and Sigmoid kernel.

Table 1. Cross-validation performance for tree types of SVM Kernels used in characters recognition process.

SVM Kernels	C	Gamma (γ)	Cross-Validation Accuracy
3rd order Polynomial	0.5	2	96.91 %
Radial Based Function (RBF)	8	2	99.47 %
Sigmoid	8	1	97.43 %

In that way, the RBF kernel is better suited to our characters dataset, so we chose to use the RBF kernel in the case of characters recognition process.

IV. PLATE LOCALIZATION

The plate localization is a very important stage in the system of car license plates identification. This is due to the enormous variations between one image to another, such as backgrounds, illumination, license angles, distance from camera to cars, light conditions and different size and type of license plates. In addition, the quality of images also crucial important in determining the successful of the plate localization.

A. Pre-processing

The initial step of the plate localization process is a pre-processing step, where it is the step to obtain the candidate plate areas from an image. The pre-processing steps consist of:

1. Conversion original image into grayscale image.

It is required to accelerate the detection process. In addition, the SURF features extractor works only with the grayscale images. Fig. 9.a shows the result of grayscale image.

2. Tophat Morphology.

It is used to find the difference between the grayscale image and the image after dilation and erosion processes. The purpose of this process is to obtain an image that has a higher brightness on the plate area. Fig. 9.b shows the result of the tophat morphology.

3. Binarization

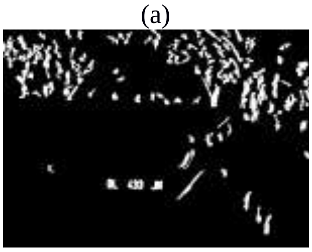
This process is to change the image pixels to black and white. The higher contrast of image pixels will be changed to 1 (white) and the darker part will be changed to 0 (black). So, that the candidate plate areas will be marked as white areas. Binarization process is done by using automatic thresholding OTSU method, which is a very effective method to be used because its capability to adapt to different lighting levels. Fig. 9.c shows the image binarization result.

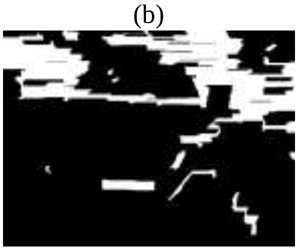
4. Closing Morphology

Closing morphology is applied with the aim is that to connect the existing characters in the plate area, so that the characters will be interconnected each other and form a detectable plate region.


(a)


(b)


(c)


(d)

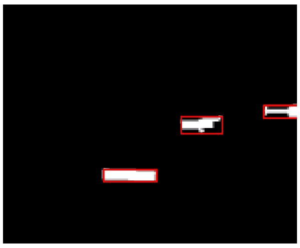

(e)

Figure 9 Preprocessing results of a) grayscale, b) tophat morphology, c) binarization, d) closing morphology, and e) contours detection and size verification.

5. Contours detection and Bounding box

This step is to find the contours of the existing objects of the previous closing morphology image result. The objective of this step is to obtain the plate candidate areas. From any detected contours, it is applied a bounding box and verified its size, whether it meets the criteria of the license plate size or not.

The criterions of the license plate has a minimum size of *minwidth* = 25 pixels, *minheight* = *aspect* x *minwidth*, and maximum size: *maxwidth* = 125 pixels, *maxheight* = *aspect* x *maxwidth*, where *aspect* is the criteria of the ratio between height and width and its value is given by 2.5. The *error* is 5% of the aspect. So, the criteria of minimum size area is *min = 25 x aspect x 25* pixels and the maximum size area is *max= 125 x aspect x 125* pixels. The minimum and maximum respect ratio between height and width are *rmin = aspect – aspect x error* and *rmax= aspect+aspect x error*, respectively. If *area* and *r* is the size area and the ratio of height and width of the evaluation plate candidate, respectively, then the pseudo code of criterion size verification is as follow:

BEGIN
IF((area<min **OR** area>max) **OR** (r<rmin **OR** r > rmax))
    return FALSE
ELSE
    return TRUE
END

If the contour does not meet the criteria, then it will be removed and only contours that meet the criteria will be retained as shown in Fig. 9.e.

6. Crop candidate plate

As a final step of the pre-processing stage is to crop the bounding box areas as the obtained plate candidates. Fig. 10 shows the results of three obtained plate candidates.

Figure 10. In the left side image shows that the candidate plates have been obtained, and in the right side images show the cropped plate candidates.

B. Plate Verification

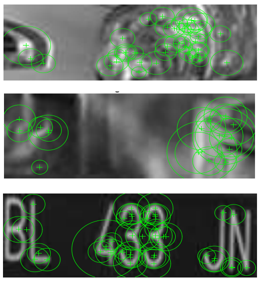
Once the plate candidate has been obtained, then it should be classified to ensure whether it is true as a real license plate or not. This process involves SURF and SVM. The plate candidate is firstly extracted its features by using SURF method. Fig.11.b shows the results of keypoints from each of the detected plate candidate in Fig. 11.a. These keypoints are then extracted their features by SURF descriptor and then classified by using bi-class linear SVM.

V. Characters Segmentation And Recognition

The characters recognition process is performed after the plate localization is completely accomplished and the plate area has been already detected. Similar to the plate localization process, the characters recognition is also required the pre-processing stage to obtain the segmented characters. The pre-processing steps in characters recognition stage are as follow:

1. Convert the detected plate area into gray-scale format.
2. Apply the tophat morphology
3. Image binarization by using automatic thresholding.

a) Detected plate candidates

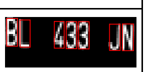
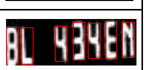
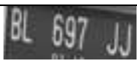


b) SURF keypoints

Figure 11. a) Detected plate candidates, and b) their corresponding detected keypoints.

The process of characters segmentation consists of contour detection and bounding rectangle. Contour detection is applied to obtain the contour of the characters, then for each detected contour is applied a bounding rectangle. Each of bounding rectangle is verified its size. It must satisfy the aspect of comparison between height and width, where the minimum aspect is = 1.2 and the maximum aspect is 5.0. It also must meet the minimum and maximum height. Where the minimum height is 55 pixels and the maximum height is 95 pixels. If these criterions are satisfied then the characters will be segmented and cropped to the further recognition process.

Table 2 Result of characters segmentation for verified plates

Verified Plates	Grayscale	Binary images	Segmented characters
			
			
			
			
			
			

Once the characters have been obtained, the SURF features extraction is applied on the same way of in the plate verification process. Then these characters are classified by using multi-class SVM with the RBF kernel. Fig. 12 shows the example of detected keypoints of characters B and 4.

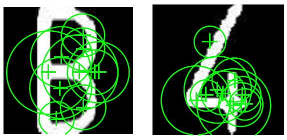


Figure 12. Detected keypoints of characters B and 4

VI. Results And Discussion

A. System Performance

To evaluate the performance of the proposed system, The total of 250 images have been tested, resulted 100% accuracy of plate candidates segmentation. Then from these segmented plate candidates, they have been successfully verified by using SURF and SVM with the accuracy is 98.4% (246 out of a total 250 images), while four of them have failed to be recognized. Then from these 246 localized plates, they have been successfully recognized their characters with the accuracy is 99.2% (244 out of the total 246 localized plates). Some of the perfect car license plates identification are shown in Fig. 13, and details of result performance are shown in table 3.

The percentage of accuracy is calculated as follow:

$$Accuracy\ (\%) = \frac{\text{Number of Succeses}}{\text{Total Number of Tested Images}}\ x\ 100\%$$

Table 3. Performance of car license plate identification

Process Stages	Number of Tested Images	Number of Successes	Number of Failures	Accuracy of Successes
Plate Candidate Segmentation (Preprocessing)	250	250	0	100.0%
Plate Identification (SURF & SVM)	250	246	4	98.4%
Character Identification (SURF & SVM)	246	244	2	99.2%

Table 4 Execution times of each process

Process	Consumed Time Rate (miliseconds)
Plate segmentation	195
Plate verification	1.46
Characters Segmentation	52
Cahracters Recognition	13.29
Total Time	**261.75**

The obtained results has indicated that the performance of Indonesian car identification plate system has a good accuracy that achieves 98.4% for plate detection and 99.2% for characters recognition as shown in table 3. The total execution time is 261.75 ms as shown in Table 4. The longest execution time occurs in the plate segmentation process that required 195 ms, while for plate verification involved SURF and SVM it just took only 1.46 ms. The characters segmentation process took about 52 ms, and for characters recognition process required 13.29 ms. Looking at the performance obtained, it can be concluded that as the name suggested the Speed-up Robust Features, SURF is a fast and robust features extraction, and SVM has good performance in identifying both the plates and the characters with a very good accuracy.

Figure 13. Results of perfect identification of the car license plates

A. *Problem Analysis*

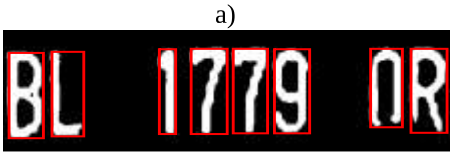
We have identified the problem of misclassifications in the designed system as shown in Fig. 14. a and b. In Fig. 14.a, it is clear that the segmentation result of character ‘Q’ resembles character ‘0’. The character ‘Q’ has truncated that causes SVM recognized it as a character ‘0’. Similarly in Fig. 15 where the character ‘B’ looks slightly defective. The segmented result of this character ‘B’ has a disconnected area as shown clearly in Fig. 15.b, so that SVM identified it as a number ‘8’.

VII. Conclusion

This study has developed an Indonesian car license plate identification system by integrating SURF (speeded-up robust features) and support vector machine (SVM) methods. By combining these two methods, this research has achieved 98.4% and 99.2% accuracy for plate detection and characters recognition, respectively. The plate verification and characters recognition have been performed in about 1.46 ms and 13.29 ms, respectively. The results are very encouraging.

For the perspective of this research, the authors need to improve the execution time of plate segmentation and also investigate a more robust pre-processing method, so that it can enhance the recognition performance, and this system can be used in a real time application system.

a)



b)

Figure 14 a) Misclassification of Character ‘Q’, b) the result of characters segmentation. It is clearly that the result of character segmentation is not perfect. The character ‘Q’ has truncated at the bottom.

a)

b)

Figure 15. a) Misclassification of Q Character b) The result of characters segmentation. It is clearly can be seen that the result of character segmentation is not perfect. The Q character truncated at the bottom.

ACKNOWLEDGMENT

The present research work has been supported by the Ministry of Research, Technology and Higher Education of the Republic of Indonesia (RISTEKDIKTI), the Aceh Polytechnic, and the University of Syiah Kuala, Banda Aceh-Indonesia. The authors gratefully acknowledge the support of these institutions.

REFERENCES

[1] Kim, K.K., Kim, K.I., Kim, J.B., Kim, H. J., “Learning-Based Approach for License Plate Recognition”, Neural Networks for Signal Processing X, 2000. Proceedings of the 2000 IEEE Signal Processing Society Workshop, pp. 614 - 623 vol.2, 2000.

[2] Bay, H., Ess, A., Tuytelaars, T., Gool, L.V., “Speeded-Up Robust Features (SURF)”, Computer Vision and Image Understanding, pp. 346-359, 2008.

[3] Khaleel, F.M., Abdullah, S.S.N.H.,”License Plate Detection Based on Speeded Up Robust Features and Bag of Words Model”, Smart Instrumentation, Measurement and Applications (ICSIMA), 2013 IEEE International Conference on , IEEE, p.1 – 5, 2013.

[4] Nasien, D.,Haron, H.,Yuhaniz, S.S.,“Support Vector Machine (SVM) For English Handwritten Character Recognition”, Second International Conference on Computer Engineering and Applications, IEEE,ISSN: 978-0-7695-3982-9,pp. 249 – 252, 2010.

[5] Liyan, T,Xiaoguang, H., Peng, F.,“Application of SVM in Embedded Character Recognition System”, Industrial Electronics and Applications, 2009. ICIEA 2009. 4th IEEE Conference, IEEE, ISSN: 978-1-4244-2800-7, pp. 1260 – 1264, 2009.

[6] J. P. D. Dalida, A. J. N. Galiza, A. G. O. Godoy, M. Q. Nakaegawa, J. L. M. Vallester and A. R. d. Cruz, "Development of intelligent transportation system for Philippine license plate recognition," 2016 IEEE Region 10 Conference (TENCON), Singapore, 2016, pp. 3762-3766.

[7] Singh. H, Pandey. D, Verma. S, Dhiman. P, "Automatic License Plate Recognition System", JETIR, Vol.2, Issue.3, pp.690-694, March 2015.

[8] Y. Wen, Y. Lu, J. Yan, Z. Zhou, K. M. von Deneen and P. Shi, "An Algorithm for License Plate Recognition Applied to Intelligent Transportation System," in IEEE Transactions on Intelligent Transportation Systems, vol. 12, no. 3, pp. 830-845, Sept. 2011.

[9] Volna, E., Kotyrba, M., “Vision system for licence plate recognition based on neural networks”, International Conference on Hybrid Intelligent Systems (HIS), IEEE, ISSN: 978-1-4799-2439-4, pp.140-143, 2013.

[10] Ghahnavieh, A.E., Amirkhani-S.A.,Raie, A.A., “Enhancing the License Plates Character Recognition Methods by Means of SVM”, Electrical Engineering (ICEE), 2014 22nd Iranian Conference, IEEE, pp. 220 – 225, Mai 2014.

[11] C. Anagnostopoulos, I. Anagnostopoulos, V. Loumos, and E. Kayafas, “A license plate-recognition algorithm for intelligent transportation system applications,” IEEE Trans. Intell. Transp. Syst., vol. 7, no. 3, pp. 377–392, Sep. 2006.

[12] A.Tihar, A.Adnan, M.Fahad “License Plate Recognition for Pakistani License plates”, Canadian journal on image processing., vol. 1, no.2, April 2010.

[13] A. Choudhury and A. Negi, "A new zone based algorithm for detection of license plate from Indian vehicle," 2016 Fourth International Conference on Parallel, Distributed and Grid Computing (PDGC), Waknaghat, 2016, pp. 370-374.

[14] Byun, H., and Seong-Whan Lee, S-W.,“Applications of Support Vector Machines for Pattern Recognition: A Survey”,SVM 2002, LNCS 2388, pp. 213-236, 2002.

[15] Burges, C.J.C., “A Tutorial on Support Vector Machines for Pattern Recognition”, Data Mining and Knowledge Discovery 2, 121-167, 1998.

[16] Maruyama, K.-I.,Maruyama, M.,Miyao, H.,Nakano, Y., “Handprinted Hiragana Recognition Using Support Vector Machines”, Frontiers in Handwriting Recognition, 2002. Proceedings. Eighth International Workshop, IEEE, ISSN: 0-7695-1692-0, pp. 55 – 60, 2002.

[17] Vert, Jean-Philippe, Koji Tsuda, and Bernhard Schölkopf "A primer on kernel methods". Kernel Methods in Computational Biology, (2004).

[18] https://github.com/openalpr/train-ocr/tree/master/eu/input, , diunduh tanggal: 10 Februari 2017.

[19] Chih-Chung Chang and Chih-Jen Lin, LIBSVM : a library for support vector machines. ACM Transactions on Intelligent Systems and Technology, 2:27:1--27:27, 2011. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

Andika Vebrina, ST, MT, was born in Banda Aceh, February 10, 1986. She received her B. Eng degree in Informatics Engineering from Universitas of Muhammadiah (UNMUHA), Banda Aceh, Indonesia in 2009. She has been working at Aceh Polytechnic, Banda Aceh, Indonesia since 2009 as a lecturer in Informatics Engineering Departement. She obtained her master degree in Electrical Enginering Departement at Syiah Kuala University, Banda Aceh, Indonesia in 2017. Now, she is focusing on research topics of mobile applications, image processing and machine learning.

Dr. Fitri Arnia, S.T., M.Eng.Sc received B. Eng degree from Universitas Sumatera Utara (USU), Medan in 1997. She finished her master and a doctoral degree from University of New South Wales (UNSW), Sydney, Australia and Tokyo Metropolitan University, Japan in 2004 and 2008 respectively. She has been with the Department of Electrical Engineering, Faculty of Engineering, Syiah Kuala University since 1999. She is a member of IEEE, APSIPA, and IAENG. Her research interests include signal, image and multimedia information processing. She serves as the Editor-in-Chief for Jurnal Rekayasa Elektrika (Accredited by RISTEKDIKTI).

Prof. Dr. Ir. Yuwaldi Away, M.Sc was born in 1964 in South Aceh, Indonesia. He obtained his degree in Electrical-Computer Engineering (1988) from Sepuluh Nopember Institute of Technology (ITS) Indonesia. He start his carrier as lecturer in 1990 at Syiah Kuala University, Indonesia (1990-now). He earned his M.Sc. (1993) from Bandung Institute of Technology (ITB) Indonesia, and he finished his Ph.D. (2000) in an Industrial Computer from the National University of Malaysia. From 1996 to 2004, he was a lecturer as well as a research assistant at the National University of Malaysia. Now he is a professor (2007-now) as well as Head of Center for Automation and Robotics Studies (PUSMATIK) at Syiah Kuala University. The research scope is a combination of theory and practical, including microprocessor-based systems, simulation, automation, and optimization.