

Traffic Sign Detection and Recognition using Features Combination and Random Forests

Ayoub ELLAHYANI

Mohamed EL ANSARI

Ilyas EL JAAFARI

LabSIV, Department of Computer Science
Faculty of Science, Ibn Zohr University
BP 8106, 80000 Agadir, Morocco.

LabSIV, Department of Computer Science
Faculty of Science, Ibn Zohr University
BP 8106, 80000 Agadir, Morocco.

LabSIV, Department of Computer Science
Faculty of Science, Ibn Zohr University
BP 8106, 80000 Agadir, Morocco.

Said CHARFI

LabSIV, Department of Computer Science
Faculty of Science, Ibn Zohr University
BP 8106, 80000 Agadir, Morocco.

Abstract—In this paper, we present a computer vision based system for fast robust Traffic Sign Detection and Recognition (TSDR), consisting of three steps. The first step consists on image enhancement and thresholding using the three components of the Hue Saturation and Value (HSV) space. Then we refer to distance to border feature and Random Forests classifier to detect circular, triangular and rectangular shapes on the segmented images. The last step consists on identifying the information included in the detected traffic signs. We compare four features descriptors which include Histogram of Oriented Gradients (HOG), Gabor, Local Binary Pattern (LBP), and Local Self-Similarity (LSS). We also compare their different combinations. For the classifiers we have carried out a comparison between Random Forests and Support Vector Machines (SVMs). The best results are given by the combination HOG with LSS together with the Random Forest classifier. The proposed method has been tested on the Swedish Traffic Signs Data set and gives satisfactory results.

Keywords—Traffic Sign Recognition (TSR); thresholding; Hue Saturation and Value (HSV); Histogram of Oriented Gradients (HOG); Gabor; Local Binary Pattern (LBP); Local Self-Similarity (LSS); Random forests

I. INTRODUCTION

Advanced driver assistance systems (ADAS) are one of the fastest-growing fields in automotive electronics. ADAS technology can be based upon vision systems [1], active sensors technology [2], car data networks [3], etc. These devices can be utilized to extract various kinds of data from the driving environments. One of the most important difficulties that ADAS face is the understanding of the environment and guidance of the vehicles in real outdoor scenes. Traffic signs are installed to guide, warn, and regulate traffic. They supply information to help drivers. In the real world, drivers may not always notice road signs. At night or in bad weather, traffic signs are harder to recognize correctly and the drivers are easily affected by headlights of oncoming vehicles. These situations may lead to traffic accidents and serious injuries. A vision-based road sign detection and recognition system is thus desirable to catch the attention of a driver to avoid traffic

hazards. These systems are important tasks not only for ADAS, but also for other real-world applications including urban scene understanding, automated driving, or even sign monitoring for maintenance. It can enhance safety by informing the drivers about the current state of traffic signs on the road and giving valuable information about precaution. However, many factors make the road sign recognition problem difficult (see Fig. 1) such as lighting condition changes, occlusion of signs due to obstacles, deformation of signs, motion blur in video images, etc.



Fig. 1: Examples for difficulties facing the traffic sign recognition (TSR) task.

A traffic sign recognition algorithm usually consists of two modules: the detection module and the classification module. The detection module receives images from the camera and finds out all the regions in the images that may contain traffic signs; then the classification module determines the category of traffic sign in each region. The information provided by the traffic signs is encoded in their visual properties: color, shape, and pictogram. Therefore, the detection and the recognition modules are based on color and shape cues of traffic signs. In this paper, we describe a fast system for vision based traffic sign detection and recognition.

The rest of the paper is organized as follows. Section 2 presents an overview of past work on traffic sign detection and recognition. Section 3 details the proposed approach to traffic sign detection and recognition. Experimental results are illustrated in Section 4. Section 5 concludes the paper.

II. OVERVIEW

Many different approaches to traffic sign recognition have been proposed and it is difficult to compare between those approaches since they are based on different data. Moreover, some articles concentrate on subclasses of signs, for example on speed limit signs and digit recognition. This section gives an overview of the techniques used in the TSR and previous works using these techniques. According to the two basic tasks in traffic sign recognition, we simply divide the overview into two categories: traffic sign detection and classification.

A. Traffic Sign Detection

The purpose of traffic sign detection is to find the locations and sizes of traffic signs in natural scene images. The well-defined colors and shapes are two main cues for traffic sign detection. Thus, we can divide the detection methods into two categories: color-based and shape-based. Color-based methods are usually fast and invariant to translation, rotation and scaling. As color can be easily affected by the lighting condition, the main difficulty of color-based methods is how to be invariant to different lighting conditions. These methods tend to follow a common scheme: the image is transformed into a color space and then thresholded. Some authors perform this thresholding directly in RGB (Red Green Blue) space, even if it is very sensitive to illumination changes. To overcome this, simple formulas relating red, green and blue components are employed. For example, Escalera et al. in [4] used different relations between the R, G and B components to segment the desired color. In [5] the difference between R and G, and the difference between R and B channels are employed to form two stable features in traffic sign detection. Ruta et al. in [6], used the color enhancement to extract red, blue and yellow blobs. This transform emphasizes the pixels where the given color channel is dominant over the other two in the RGB color space. In addition to RGB space, other color spaces such as YUV and HSI are also used. For example, The YUV system is considered in [7] to detect blue rectangular signs. In [8] a segmentation method in both La-b and HSI color spaces is used to extract candidate blobs for chromatic signs. At the same time, white signs are detected with the help of an achromatic decomposition. Then a post-processing step is performed in order to discard non-interest regions, to connect fragmented signs, and to separate signs located at the same post.

In the other hand, shape-based methods employ either Haar-like features in frameworks inspired by the popular Viola-Jones detector or the orientation and intensity of image gradients in frameworks inspired by the Generalized Hough Transform. The first sub-category comprises the works by Bahlmann et al. [9] and by Brkic et al. [10], whereas in the second we find the Regular Polygon Detector [11], the Radial Symmetry Detector [12], the Vertex Bisector Transform [13], the Bilateral Chinese Transform and, alike, the two schemes of Single Target Voting for triangles and circles proposed by Houben [14]. Many recent approaches use gradient orientation information in the detection phase, for example, in [11], Edge Orientation Histograms are computed over shape-specific sub-regions of the image. Gao et al. [15] classify the candidate traffic signs by comparing their local edge orientations at arbitrary fixation points with those of the templates. In [16],

the Regions of Interest (ROI) obtained from color-based segmentation are classified using the HOG feature. To integrate color information in the HOG descriptor, Creusen et al. [17] concatenate the HOG descriptors calculated on each of the color channels. The advantages of this feature are its scale-invariance, the local contrast normalization, the coarse spatial sampling and the fine weighted orientation binning.

B. Traffic Sign Recognition

The purpose of traffic sign recognition is to classify the detected traffic signs to their specific sub-classes. Regarding the recognition problem, it is common to use some features with machine learning algorithms. Maldonado et al. in [18] utilized different one-vs-all SVMs with Gaussian kernel for each color and shape classification to recognize signs. In [19] SVMs are used with HOG features to carry out classification on candidate regions provided by the interest region detectors. It withstand great appearance variations thanks to the robustness of local features, which typically occur in outdoor data, especially dramatic illumination and scale changes. Zaklouta [20] uses different sized HOG features, and adopts random forest based classification to achieve high detection accuracy. Tang [21] proposes an efficient method of traffic sign recognition using complementary features to reduce the computation complexity of traffic sign detection, and then uses the SVM to implement the traffic sign classification. The complementary features used in Tang [21] include HOG [22] and LBP [23]. Convolutional Neural Network (CNN) is another method used for traffic sign classification. It is proved in [24] that the performance of CNN on traffic sign classification outperforms the human performance. In [25], a CNN together with a Multi-Layer Perception (MLP), which is trained on HOG features, was used. In [26], a CNN with multi-scale features by using layer-skipping connection is presented. In [1], the authors suggest a hinge loss stochastic gradient descent method to train convolutional neural networks. The method yields to high accuracy rates. However, a high computing cost is paid to train the data when using CNNs.

In general, the quality of the results obtained by any study on TSR varies from one research group to another. It is very difficult to decide which approach gives better overall results, mainly due to the lack of a standard database of road images. It is not possible to know, for example, how well the systems respond to changes in illumination of the images since in the different studies it is usually not specified whether images with low illumination have been used in the experiments. Another disadvantage of the lack of a standardised database of road images is that some studies are based on a small set of images since the compilation of a set of road scene images is a very time-consuming task. The problem with working with such small data sets is that it is difficult to evaluate the reliability of the results.

III. PROPOSED METHOD

The proposed system consists of three stages: segmentation, shape detection and recognition. In the first stage, we aim to segment the images to extract ROIs. In the second one, we detect the desired shapes from the ROIs. In the last stage, we recognize the information included in the detected traffic signs. Fig. 2 illustrates the algorithm scheme of the proposed

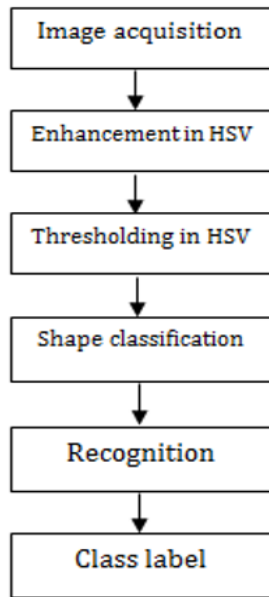


Fig. 2: Algorithm scheme.

method. In this section, we detail each step of the proposed approach.

A. Segmentation

Color segmentation algorithms are influenced by weather condition, daytime, shadows, orientation of objects in relation to the sun and many other parameters. These parameters change frequently in dense urban area scenes. In addition, there are many other objects in the street of the same color as traffic signs (red and blue). Therefore, the color information is only used to generate ROIs without performing classification.

To overcome the difficulties related to illumination changes and possible deterioration of the signs, the HSV color space is used in our system. We implement both enhancement and thresholding techniques. First, we enhance the input image in HSV color space. Then, we segment the image using fixed thresholds. These thresholds were empirically deduced using traffic sign images. The resulting binary image is then post-processed to discard insignificant ROIs and to reduce the number of ROIs to be provided to shape classification stage.

1) *Enhancement*: Approved by many experiments, HSV color space is a good choice for color image enhancement. There is only a weak correlation between HSV components, which indicates that a modification to one component will only slightly change another. Unfortunately, in some situation, the slightly change in HSV will result in great color distortion. In this paper, the hue and saturation component are kept intact and only value component of the input image is subjected for enhancement. This enhancement is done according to two steps: Luminance enhancement and Contrast enhancement.

First, The luminance enhancement is applied to the value component using the formula provided in [27]. Suppose that $V_1(x, y)$ denote the normalized V channel in HSV space and $V_2(x, y)$ be the transferred value by applying nonlinear transfer



Fig. 3: Enhancement results.

function defined below.

$$V_2 = \frac{V_1^{0.75z+0.25} + 0.4(1-z)(1-V_1) + V_1^{(2-z)}}{2} \quad (1)$$

where, z is the image dependent parameter and is defined as follows

$$z = \begin{cases} 0, & \text{for } L \leq 50 \\ \frac{L-50}{100}, & \text{for } 50 < L \leq 150 \\ 1, & \text{for } L > 150 \end{cases} \quad (2)$$

where, L is the value (V) level corresponding to the cumulative probability distribution function (CDF) of 0.1. In equation 2 the parameter z defines the shape of the transfer function or the amount of luminance enhancement for each pixel value.

The second step is the contrast enhancement. In this process, the Gaussian convolution using Gaussian function $G(x, y)$ is carried out on the original V channel of the input image in HSV space. The convolution can be expressed as:

$$V_3(x, y) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} V(m, n)G(m+x, n+y) \quad (3)$$

V_3 in equation 3 denotes the convolution result, which contains the luminance information from the surrounding pixels. The amount of contrast enhancement of the centre pixel is now determined by comparing centre pixel value with the Gaussian convolution result. This process is described in the following equation:

$$V_4(x, y) = 255V_2(x, y)^{E(x, y)} \quad (4)$$

where

$$E(x, y) = \left[\frac{V_3(x, y)}{V_2(x, y)} \right]^g \quad (5)$$

where, g is the parameter determined from the original value component image in HSV space for tuning the contrast enhancement process. This parameter g is determined using following equation:

$$g = \begin{cases} 1.75, & \text{for } \sigma \leq 2 \\ \frac{27-2\sigma}{13}, & \text{for } 2 < \sigma < 10 \\ 0.5, & \text{for } \sigma \geq 10 \end{cases} \quad (6)$$

where, σ denotes the standard deviation of the individual block of the original value component image. The standard deviation is determined globally, as it was done in [27]. Fig. 3 shows an example of image before and after enhancement process.

TABLE I: Thresholds used for road sign detection.

	Red	Bleu
Hue	$0 < H < 12$ Or $300 < H < 360$	$190 < H < 260$
Saturation	$25 < S < 250$	$70 < S < 250$
Value	$30 < V < 200$	$56 < V < 128$



Fig. 4: Segmentation results.

2) *Thresholding*: After the enhancement process, we refer to thresholding to segment the image into ROIs. Each image element is classified according to its hue, saturation, and value. A pixel color is considered as red or blue using the threshold values shown in Table I. The hue obtained H is within the interval $[0, 360]$, the saturation S and intensity I is within $[0, 255]$. We further refer to the achromatic decomposition used in [18] to segment white color. This achromatic decomposition is defined as:

$$f(R, G, B) = \frac{|R - G| + |G - B| + |B - R|}{3D} \quad (7)$$

The R, G and B represent the brightness of respective color. D is degree of extracting an achromatic and it is empirically set to $D = 20$ in [18]. An achromatic color is represented by $f(R, G, B)$ of less than 1, and an $f(R, G, B)$ of greater than 1 represents chromatic colors.

After the segmentation stage, we obtain a binary image with the pixels of interest being white and others black (see Fig. 4(b)). Then, according to the size and the aspect ratio of the blobs, we eliminated noise and blobs considered as non-interest. The limits for both criteria, i.e., size and aspect ratio, were empirically derived based on road images (see Fig. 4(c)).

B. Shape Classification

In this stage, we classify the blobs that were obtained from the segmentation stage according to their shape. We only consider triangular, circular and rectangular shapes. Thus, Distance to Borders (DtBs) [18] are used as feature vectors for the inputs of a random forest classifier. DtBs are the distances from the external edge of the blob to its bounding box. These features are widely used to classify shapes, and show its performance in many traffic sign recognition works. Fig. 5 shows these distances for a triangular shape. After computing these features, a random forest classifier is used to classify the ROIs into appropriate shapes. A Random Forest is an ensemble of classification trees, where each tree contributes with a single vote for the assignment of the most frequent class to the input data. It adds an additional layer of randomness to bagging. In addition to constructing each tree using a different bootstrap sample of the data, random forests change

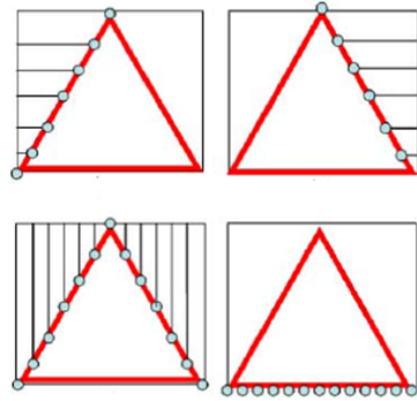


Fig. 5: DtBs for a triangular shape.

how the classification or regression trees are constructed. In standard trees, each node is split using the best split among all variables. In a random forest, each node is split using the best among a subset of predictors randomly chosen at that node. This somewhat counter intuitive strategy turns out to perform very well compared to many other classifiers, and is robust against overfitting [28]. Random Forests have received increasing interest because they can be more accurate and robust to noise than single classifiers [20] [16].

The proposed method is invariant to translation, scale and rotation. First, it is invariant to translation because it does not matter where the candidate blob is. Second, the method is invariant to scale due to the normalization of the DtB vectors to the bounding-box dimensions. Finally, the detection process is invariant to rotation because the most external pixels of each blob are detected to determine the original orientation, and after this, all blobs are oriented in a reference position. In conclusion, samples of DtB vectors show a similar evolution for each geometric shape.

C. Recognition

Once the candidate blobs are classified into a shape class, the recognition process is initiated. The main objectives of this stage to be based on a method with a high accuracy but at the same time, the memory and the complexity of the algorithm used have to be minimized.

In this work, we compare the Random Forests classifier, to the state-of-the-art SVM classifier. As we will see in section 4, random forests performance is better than SVMs in both the accuracy rate and the execution time.

For the feature extraction, inspired by the existing ones, we try to introduce new ones using different combinations. HOG, Gabor filters, LBP, and LSS are used in this work. The performance and the execution time of these features as well as the classifiers ones are shown in section 4.

1) *Features extraction*: We used in this work four kinds of features namely, HOG, Gabor, LBP, and LSS. The first feature used is HOG feature. It was proposed by Navneet Dalal and Bill Triggs [22] for pedestrian detection. The basic idea of HOG features is that the local object appearance and shape can often be characterized rather well by the distribution of the local intensity gradients or

edge directions, even without precise knowledge of the corresponding gradient or edge positions. The method is very simple and fast so the histogram can be calculated quickly.

The second one is Gabor feature. Gabor filters have been applied to many signal processing and pattern recognition problems. They are able to explore the local spectrum characteristics of image. A 2D Gabor filter is a band-pass spatial filter with selectivity to both orientation and spatial frequency [29].

The third one is LBP. It was proposed by T. Ojala [23], and it is a popular texture descriptor. The concept of LBP feature vector, in its simplest form, is similar to the HOGs. The window is divided into cells. For each pixel in a cell, we compare the center pixels value to each of its 8 neighbours, and the pixels value is set to 1 if its value is greater than the center pixels value, or set to 0 otherwise. Then compute the histogram, over the cell, of the frequency of each 'number' occurring, and normalize it to obtain histograms of all cells. This gives the features vector for the window.

The last one is LSS feature. Generally in LSS, The selected image is partitioned into smaller cells which, conveniently compared with a patch located at the image center. The resulting distance surface is normalized and projected into the space intervals partitioned by the number of angle intervals and radial intervals. The maximum value in an interval space would be considered as the value of the feature.

2) *Classifiers*: Two classifiers were used in this work: random forests and SVMs. The results of the comparison are presented in section 4.

As mentioned in III-B, Random Forests have received increasing interest because they can be more accurate and robust to noise than single classifiers. Another advantage of Random Forests is their ease of use in the sense that they have only two parameters (the number of variables in the random subset at each node and the number of trees in the forest), and is usually not very sensitive to their values. The main idea of random forests consists of an arbitrary number of simple trees, where the final predicted class for a test object is the mode of the predictions of all individual trees.

In the other hand, SVMs are used to extend our study of classifiers for TSR. The algorithm attempts to separate the positive examples of negative examples. The basic concept of SVM is to transform the input vectors to a higher dimensional space by a nonlinear transform, and then an optimal hyperplane that separates the data, can be found. This hyperplane should have the best generalization capability. In many cases, the data cannot be separated by a linear function. The use of a kernel function becomes essential in these cases. SVM is designed to solve a binary classification problem. However, for a road sign inventory problem, which is a multiple classification problem, classification is accomplished through combinations of binary classification problems. There are two ways to do that: one-vs.-one or one-vs.-all.

IV. EXPERIMENTAL RESULTS

This section presents the results obtained by the proposed approach. Evaluation of the classifiers as well as the features presented in III-C1 are presented to justify the choice of the proposed system. All the tests were performed on the public STS data set [30] using a 2.7 GHz Intel i5 processor.

A. Data Set

We implement our method on the Swedish Traffic Sign data set (STSD). It is a public data set which contains sequences videos and includes more than 20 000 images in which 20% of the images are labeled. It contains 3488 traffic signs. The images in STSD are obtained from highways and cities record from more than 350 km of Swedish roads Fig. 6.



Fig. 6: Examples of STS Data Set images.

B. Traffic Sign Detection

The evaluation of the detection stage is performed based on precision-recall curve, where the recall and precision values are computed as follows:

$$recall = \frac{\text{Number of correctly detected signs}}{\text{Number of true signs}} \times 100 \quad (8)$$

$$precision = \frac{\text{Number of correctly detected signs}}{\text{Number of detected signs}} \times 100 \quad (9)$$

The precision-recall curves of the proposed method when applied to STS data set are depicted in Fig. 7. The best trade-off between the recall and precision values as well as the Area Under Curve (AUC) of the detection module are listed in Table II. It can be seen that the method yields the best results with recall of 93.41% at a precision of 95.12%. The AUC of the precision-recall curve is 94.50%.

Fig. 8(a) shows an example among images used to test the proposed detection approach. The corresponding segmentation

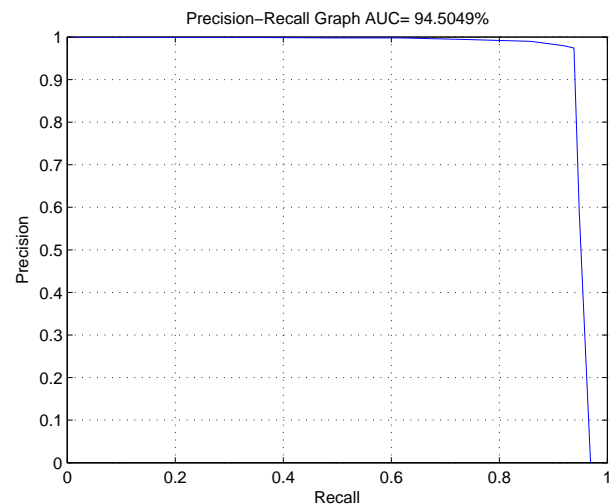


Fig. 7: Precision-recall curve of the proposed detection approach when applied STS data sets.

TABLE II: The best trade-off between the recall and precision values as well as the AUC obtained by the detection method on STS data set in %.

	Recall	Precision	AUC
On the STS Data set	93.41	95.12	94.50

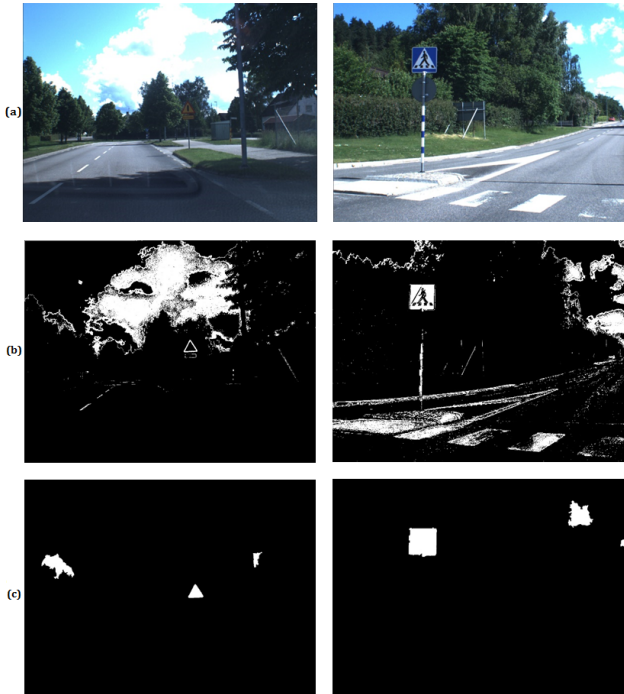


Fig. 8: Example of test images (a) and (b), (c) segmentation phase's results.



Fig. 9: Final detection results by the proposed detection method.

results with and without using size and aspect ratio constraints are illustrated in Fig. 8(b). Referred to these figures, some regions are discarded as non-interest objects according to their size and aspect ratio. Therefore, the detection process can be reduced as the number of ROIs is reduced. The segmentation method succeeds to detect the road sign present in Fig. 8(a) among the extracted ROIs in Fig. 8(b). However, some ROIs have been detected even they do not represent road signs. The shape classification method has been applied to the ROIs in Fig. 8(c). The DtBs of the extracted ROIs have been computed and feed to random forest classifier. Fig. 9 shows the final detection results by the proposed detection method. Red bounding box represents detected region of traffic sign.

C. Traffic Sign Recognition

To evaluate the recognition stage, a comparison between features and classifiers used in the system is performed. To obtain optimal design parameters of each descriptor, we run some cross-validation experiments on the training dataset: divide the training images into a basic training set and a validation set. By training classifier on the basic training set and evaluate on the validation set, we selected the setting of maximum validation accuracy and a setting of lower-dimensionality. After that, the classifier is re-trained on the whole training set with selected feature extraction settings.

To compute the HOG feature vector, we normalize the window detected in the previous stage to 40×40 , and the normalized image is divided into 8×8 overlapping blocks, which gives us a total number of 49 blocks. Each one of these blocks is divided to 2×2 cells, and each cell contains a 5×5 pixels. In each cell, we obtain a gradient histogram of 9 bins. For the Gabor feature, we used tow scales and eight orientations. The window is partitioned into 16×16 blocks and sampling interval varies according to the block sizes. For the LBP feature: we employ a basic LBP descriptor to compute the LBP features. The normalized window is partitioned into 6×6 non-overlapping blocks. Using the uniform patterns method, we extract 59 features per block, and finally form the LBP feature vector. The last feature is LSS. It has four primary parameters: the size of image, the radius of window, the interval radius of image patches and angle interval. These parameters are closely associated with each other. In our implementation, we used 3×3 patches, correlated against a surrounding window with radius equal to 10. Our log-polar coordinates was partitioned into 80 bins (20 angles and 4 radial intervals).

After calculating the four different features individually, we concatenate them to form new features. In Table III, different compound features are listed to compare their performance on the STS data set. As we can see from Table III among the four single feature descriptors, the HOG feature has a Correct Classification Rate (CCR) of 95.38%, higher than the Gabor, LBP, and LSS features. According to the results, combining two different features can improve classification accuracy evidently. Particularly, the combination of HOG and LSS features gives a CCR of 96.13%, significantly better than the best single feature HOG or LSS. Each combination of two features outperforms its constituent single features. This confirms that the different features are complementary.

Table III gives also a comparison between the state-of-the-art SVMs with radial basis function (RBF) kernel, $C = 7$ and $G = 0.09$ and Random Forest with 600 trees and 100 variables, in the terms of CCR and running time. It is obvious from the table that the Random Forest classifier provides accurate results with less running time when compared to the SVM classifier. Thus, we have adopted in the proposed recognition method the Random Forest classifier together with the HOG+LSS features.

Figs. 10, 11 and 12 illustrate examples of recognition results when the proposed approach is applied to images of various traffic environments. In Fig. 10, the traffic signs contained in the images have been successfully detected and

TABLE III: The CCR and the average running time of the classifiers and features used in this work.

Feature	Random Forests		SVMs	
	CCR(%)	Execution Time(ms)	CRR(%)	Execution Time(ms)
HOG	95.38	20.82	94.52	37.36
Gabor	94.89	31.21	94.23	41.67
LBP	94.63	12.30	94.02	18.45
LSS	94.24	10.72	3.84	13.38
HOG+Gabor	95.59	38.51	95.15	58.96
HOG+LBP	95.94	25.23	95.46	43.48
HOG+LSS	96.13	22.18	95.68	42.81
Gabor+LBP	95.39	33.81	94.79	49.34
Gabor+LSS	95.42	33.47	94.47	49.52
LBP+LSS	95.29	13.90	94.44	20.91



Fig. 10: Detection and recognition results.



Fig. 11: Examples of recognition with misdetection.



Fig. 12: Examples of recognition with confused classification.

recognized. In Fig. 11, the system was not able to detect traffic signs. Consequently, the ROIs corresponding to the signs were not feed to the recognition stage. In Fig. 12, the traffic signs contained in the images have been successfully detected. However, the system could not recognize them due to the motion blur in the signs.

V. CONCLUSION

In this paper, a fast system for Traffic Sign Detection and Recognition was described. In the first stage, we refer to color segmentation to reduce the search space. We used an enhancement then a thresholding on the HSV color space. In the second stage, the circular, rectangular and triangular signs are

detected using the Distance to Border feature and a Random Forest classifier. The detected candidates are identified using The Random forests classifier with a combination of HOG and LSS features. The system achieves correct classification rate of over 96% at a processing rate of 8–10 frames/s. In the future work, we can use adaptive thresholds to overcome the color segmentation problems. Temporal information could also be integrated to track the detected traffic signs and reinforce the decision making process. This would also allow us to restrict the search space in the current image considering previous detections information, which can accelerate the candidate detection.

REFERENCES

- [1] J. Jin, K. Fu, and C. Zhang, "Traffic sign recognition with hinge loss trained convolutional neural networks," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 15, no. 5, pp. 1991–2000, 2014.
- [2] S. P. Biswas, P. Roy, N. Patra, A. Mukherjee, and N. Dey, "Intelligent traffic monitoring system," in *Proceedings of the Second International Conference on Computer and Communication Technologies*. Springer, 2016, pp. 535–545.
- [3] A. Mukherjee, S. Chakraborty, A. T. Azar, S. K. Bhattacharyay, B. Chatterjee, and N. Dey, "Unmanned aerial system for post disaster identification," in *Circuits, Communication, Control and Computing (I4C), 2014 International Conference on*. IEEE, 2014, pp. 247–252.
- [4] A. De La Escalera, L. E. Moreno, M. A. Salichs, and J. M. Armingol, "Road traffic sign detection and classification," *Industrial Electronics, IEEE Transactions on*, vol. 44, no. 6, pp. 848–859, 1997.
- [5] M. Bénallal and J. Meunier, "Real-time color segmentation of road signs," in *Electrical and Computer Engineering, 2003. IEEE CCECE 2003. Canadian Conference on*, vol. 3. IEEE, 2003, pp. 1823–1826.
- [6] A. Ruta, Y. Li, and X. Liu, "Real-time traffic sign recognition from video by class-specific discriminative features," *Pattern Recognition*, vol. 43, no. 1, pp. 416–430, 2010.
- [7] J. Miura, T. Kanda, and Y. Shirai, "An active vision system for real-time traffic sign recognition," in *Intelligent Transportation Systems, 2000. Proceedings. 2000 IEEE*. IEEE, 2000, pp. 52–57.
- [8] J. Lillo-Castellano, I. Mora-Jiménez, C. Figuera-Pozuelo, and J. Rojo-Álvarez, "Traffic sign segmentation and classification using statistical learning methods," *Neurocomputing*, vol. 153, pp. 286–299, 2015.
- [9] C. Bahlmann, Y. Zhu, V. Ramesh, M. Pellkofer, and T. Koehler, "A system for traffic sign detection, tracking, and recognition using color, shape, and motion information," in *Intelligent Vehicles Symposium, 2005. Proceedings. IEEE*. IEEE, 2005, pp. 255–260.
- [10] S. Šegvić, "Traffic sign detection as a component of an automated traffic infrastructure inventory system," in *33rd annual Workshop of the Austrian Association for Pattern Recognition (OAGM/AAPR)*. Hrvatska znanstvena bibliografija i MZOS-Svibor, 2009.
- [11] G. Loy and N. Barnes, "Fast shape-based road sign detection for a driver assistance system," in *Intelligent Robots and Systems, 2004.(IROS*

- 2004). *Proceedings. 2004 IEEE/RSJ International Conference on*, vol. 1. IEEE, 2004, pp. 70–75.
- [12] N. Barnes, A. Zelinsky, and L. S. Fletcher, “Real-time speed sign detection using the radial symmetry detector,” *Intelligent Transportation Systems, IEEE Transactions on*, vol. 9, no. 2, pp. 322–332, 2008.
- [13] R. Belaroussi and J.-P. Tarel, “Angle vertex and bisector geometric model for triangular road sign detection,” in *Applications of Computer Vision (WACV), 2009 Workshop on*. IEEE, 2009, pp. 1–7.
- [14] S. Houben, “A single target voting scheme for traffic sign detection,” in *Intelligent Vehicles Symposium (IV), 2011 IEEE*. IEEE, 2011, pp. 124–129.
- [15] X. W. Gao, L. Podladchikova, D. Shaposhnikov, K. Hong, and N. Shevtsova, “Recognition of traffic signs based on their colour and shape features extracted using human vision models,” *Journal of Visual Communication and Image Representation*, vol. 17, no. 4, pp. 675–685, 2006.
- [16] X. Qingsong, S. Juan, and L. Tiantian, “A detection and recognition method for prohibition traffic signs,” in *Image Analysis and Signal Processing (IASP), 2010 International Conference on*. IEEE, 2010, pp. 583–586.
- [17] I. M. Creusen, R. G. Wijnhoven, E. Herbschleb, and P. De With, “Color exploitation in hog-based traffic sign detection,” in *Image Processing (ICIP), 2010 17th IEEE International Conference on*. IEEE, 2010, pp. 2669–2672.
- [18] S. Maldonado-Bascón, S. Lafuente-Arroyo, P. Gil-Jimenez, H. Gómez-Moreno, and F. López-Ferreras, “Road-sign detection and recognition based on support vector machines,” *Intelligent Transportation Systems, IEEE Transactions on*, vol. 8, no. 2, pp. 264–278, 2007.
- [19] S. Salti, A. Petrelli, F. Tombari, N. Fioraio, and L. Di Stefano, “Traffic sign detection via interest region extraction,” *Pattern Recognition*, vol. 48, no. 4, pp. 1039–1049, 2015.
- [20] F. Zaklouta and B. Stanculescu, “Real-time traffic-sign recognition using tree classifiers,” *Intelligent Transportation Systems, IEEE Transactions on*, vol. 13, no. 4, pp. 1507–1514, 2012.
- [21] S. Tang and L.-L. Huang, “Traffic sign recognition using complementary features,” in *Pattern Recognition (ACPR), 2013 2nd IAPR Asian Conference on*. IEEE, 2013, pp. 210–214.
- [22] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 886–893.
- [23] T. Ojala, M. Pietikäinen, and D. Harwood, “A comparative study of texture measures with classification based on featured distributions,” *Pattern recognition*, vol. 29, no. 1, pp. 51–59, 1996.
- [24] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, “The german traffic sign recognition benchmark: a multi-class classification competition,” in *Neural Networks (IJCNN), The 2011 International Joint Conference on*. IEEE, 2011, pp. 1453–1460.
- [25] D. Cireşan, U. Meier, J. Masci, and J. Schmidhuber, “A committee of neural networks for traffic sign classification,” in *Neural Networks (IJCNN), The 2011 International Joint Conference on*. IEEE, 2011, pp. 1918–1921.
- [26] P. Sermanet and Y. LeCun, “Traffic sign recognition with multi-scale convolutional networks,” in *Neural Networks (IJCNN), The 2011 International Joint Conference on*. IEEE, 2011, pp. 2809–2813.
- [27] L. Tao and V. Asari, “An integrated neighborhood dependent approach for nonlinear enhancement of color images,” in *Information Technology: Coding and Computing, 2004. Proceedings. ITCC 2004. International Conference on*, vol. 2. IEEE, 2004, pp. 138–139.
- [28] L. Breiman, “Random forests,” *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.
- [29] J. G. Daugman, “Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters,” *JOSA A*, vol. 2, no. 7, pp. 1160–1169, 1985.
- [30] <https://www.cvl.isy.liu.se/research/datasets/traffic-signs-dataset/>.