Applied Soft Computing xxx (2016) xxx–xxx

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Traffic sign detection and recognition based on random forests

Q1 Ayoub Ellahyani*, Mohamed El Ansari, Ilyas El Jaafari

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

ARTICLE INFO

Article history:
Received 15 June 2015
Received in revised form
23 December 2015
Accepted 31 December 2015
Available online xxx

Keywords:
Traffic sign detection
Traffic sign recognition
Color segmentation
Random forests
Support vector machines (SVMs)
Histogram of oriented gradients (HOG)
Local self-similarity (LSS)

ABSTRACT

In this paper we present a new traffic sign detection and recognition (TSDR) method, which is achieved in three main steps. The first step segments the image based on thresholding of HSI color space components. The second step detects traffic signs by processing the blobs extracted by the first step. The last one performs the recognition of the detected traffic signs. The main contributions of the paper are as follows. First, we propose, in the second step, to use invariant geometric moments to classify shapes instead of machine learning algorithms. Second, inspired by the existing features, new ones have been proposed for the recognition. The histogram of oriented gradients (HOG) features has been extended to the HSI color space and combined with the local self-similarity (LSS) features to get the descriptor we use in our algorithm. As a classifier, random forest and support vector machine (SVM) classifiers have been tested together with the new descriptor. The proposed method has been tested on both the German Traffic Sign Detection and Recognition Benchmark and the Swedish Traffic Signs Data sets. The results obtained are satisfactory when compared to the state-of-the-art methods.

1. Introduction

Q3

Advanced Driver Assistance Systems (ADAS) play an important role in enhancing car safety and driving comfort. One of the most important difficulties that ADAS face is the understanding of the environment and guidance of the vehicles in real outdoor scenes [1]. Humans driving is a task based almost entirely on visual information, and one of the tasks in successful driving involves the identification of traffic signs. Traffic signs provide information about the current state of the road, restrictions, prohibitions, warnings, and other helpful information for navigation. The information provided by the road signs is encoded in their visual traits: shape, color and pictogram.

Road sign recognition has been a challenge problem for many years and is an important task not only for ADAS, but also for other real-world applications including urban scene understanding, automated driving, or even sign monitoring for maintenance. It is a relatively constrained problem in the sense that signs are unique, rigid, intended to be clearly visible for drivers, and have little variability in appearance Fig. 1. However, there are many factors that make the road sign recognition problem difficult such as:

- The colors of road signs, particularly red, may fade after long exposure to the sun Fig. 1(a).
- Air pollution and weather conditions (e.g. rain, snow, fog, shadows, and clouds) may decrease the visibility of road signs Fig. 1(b).
- Outdoor lighting conditions varying from day to night may affect the colors of road signs Fig. 1(c).
- Obstacles, such as vehicles, pedestrians, and other road signs, may partially occlude road signs Fig. 1 (d).
- Video images of road signs will have motion blur if the camcorder is mounted on a moving vehicle due to vehicle vibration as well as motion Fig. 1(e).

In this paper, we present a new traffic sign detection and recognition approach including three stages. The first stage segments the images to extract ROIs. The segmentation is usually performed based on the color information, which is known a priori [2–5]. The second one detects traffic shapes. Given that the geometric form of traffic signs is limited to triangular, circular, rectangular and octagonal forms, the geometric information is used to identify traffic shapes from ROIs provided by the first stage. Most of authors use machine learning algorithms such as SVMs and neural networks (NNs) to classify shapes provided by the segmentation step [5–7]. In this paper, we propose to use the invariant geometric moments with a simple metric to match the ROIs provided by the segmentation process with triangular, circular and rectangular shapes. It gives better results in a lower processing time compared to machine learning algorithms.

Q2 * Corresponding author. Tel.: +212 672956896.
E-mail addresses: ayoub.ellahyani@gmail.como, rellahyani.ayoub@edu.uiz.ac.ma (A. Ellahyani), m.elansari@uiz.ac.ma, melansari@gmail.com (M.E. Ansari), eljaafari.ilyas@gmail.com (I.E. Jaafari).

http://dx.doi.org/10.1016/j.asoc.2015.12.041
1568-4946/

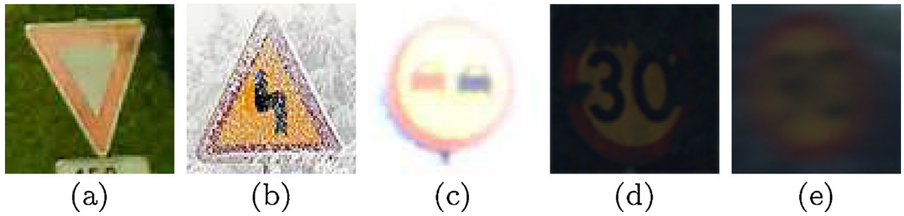

Fig. 1. Examples for difficulties facing the traffic sign recognition (TSR) task.

The third stage recognizes the traffic signs based on the information included in their pictograms. The new method constitutes an improvement of the one presented recently in [8] where grey level-based HOG features have been used instead of HSI-based ones, which are adopted in the current work. Thus, we integrated the color information into the HOG features by using the HSI components to compute the descriptor instead of gray-scale images. Moreover, in this work, we combined the HOG features computed from the HSI color space with the LSS features to form a new descriptor. These features were provided to the Random Forest classifier to perform the 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.

2. Related work

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. Regarding the detection problem, different approaches have been proposed. In the older studies, e.g. [2,6], as well as in many recent ones, e.g. [9–12], it was common to employ color segmentation [2–4,13,5]. Some authors perform this 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 [2] used different relations between the R, G and B components to segment the desired color. In [3] 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 [4], 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 [13] to detect blue rectangular signs. In [9] a segmentation method in both L-a-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.

Another cue used to identify traffic signs is the geometric information. Those shape-based algorithms are generally used directly on scene images, or as a second step after color segmentation. In [2,14,15] a corner detector is used to identify the shape information. Maldonado et al. in [16] used a signature defined as the distance from the mass center to the edge of the blob as a function of the angle to classify blobs as, triangles, squares, or circles. Gavrilla et al. [6] used Distance Transform (DT) and Template Matching (TM) to detect circular and triangular signs. Similarly, Ruta et al. [4] used the Color Distance Transform, where a DT is computed for every color channel separately. Larsson et al. [17] used locally segmented contours combined with an implicit star-shaped object model as prototypes for the different sign classes. The contours are described by Fourier descriptors. Hough transform is another technique employed to detect shapes. In [18] a proprietary and undisclosed algorithm is used to detect rectangles, and Hough Transform for the detection of circles. Loy and Zelinsky [19] proposed a technique similar to Hough transform called fast radial transform, which was successfully used for sign detection in [14,20]. Many recent approaches use gradient orientation information in the detection phase, for example, in [7], Edge Orientation Histograms are computed over shape-specific sub-regions of the image.

After the localisation of region of interests ROIs, classification techniques employed to determine the content of the detected traffic signs. Learning approaches are the most used techniques. Maldonado et al. in [5] utilized different one-vs-all Support Vector Machines (SVMs) with Gaussian kernel for each color and shape classification to recognize signs. In [10] 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. In [12], the authors suggest a hinge loss stochastic gradient descent method to train convolutional neural networks (CNNs). The method yields to high accuracy rates. However, a high computing cost is paid to train the data. Lim et al. in [21] used also Neural Networks (NNs), and improved their results by preselecting the color-shape features using Principal Components Analysis (PCA) and Fisher Linear Discriminant. Many other researchers use Nearest Neighbour approaches to classify traffic signs. For example, Kuo et al. in [15] used K-d tree to identify the content of the sign and yields to high accuracy rates. In [13], the identification of signs is carried out by a normalized correlation-based pattern matching using a traffic-sign database.

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.

3. Proposed method

As depicted in Fig. 2, the proposed method is achieved in three main steps. The first one segments the images to extract ROIs. The second one detects the shapes from the ROIs. The last step

Please cite this article in press as: A. Ellahyani, et al., Traffic sign detection and recognition based on random forests, Appl. Soft Comput. J. (2016), http://dx.doi.org/10.1016/j.asoc.2015.12.041

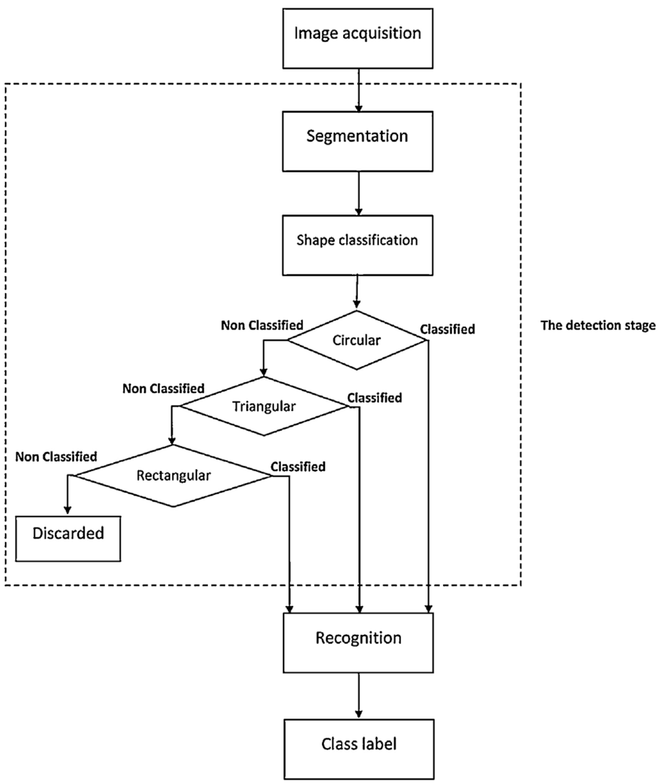

Fig. 2. Algorithm scheme.

Table 1
Threshold values used for the ROIs extraction.

	Red	Blue
Hue	$0 \leq H \leq 10$ Or $300 \leq H \leq 360$	$190 \leq H \leq 260$
Saturation	$25 \leq S \leq 250$	$70 \leq S \leq 250$
Intensity	$30 \leq I \leq 200$	$56 \leq I \leq 128$

recognizes the information included in the detected traffic signs. In this section, we detail each step of the proposed approach.

3.1. Segmentation

Color segmentation algorithms are influenced by weather condition, day time, shadows, orientation of objects in relation to the sun and many other parameters [22]. 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 HSI color space is used in our system. Each image pixel is classified according to its hue, saturation, and intensity using selected thresholds for red and blue colors. These thresholds were deduced from the analysis of the histograms of hue, saturation, and intensity components corresponding to the red and blue manually segmented signs. The signs used to extract these histograms were collected from the German Traffic Sign Detection Benchmark (GTSDB) data set. Here, the hue takes values ranging from 0 to 360, the saturation S and intensity I take values ranging from 0 to 255. The thresholds considered for the segmentation step in the current paper are depicted in Table 1.

The thresholds mentioned in Table 1 allow to extract ROIs with blue and red colors only. The achromatic decomposition can be used to extract white ROIs (signs) from the images as it was done in [5] and [23] according to

$$f(R, G, B) = \frac{(|R-G| + |G-B| + |B-R|)}{3D} \quad (1)$$

where R, G, and B represent the brightness of the respective color. D is the degree of extraction of an achromatic color. Referred to [5], the best segmentation is provided with $D = 20$. Achromatic and chromatic colors are represented by an $f(R, G, B)$ less than 1 and an $f(R, G, B)$ greater than 1, respectively.

The segmentation step provides binary images where the ROIs are represented with white pixels. Insignificant ROIs are discarded based on size and aspect ratio constraints. A ROI is considered as significant if:

- its aspect ratio is between 1/1.9 and 1.9;
- the area covered by the ROI is between $wh/25$ and $wh/3$, where w and h are the image width and height respectively.

The size (area of ROI) and aspect ratio thresholds are selected empirically using collected images from the (GTSDB) data set.

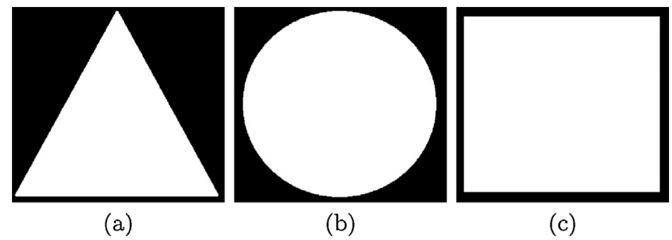

Fig. 3. Binary patches of the possible sign shapes. (a) Triangular shape. (b) Circular shape. (c) Rectangular shape.

3.2. Shape classification

The approach used to classify shapes from extracted ROIs is described in this section. Most of the methods available in the literature call some classifiers, e.g. SVM, to detect the sign shape. Here, a simple invariant geometric moments based method is used to achieve shape classification. Invariant moments are introduced in [24] where the famous Hu's seven invariant geometric moments were derived. Hu described his h_1 to h_6 moments as absolute orthogonal invariants (independent of position, size, and orientation) and h_7 as a skew orthogonal invariant (useful in distinguishing mirror images). These features are capable of recognizing simple objects.

The shapes need to be recognized are circles, triangles, and rectangles. They are all simple objects and we believe the invariant moments can help to recognize them perfectly. The ROIs are binary patches to provide to our shape classification system. The invariant moments are computed for each ROI and compared with those of the target patches of the possible three shapes. Binary patches of the different shapes are created as illustrated in Fig. 3. Note that the octagonal shapes are considered belonging to the same shape class as the circular ones.

Among the detected ROIs in the segmentation step, only those having their moments close to those of the target shapes will be considered as valid shape classes. Different metrics have been tested to match the ROIs with the appropriate shape classes. The metric used is defined as follows:

$$I(A, B) = \sum_{i=1}^{7} |m_i^A - m_i^B| \quad (2)$$

Please cite this article in press as: A. Ellahyani, et al., Traffic sign detection and recognition based on random forests, Appl. Soft Comput. J. (2016), http://dx.doi.org/10.1016/j.asoc.2015.12.041

where m_i^A and m_i^B are defined as follows:

$$m_i^A = sign(h_i^A) \log |h_i^A| \tag{3}$$

$$m_i^B = sign(h_i^B) \log |h_i^B| \tag{4}$$

where h_i^A and h_i^B are the values of the Hu moments of the ROI and the patch respectively.

The metric I will be used to find correspondences between the detected ROIs and the patches in Fig. 3. After computing metrics of the ROI over the three patches, the metric with the minimum value indicate the class of the shape. Note that a ROI is rejected if its corresponding metric value is above a threshold which was empirically derived based on collected images from the (GTSDB) data set.

3.3. Recognition

Once the candidate ROIs (blobs) are classified into a shape class they are provided to the recognition module in charge of identifying the sign. Most of the road signs contain a pictogram, a string of characters, or both. The recognition module is a classifier which should be fed with features describing the signs to identify. Different classifiers as well as different features have been used in the literature. To decide which classifier we will use in the proposed approach, the Random Forest and SVM classifiers have been tested on using different features. Once the classifier is selected, different features have been considered such as Histogram of Oriented Gradient (HOG), Local Binary Pattern (LBP), and Local Self-Similarity (LSS). Firstly, all the features have been tried independently with the Random Forests classifier. Secondly, combinations between those features have been used to create new ones.

In the following, overviews of the Random Forest classifier, SVM classifier, and the features used to test these classifiers are given. The method followed to decide which feature to be used is detailed and the recognition system is deduced from the comparison results.

3.3.1. Random forests

Random forests have received increasing interest because they can be more accurate and robust to noise than single classifiers [25,26]. It 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 [27]. 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 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 counterintuitive strategy turns out to perform very well compared to many other classifiers, and is robust against over-fitting [27]. Another advantage of Random Forests is their ease of use in the sense that they have only two parameters, e.g. the number of variables in the random subset at each node and the number of trees in the forest. The Random forests are not very sensitive to the values of the two parameters.

3.3.2. Support vector machine (SVMs)

SVMs were introduced first by Vapnik [28] and some extensive introductions were presented later in [29]. SVM attempts to separate the positive samples from negative ones. Each sample should be represented by a vector of dimension n. The basic concept of SVM is to transform the input vectors to a higher dimensional space by a nonlinear transform, and then an hyperplane which separates the data can be found. This hyperplane should have the best generalization capability. As shown in Fig 4, the black circles and the white circles are the training data set which belong to two classes. The hyperplane H that separates the positive samples from the negative ones is found, ensuring that the margin between the closest positives and negatives is maximal. Hyperplanes $H1$ and $H2$ are the border of each class. The data located on $H1$ and $H2$ are called support vectors.

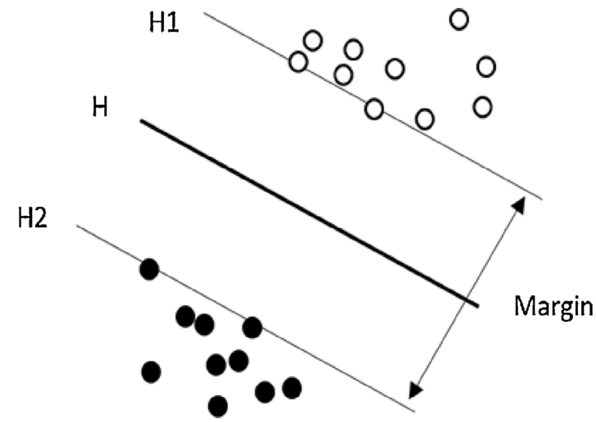

Fig. 4. The SVM binary classification.

SVM is designed to solve binary classification problems. Solving multi-classes problems is accomplished through combinations of binary classification problems. There are two ways to achieve that such us one-vs.-one or one-vs.-all.

3.3.3. Feature extraction

Here, the features used in the recognition step such as HOG, LBP and LSS are presented. A comparison is done to decide which of those features will be used in the signs recognition problem.

HOG features was proposed by Navneet Dalal and Bill Triggs in [30] for pedestrian detection. Motivated by its success in pedestrian detection, HOG features used to recognize traffic sign in many recent works [11,25,10]. 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.

In this work, we propose to extend HOG features to HSI color space. Instead of computing those features from grayscale images, we do compute them from HSI color images.

To compute the HOG features, we normalize the window detected in the previous stage to 40×40. The normalized window is divided into 8×8 overlapping blocks, which gives a total number of 49 blocks. Each one of these blocks is divided into 2×2 cells with 5×5 pixels. The gradient histogram with 9 bins is computed at each cell. At each HSI channel, a 1764 HOG vector is computed. The HOG vector corresponding to the HSI space, named HSI-HOG, is deduced by concatenating the three vectors obtained at each HSI channel. This results in a 5292 HSI-HOG vector. Fig. 5 illustrates how the HSI-HOG features are computed.

The second feature involved in our experiments is the LBP. It is a texture descriptor which was introduced in [31]. The concept of LBP feature vector is similar to the one of the HOG features. The window is divided into cells. For each pixel in a cell, we compare the center pixel value to the neighboring ones and considering the result as a binary number. 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.

Please cite this article in press as: A. Ellahyani, et al., Traffic sign detection and recognition based on random forests, Appl. Soft Comput. J. (2016), http://dx.doi.org/10.1016/j.asoc.2015.12.041

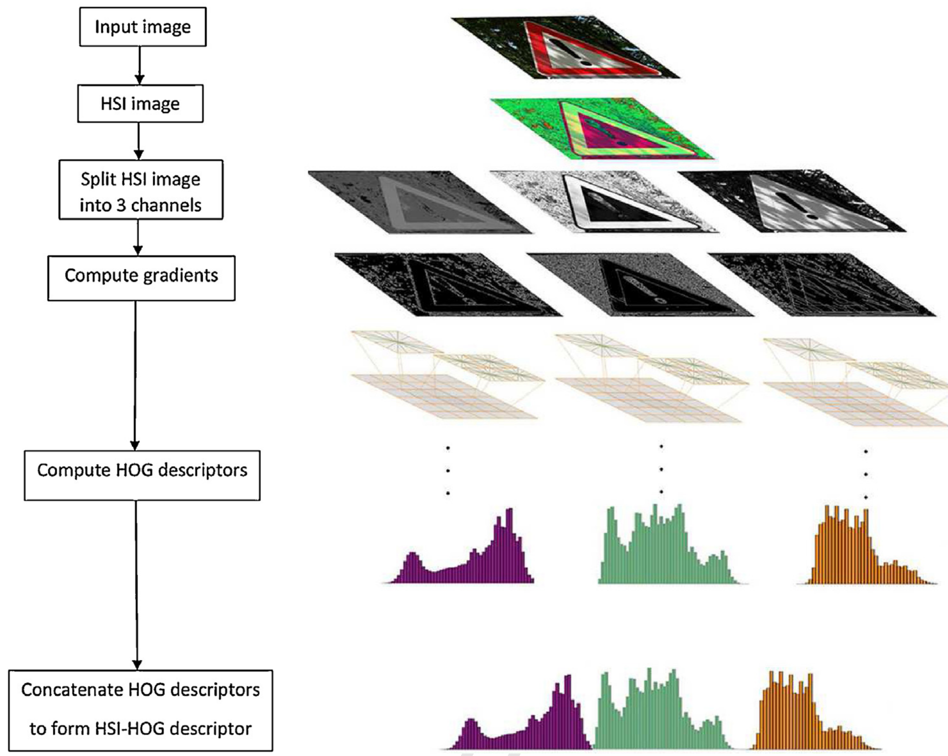

Fig. 5. An example on how the HSI-HOG descriptor is computed.

HOG performs poorly when the background is cluttered with noisy edge points [31]. The concatenation between the HOG and LBP features, to create new features called HOG+LBP, allows to reduce the effect of the noise on the recognition results. Given that the result obtained by HSI-HOG is better than the classic grayscale HOG, the concatenation HSI-HOG with LBP, named HSI-HOG+LBP, is used instead of HOG+LBP.

LSS [32] is the third feature adopted in the current paper. 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. LSS 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 windows with radius equal to 10. Our log-polar coordinates was partitioned into 80 bins (20 angles and 4 radial intervals). LSS features was concatenated with the HSI-HOG features to form a descriptor, named HSI-HOG+LSS. This descriptor is used by a classifier to recognize traffic signs.

In this work, we compared many features to figure out which ones give the best results. First, a simple HOG on grayscale image is used, then we tried to use HOG on color image: we computed the HOG features for the three channels of the HSI color image, and concatenate these features to form HSI-HOG features. Moreover, we combine the HSI-HOG features with the LBP features to form an augmented feature vector. This combination is used successfully to detect humans in [33]. We also used the combination HSI-HOG with LSS features to form another feature vector.

The ROIs provided by the shape classification step are normalized to 40×40. For each normalized ROI, the HOG descriptor for the three channels Hue, Saturation, and Intensity are computed. The so-called HSI-HOG is created by concatenating the HOGs of the three HSI channels. The final HSI-HOG+LSS vector is feed to random forests classifier to classify the detected shapes.

4. Experimental results

This section presents the results obtained by the proposed approach. Evaluation of the classifiers as well as the features presented in Section 3.3.3 are presented to justify the choice of the proposed system. All the tests were performed on the public GTSRB, GTSDB [34], and the STS data sets [35] using a 2.7 GHz Intel i5 processor. A comparison with the state-of-the-art methods is given to assess the performance of the new method.

4.1. Data sets

The public available data sets called German Traffic Sign Recognition Benchmark (GTSRB), German Traffic Sign Detection Benchmark (GTSDB), and Swedish Traffic Signs (STS) are adopted for the performance evaluation. The GTSRB data set contains 51,839 German traffic signs in 43 classes (39,209 training images and 12,630 test images). These 43 classes of traffic signs have been divided into six subsets: speed limit, other prohibitory, derestriction, mandatory, danger, and unique signs Fig. 6. The GTSDB data set provides 900 full images (600 for training, 300 for testing).

The STS data set contains more than 20000 images in which 20% of the images are labeled. It contains 3488 traffic signs recorded from more than 350 km of Swedish roads.

For the evaluation of the detection stage (segmentation and shape classification), we used both GTSDB and STS data sets. The GTSRB data set will be used to compare our recognition module with other state-of-the-art methods.

Please cite this article in press as: A. Ellahyani, et al., Traffic sign detection and recognition based on random forests, Appl. Soft Comput. J. (2016), http://dx.doi.org/10.1016/j.asoc.2015.12.041

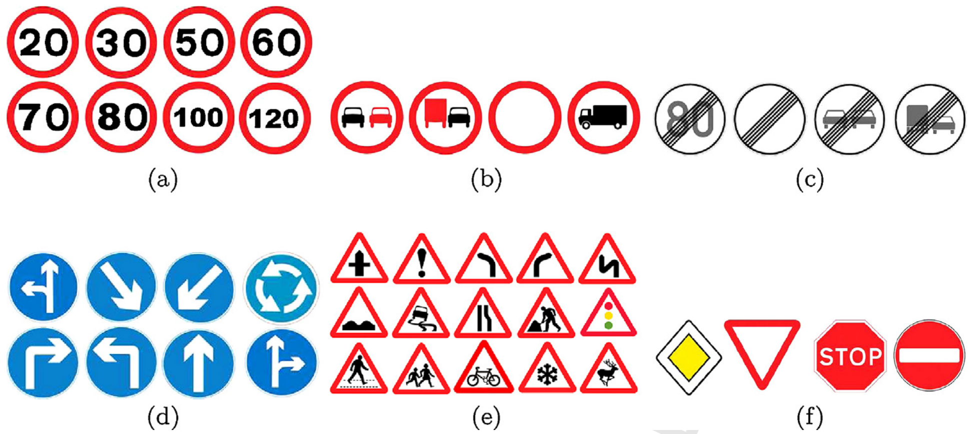

Fig. 6. Subsets of traffic signs in the GTSRB data set. (a) Speed limit signs. (b) Other prohibitory signs. (c) Derestriction signs. (d) Mandatory signs. (e) Danger signs. And (f) unique signs.

4.2. Segmentation and shape classification

In the following of the paper, we refer to shape detection stage as the segmentation step followed by the shape classification one of the proposed method for traffic sign detection and recognition. To evaluate the detection stage, we used two different data. The first one is formed from 300 images of the GTSDB data set. The second one contains 5000 images of the STS data set. All the images were normalized to 640×480 pixels using bilinear interpolation.

Fig. 7(a) shows an example among images used to test the proposed detection approach. The corresponding segmentation results with and without using size and aspect ratio constraints are illustrated in Figs. 7(b) and (c), respectively. Referred to these Figs, 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. 7(a) among the extrcated ROIs in Fig. 7(c). However, some ROIs have been detected even they do not represent road signs.

The shape classification method has been applied to the three ROIs in Fig. 7(c). The moments invariants of the extracted ROIs have been computed and matched to those of the models in Fig. 3 based on the metric depicted in 2. Fig. 7(d) shows the shape classification results. The appropriate shape has been assigned to the first ROI, which is classified as circular. No shapes have been assigned to the two other ROIs. Fig. 7(e) shows the final detection results by proposed detection method. Red bounding box represents detected region of traffic sign.

A correct detected sign is considered true positive if the corresponding bounding box overlaps with at least 50% of the area covered by the right traffic sign present in the image. The evaluation of the detection stage is performed based on precision–recall curve, where the recall and precision values are computed as follows:

$$recall = \frac{number\ \ of\ \ correctly\ \ detected\ \ signs}{number\ \ of\ \ true\ \ signs} \times 100 \tag{5}$$

$$precision = \frac{number\ \ of\ \ correctly\ \ detected\ \ signs}{number\ \ of\ \ detected\ \ signs} \times 100 \tag{6}$$

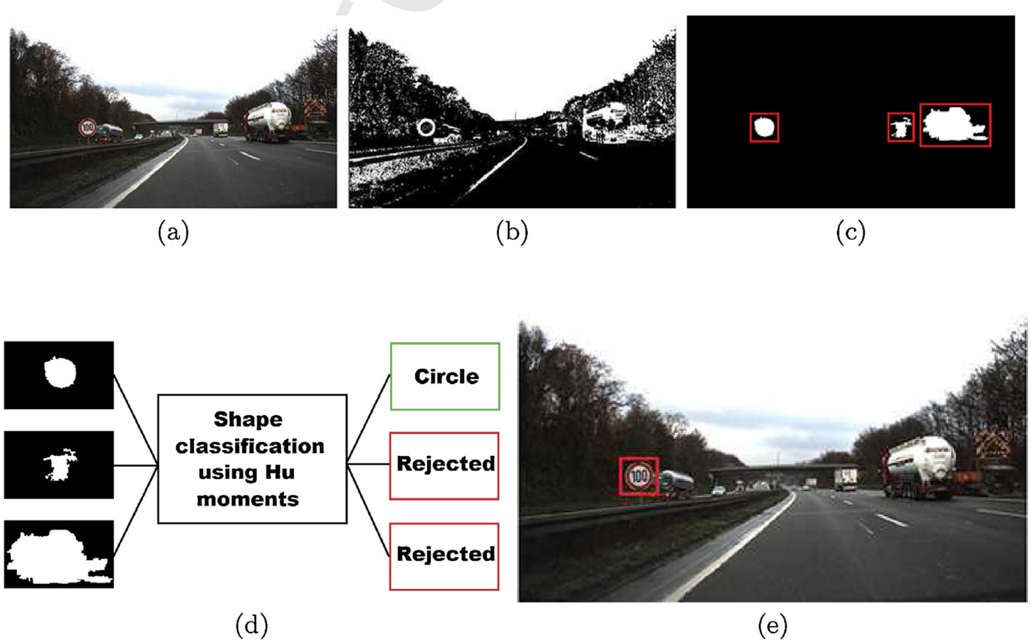

Fig. 7. An example of the results obtained by the detection stage. (a) Original image. (b) Segmentation results. (c) Segmentation results after taking account the aspect ratio and the size. (d) Shape classification results. And (e) sign detection results.

Please cite this article in press as: A. Ellahyani, et al., Traffic sign detection and recognition based on random forests, Appl. Soft Comput. J. (2016), http://dx.doi.org/10.1016/j.asoc.2015.12.041

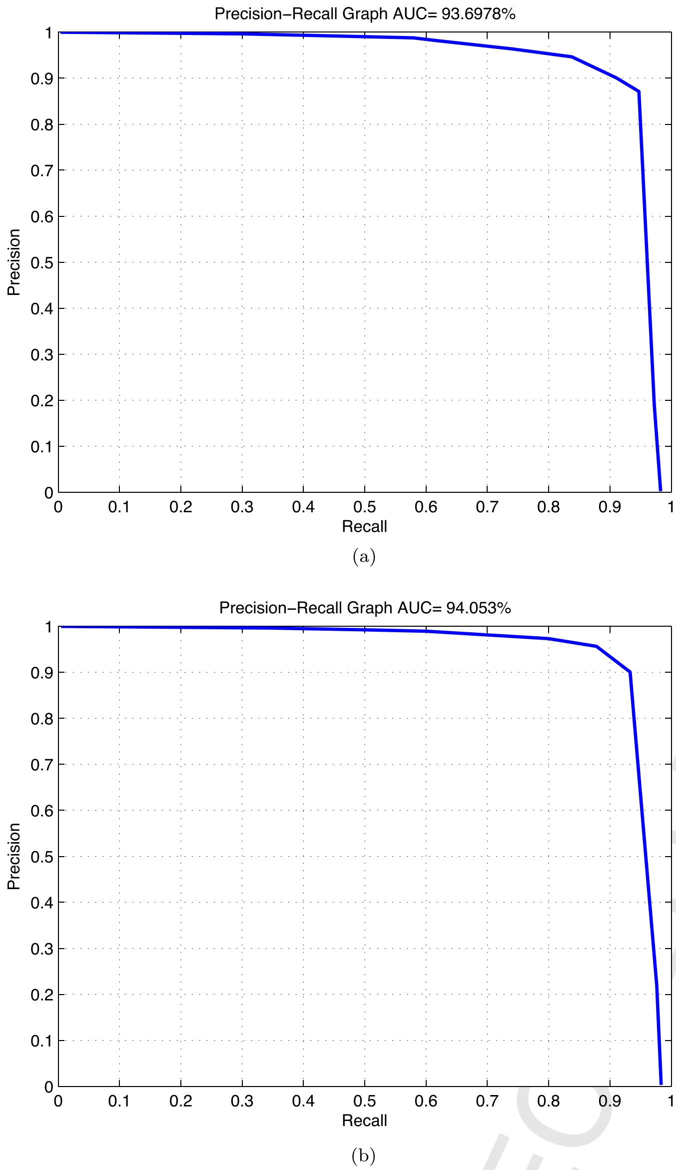

Fig. 8. Precision–recall curves of the proposed detection approach when applied to (a) GTSDB and (b) STS data sets.

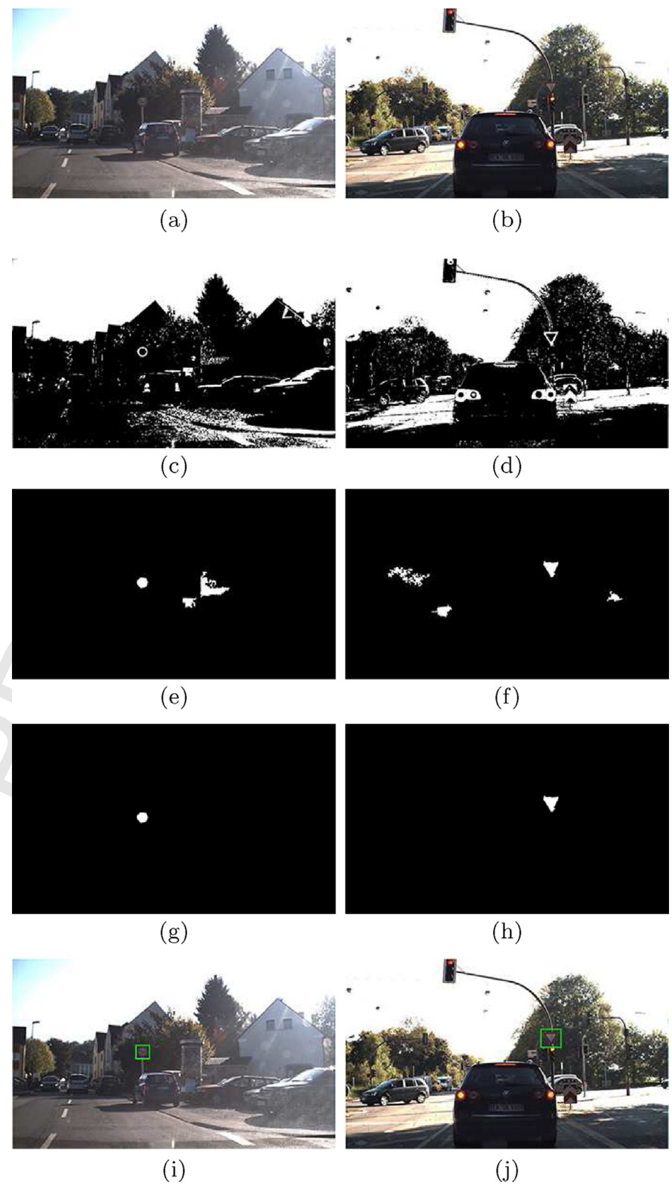

Fig. 9. Example of detection results. (a) and (b) Original images. (c) and (d) Segmentation results. (e) and (f) segmentation results after taking account the size and aspect ratio constraints. (g) and (h) shape classification results. (i) and (j) detection results.

Table 2
The best trade-off between the recall and precision values as well as the AUC obtained by the detection method on the GTSDB and STS data sets.

	Recall	Precision	AUC
GTSDB data set	91.07%	90.13%	93.69%
STS data set	93.27%	90.27%	94.05%

The precision–recall curves of the proposed method when applied to GTSDB and STS data sets are depicted in Fig. 8(a) and (b), respectively. The best trade-off between the recall and precision values as well as the Area Under Curve (AUC) of both data sets used are listed in Table 2. It can be seen that the method yields the best results with recall of 91.07% at a precision of 90.13% on the GTSDB and recall of 93.27% at a precision of 90.27% on the STS. The AUC of the two precision–recall curves are 93.69% and 94.05%, respectively.

More detection results are illustrated in Fig. 9. The first two of the same figure depicts the test images. The corresponding segmentation results without and with taking account the size and aspect ratio are illustrated in the second and third rows of Fig. 9, respectively. We can remark from Figs. 9(e) and (f) how the number of insignificant ROIs is reduced. The numbers of ROIs retained from Figs. 9(e) and (f) are 3 and 4, respectively. Among those ROIs only one ROI at each image is considered as a candidate sign (classified as a valid shape). The other ROIs were discarded and will not be processed in the recognition step. The ROIs classified as valid shapes are shown in Figs. 9(g) and (h). The classified ROI are depicted on the corresponding test images by the means of green bounding boxes in Figs 9(i) and (j).

4.3. Recognition

To evaluate the recognition stage, we used the training and the testing GTSRB data sets. A comparison between features and classifiers used in the system is performed. The proposed recognition approach is compared to the state-of-the-art works to assess its performances.

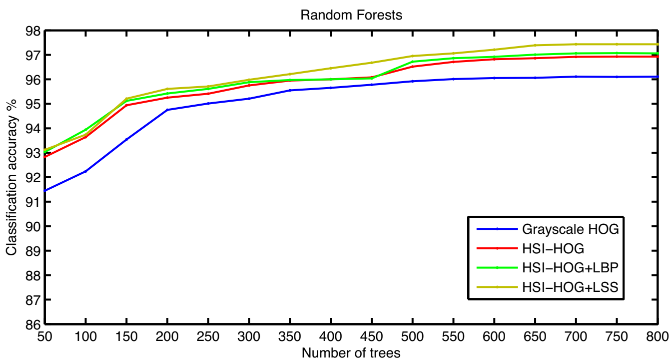

Fig. 10. The average classification accuracy of the random forest classifier with different features.

Table 3
Performance of each feature descriptor on GTSRB data set.

Feature	CCRs(%) of all road signs	CCRs(%) of each subset					
		(a)	(b)	(c)	(d)	(e)	(f)
HOG	96.11	95.61	98.32	88.15	98.05	94.08	98.14
HSI–HOG	96.73	96.09	99.18	88.24	99.02	94.39	99.08
HSI–HOG+LBP	97.06	96.28	99.31	89.23	99.31	95.21	99.03
HSI–HOG+LSS	97.43	96.58	99.15	89.47	99.33	96.37	99.19

Table 4
The CCR and the average running time of the classifiers used in this work.

Feature	CCRs(%) of all subsets		Run time (ms/frame)	
	Random forest	SVM	Random forest	SVM
HOG	96.11	95.93	21.24	46.18
HSI-HOG	96.73	96.29	28.51	51.96
HSI-HOG+LBP	97.06	96.72	29.75	53.87
HSI-HOG+LSS	97.43	96.91	28.93	53.12

The parameters of the classifiers and features used in the system are obtained from cross-validation experiments achieved on the training data set, which we divide into a training and validation subsets. Training and testing the classifiers on the two subsets, using different parameters settings, allows us to select the classifiers and features parameters that maximize the validation accuracy. The classifiers are trained again, using the selected parameters, on the GTSRB training data set.

In this work, we used a Random Forest with 100 variables and 750 trees. The classification accuracy increases with the number of trees and becomes constant as this number reaches the value 750 (see Fig. 10). Therefore, 750 is chosen for the number of trees to be used in the Random Forest classifier.

Table 3 shows Correct Classification Rates (CCRs) provided by Random forests classifier by applying the proposed recognition method to the GTSRB data set, which is composed of six subsets as illustrated in Fig. 6. The CCR is computed independently for each subset and for the whole road signs of the GTSRB data set. Different features have been used such as HOG, HSI–HOG, HSI–HOG+LBP and HSI–HOG+LSS. We remark from the table that the color cue improves the classification performance, e.g. HSI–HOG features give better results than the grayscale HOG in term of CCR. The recognition results improved more when the HSI–HOG is combined with LBP or LSS features. The HSI–HOG provides a higher CCR when combined with LSS rather than LBP.

Table 4 gives a comparison between SVM with radial basis function (RBF) kernel, $C = 7$ and $G = 0.09$ and Random Forest 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. That's why, we have adopted in the proposed recognition method the Random Forest classifier together with the HSI–HOG+LSS features.

Table 5
Comparison between the proposed recognition method and other published methods using the GTSRB data set.

Method	CCR (%)
Committee of CNNs [36]	99.46
Multi-scale CNNs [37]	98.31
The proposed Method	**97.43**
Random forests [25]	96.14
LDA on HOG2 [38]	95.68

A comparison against some state-of-the-art works is presented to assess the performances of the new method. Such a comparison has been done with the methods presented in [36,37,25,38] when those methods have been tested on the GTSRB data set as we have done for the new method. The classification accuracies obtained by the proposed method as well as the earlier proposed methods are depicted in Table 5.

The methods presented in [36,37] use convolutional neural networks to perform the recognition. The Random Forest together with HOG features have been used in [25]. The last approach included in the table performs the recognition based on Linear Discriminant Analysis (LDA) with HOG features. We remark from the table that the committee of CNNs and Multi-scale CNNs-based approaches provides best accuracies compared to our approach. The accuracies for the two methods as well as the new one are 99.46%, 98.31% and 97.43%, respectively. However, a high computing cost is paid for both methods to train the data set which makes them computationally very demanding. The proposed method is more accurate compared to the method in [25] and the LDA-based method for which the classification accuracies are 96.14% and 95.68%, respectively. The computational cost of the three methods is much lower than the CNNs-based methods.

There are many differences between the method presented in this work and the one presented in [25]. The authors in [25] used color enhancement in the RGB color space to detect signs. However, in our work, we used color thresholding in the HSI color space in order to overcome illumination problems. Moreover, we used invariant geometric moments to classify ROIs provided by the color segmentation process. This allowed us to eliminate blobs with insignificant shapes. In the recognition stage, the authors in [25] used a grayscale-based HOG features with a Random Forests of 500 trees. However, in our work, we computed HOG features using the HSI color space. Then, we combined these features with LSS features to get our final descriptor which was feed to Random Forests of 750 trees. Using both color and texture based features in our recognition module allowed us to obtain better results in the term of classification accuracy compared to the one in [25].

4.4. Entire system

The last evaluation setup was to test the performance of the entire detection and recognition system.

The GTSDB data set will be used to test the performances of the whole system (detection and recognition). The GTSRB data set is composed from training and testing data sets. The training data set includes only images for correct road signs. In our algorithm, each shape detected should be classified as road sign if it is among the road signs included in the training database. if not, it will be classified as no road sign. To be useful for our method, some false positives should be added to the training database. Some images without road signs have been collected from different road scenes and added to training database. The proposed classifier has been

Please cite this article in press as: A. Ellahyani, et al., Traffic sign detection and recognition based on random forests, Appl. Soft Comput. J. (2016), http://dx.doi.org/10.1016/j.asoc.2015.12.041

trained using the new training database and has been tested on the GTSDB database.

Fig. 11 shows the precision–recall curve obtained by the new method for traffic sign detection and recognition. It achieves 94.21% AUC on an average run time of 8-10 frames per second. We remark that the AUC has been improved compared to the one we have obtained the detection stage, which is 93.69%. This can be justified by the reduction of the false positives in the recognition stage.

Figs. 12, 13 and 14 illustrate examples of recognition results when the proposed approach applied to images of various traffic environments. In Fig. 12, the traffic signs contained in the images have been successfully detected and recognized. In Fig. 13(a) and (b), the road signs were too far to be detected. After the color segmentation, they were discarded because they were not meeting the size constraint. In Fig. 13(c) and (d), the signs color in the images was changed due to the shadows. Consequently, the ROIs corresponding to the signs were not extracted by the segmentation method. In Fig. 14, the traffic signs contained in the images

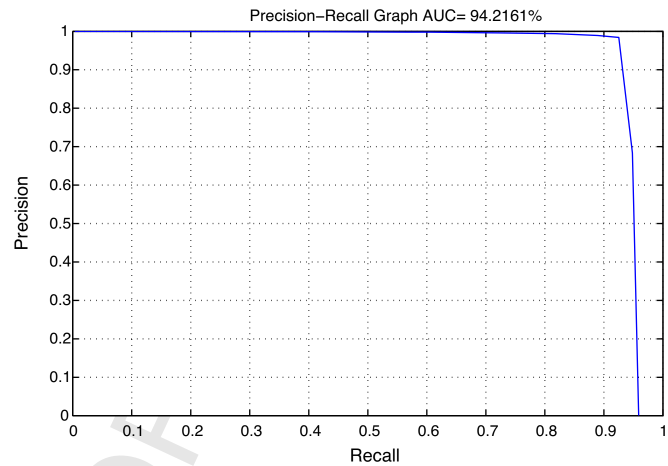

Fig. 11. Precision–recall curves of the proposed detection and recognition method.

Fig. 12. Detection and recognition results.

Fig. 13. Examples of recognition with misdetection.

Please cite this article in press as: A. Ellahyani, et al., Traffic sign detection and recognition based on random forests, Appl. Soft Comput. J. (2016), http://dx.doi.org/10.1016/j.asoc.2015.12.041

Fig. 14. Examples of recognition with confused classification.

have been successfully detected. However, the system could not recognize them due to the motion blur in the signs.

5. Conclusion and perspectives

In this paper, a three stages system for real-time Traffic Sign Detection and Recognition has been presented. The first stage segments the images into ROIs based on color information. Only significant ROIs will be considered referred to their size and aspect ratio constraints. In the second stage, the circular, rectangular and triangular shapes are detected using invariant geometric moments. In the recognition stage, we combine the HOG features computed in the HSI color space with LSS features to form a new descriptor. The Random Forest classifier is used with this descriptor to recognize the detected shapes. The entire system achieves 94.21% AUC on the GTSDB data set at a processing rate of 8–10 frames/s. In the future work, we are planning to use adaptive thresholding to overcome the color segmentation problems. On the other hand, 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. Moreover, the feature selection can be employed to accelerate the recognition phase by reducing the size of the descriptor vectors. Further, this could be combined with other classifiers, such as the Neural Networks.

References

[1] M.S. Prieto, A.R. Allen, Using self-organising maps in the detection and recognition of road signs, Image Vis. Comput. 27 (6) (2009) 673–683.
[2] A. De La Escalera, L.E. Moreno, M.A. Salichs, J.M. Armingol, Road traffic sign detection and classification, IEEE Trans. Ind. Electron. 44 (6) (1997) 848–859.
[3] M. Bénallal, J. Meunier, Real-time color segmentation of road signs, in: Canadian Conference on Electrical and Computer Engineering, IEEE CCECE 2003, Vol. 3, IEEE, 2003, pp. 1823–1826.
[4] A. Ruta, Y. Li, X. Liu, Real-time traffic sign recognition from video by class-specific discriminative features, Pattern Recogn. 43 (1) (2010) 416–430.
[5] S. Maldonado-Bascón, S. Lafuente-Arroyo, P. Gil-Jimenez, H. Gómez-Moreno, F. López-Ferreras, Road-sign detection and recognition based on support vector machines, IEEE Trans. Intell. Transp. Syst. 8 (2) (2007) 264–278.
[6] D.M. Gavrila, Multi-feature hierarchical template matching using distance transforms, in: Fourteenth International Conference on Pattern Recognition, Proceedings, vol. 1, IEEE, 1998, pp. 439–444.
[7] B. Alefs, G. Eschemann, H. Ramoser, C. Beleznai, Road sign detection from edge orientation histograms, in: Intelligent Vehicles Symposium, IEEE, IEEE, 2007, pp. 993–998.
[8] A. Ellahyani, M.E. Ansari, A three stages system for traffic sign detection and recognition, in: Third World Conference on Complex Systems (WCCS), IEEE, 2015, p. 115.
[9] J. Lillo-Castellano, I. Mora-Jiménez, C. Figuera-Pozuelo, J. Rojo-Álvarez, Traffic sign segmentation and classification using statistical learning methods, Neurocomputing 153 (2015) 286–299.
[10] S. Salti, A. Petrelli, F. Tombari, N. Fioraio, L. Di Stefano, Traffic sign detection via interest region extraction, Pattern Recogn. 48 (4) (2015) 1039–1049.
[11] C. Yao, F. Wu, H.-j. Chen, X.-l. Hao, Y. Shen, Traffic sign recognition using hog-svm and grid search, in: 12th International Conference on Signal Processing (ICSP), IEEE, 2014, pp. 962–965.
[12] J. Jin, K. Fu, C. Zhang, Traffic sign recognition with hinge loss trained convolutional neural networks, IEEE Transactions on Intelligent Transportation Systems 15 (5) (2014) 1991–2000.
[13] J. Miura, T. Kanda, Y. Shirai, An active vision system for real-time traffic sign recognition, in: Intelligent Transportation Systems, Proceedings, IEEE, IEEE, 2000, pp. 52–57.
[14] C.F. Paulo, P.L. Correia, Automatic detection and classification of traffic signs, in: Eighth International Workshop on Image Analysis for Multimedia Interactive Services, WIAMIS'07, IEEE, 2007, p. 11.
[15] W.-J. Kuo, C.-C. Lin, Two-stage road sign detection and recognition, in: IEEE International Conference on Multimedia and Expo, IEEE, 2007, pp. 1427–1430.
[16] S.M. Bascón, J.A. Rodríguez, S.L. Arroyo, A.F. Caballero, F. López-Ferreras, An optimization on pictogram identification for the road-sign recognition task using SVMS, Comput. Vis. Image Underst. 114 (3) (2010) 373–383.
[17] F. Larsson, M. Felsberg, Using Fourier descriptors and spatial models for traffic sign recognition, in: Image Analysis, Springer, 2011, pp. 238–249.
[18] F. Moutarde, A. Bargeton, A. Herbin, L. Chanussot, Robust on-vehicle real-time visual detection of American and European speed limit signs, with a modular traffic signs recognition system, in: Intelligent Vehicles Symposium, IEEE, IEEE, 2007, pp. 1122–1126.
[19] G. Loy, A. Zelinsky, Fast radial symmetry for detecting points of interest, IEEE Transactions on Pattern Analysis and Machine Intelligence 25 (8) (2003) 959–973.
[20] G. Loy, N. Barnes, Fast shape-based road sign detection for a driver assistance system, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, (IROS 2004), Proceedings, vol. 1, IEEE, 2004, pp. 70–75.
[21] K.H. Lim, K.P. Seng, L.M. Ang, Intra color-shape classification for traffic sign recognition, in: International Computer Symposium (ICS), IEEE, 2010, pp. 642–647.
[22] B. Soheilian, N. Paparoditis, B. Vallet, Detection and 3d reconstruction of traffic signs from multiple view color images, ISPRS J. Photogr. Remote Sens. 77 (2013) 1–20.
[23] H. Liu, D. Liu, J. Xin, Real-time recognition of road traffic sign in motion image based on genetic algorithm, in: International Conference on Machine Learning and Cybernetics, Proceedings, vol. 1, IEEE, 2002, pp. 83–86.
[24] M.-K. Hu, Visual pattern recognition by moment invariants, IRE Trans. Inf. Theory 8 (2) (1962) 179–187.
[25] F. Zaklouta, B. Stanciulescu, Real-time traffic-sign recognition using tree classifiers, IEEE Trans. Intell. Transp. Syst. 13 (4) (2012) 1507–1514.
[26] X. Qingsong, S. Juan, L. Tiantian, A detection and recognition method for prohibition traffic signs, in: International Conference on Image Analysis and Signal Processing (IASP), IEEE, 2010, pp. 583–586.
[27] L. Breiman, Random forests, Mach. Learn. 45 (1) (2001) 5–32.
[28] V.N. Vapnik, V. Vapnik, Statistical Learning Theory, Vol. 1, Wiley, New York, 1998.

Please cite this article in press as: A. Ellahyani, et al., Traffic sign detection and recognition based on random forests, Appl. Soft Comput. J. (2016), http://dx.doi.org/10.1016/j.asoc.2015.12.041

[29] N. Christiannini, J. Shawe-Taylor, Support Vector Machines and other Kernel-Based Learning Methods, 2000.

[30] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR, vol. 1, IEEE, 2005, pp. 886–893.

[31] T. Ojala, M. Pietikä inen, D. Harwood, A comparative study of texture measures with classification based on featured distributions, Pattern Recogn. 29 (1) (1996) 51–59.

[32] E. Shechtman, M. Irani, Matching local self-similarities across images and videos, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR'07, IEEE, 2007, pp. 1–8.

[33] H. Wang, M.M. Ullah, A. Klaser, I. Laptev, C. Schmid, Evaluation of local spatio-temporal features for action recognition, in: BMVC 2009-British Machine Vision Conference, BMVA Press, 2009, pp. 124–131.

[34] http://benchmark.ini.rub.de/.

[35] https://www.cvl.isy.liu.se/research/datasets/traffic-signs-dataset/.

[36] D. Ciresan, U. Meier, J. Masci, J. Schmidhuber, A committee of neural networks for traffic sign classification, in: The 2011 International Joint Conference on Neural Networks (IJCNN), IEEE, 2011, pp. 1918–1921.

[37] P. Sermanet, Y. LeCun, Traffic sign recognition with multi-scale convolutional networks, in: The 2011 International Joint Conference on Neural Networks (IJCNN), IEEE, 2011, pp. 2809–2813.

[38] J. Stallkamp, M. Schlipsing, J. Salmen, C. Igel, The German traffic sign recognition benchmark: a multi-class classification competition, in: The 2011 International Joint Conference on Neural Networks (IJCNN), IEEE, 2011, pp. 1453–1460.

Please cite this article in press as: A. Ellahyani, et al., Traffic sign detection and recognition based on random forests, Appl. Soft Comput. J. (2016), http://dx.doi.org/10.1016/j.asoc.2015.12.041