Fast Traffic Sign Recognition via High-Contrast Region Extraction and Extended Sparse Representation

Chunsheng Liu, Faliang Chang, Zhenxue Chen, and Dongmei Liu

***Abstract*—In this paper, we propose a high-performance traffic sign recognition (TSR) framework to rapidly detect and recognize multiclass traffic signs in high-resolution images. This framework includes three parts: a novel region-of-interest (ROI) extraction method called the *high-contrast region extraction* (HCRE), the *split-flow cascade tree detector* (SFC-tree detector), and a rapid occlusion-robust traffic sign classification method based on the *extended sparse representation classification* (ESRC). Unlike the color-thresholding or extreme region extraction methods used by previous ROI methods, the ROI extraction method of the HCRE is designed to extract ROI with high local contrast, which can keep a good balance of the detection rate and the extraction rate. The SFC-tree detector can detect a large number of different types of traffic signs in high-resolution images quickly. The traffic sign classification method based on the ESRC is designed to classify traffic signs with partial occlusion. Instead of solving the sparse representation problem using an overcomplete dictionary, the classification method based on the ESRC utilizes a content dictionary and an occlusion dictionary to sparsely represent traffic signs, which can largely reduce the dictionary size in the occlusion-robust dictionaries and achieve high accuracy. The experiments demonstrate the advantage of the proposed approach, and our TSR framework can rapidly detect and recognize multiclass traffic signs with high accuracy.**

***Index Terms*—Traffic sign recognition (TSR), traffic sign detection (TSD), region of interest extraction, split-flow cascade, sparse representation classification (SRC).**

I. Introduction

TRAFFIC sign detection and recognition based on computer vision and pattern recognition is being studied for several purposes, such as autonomous driving and assisted driving. Recognition of traffic signs allows a driver to be warned about inappropriate actions and potentially dangerous situations. Traditionally, traffic sign recognition (TSR) systems consist of two phases: detection and classification [1]; for some TSR systems, region of interest (ROI) extraction is an important step before detection.

Before traffic sign detection, ROI extraction is often the first step in an automatic traffic sign detection and recognition framework. The ROI extraction methods are utilized to find sign candidates or reduce the search space of the following traffic sign detection. The distinct color characteristic of traffic signs provides an import cue to extract regions of interest. The developed color-based segmentation methods are based on the RGB color space [3]–[5], the HSV color space [6]–[8], the HIS color space [9], [10] and various other color spaces [11], [12]. Generally, color-based segmentation relies on a thresholding of the input image in a particular color space [1]. Some color-based methods that do not rely on thresholding have been put forward, including Neural Network learning [13], a quadtree interest-region-finding algorithm [14] and a biologically inspired attention method [15]. Edges and shapes are very important characteristics of traffic signs. Greenhalgh and Mirmehdi [31] used an extreme region detection method called CMSERs to find extreme regions in RGB channels. The method based on CMSERs can successfully extract red and blue colors, but is limited in extracting other colors.

Traffic sign detection is the process of locating the precise position of traffic signs in an entire image or in the extracted regions of interest. Generally, the most popular feature is the location of edges, and this reflects on the most popular choice in the detection methods. Hough transforms have been used to detect circles, triangles and rectangles [16]–[18]. Hough transforms are computationally expensive. A derivative method of Hough called the radial symmetry detector was put forward as a fast detection method [19], [20]. This algorithm finds gradients with a magnitude above a certain threshold and votes for the most likely sign centers in an image based on symmetric edges. The combination of HOG features and the Support Vector Machine (SVM) has been used in sign detection [21], [22], in which the detection is treated as a SVM-based classification problem. Boosted cascade based methods [23], [24] have been used to detect different types of signs. These cascade-based TSD systems can reach a high detection rate in a short detection time and are limited in their generalization ability. In [2], a split-flow cascade structure with strong generalization ability was put forward to construct a cascaded detector to detect different German traffic signs.

After traffic sign detection, traffic sign recognition is performed to classify the detected traffic signs into correct classes.

Manuscript received August 31, 2014; revised December 11, 2014, March 16, 2015, and May 28, 2015; accepted July 19, 2015. This work was supported in part by the National Natural Science Foundation of China under Grants 61273277 and 61203261, by the Specialized Research Fund for the Doctoral Program of Higher Education under Grant 20130131110038, by the Project sponsored by SRF for ROCS, SEM under Grant 20101174, and by Shandong province Natural Science Foundation Committee under Grant ZR2011FM032. The Associate Editor for this paper was J. M. Alvarez.

The authors are with the School of Control Science and Engineering, Shandong University, Ji'nan 250061, China (e-mail: flchang@sdu.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2015.2459594

The common approaches for recognition are divided into two methodologies: one is multi-category classification, which recognizes multiclass traffic signs directly, and the other treats the recognition task as a binary-classification-based problem. The recognition methods based on PCA [25], Neural Networks (NN) [26]–[28] or Sparse Representation Classification (SRC) [29], [30] have been designed to directly recognize one traffic sign from different types of traffic signs. Recently, several NN methods, such as Convolutional Neural Networks [26], Multi-Scale Convolutional Neural Networks [27] and Committee of Convolutional Neural Networks [28], have been used in TSR; also, several SRC-based methods including the SRC [29] and the sparse-representation-based graph embedding (SRGE) [30], have been applied to recognize traffic signs, especially ones with partial occlusion. Alternately, the recognition methods based on SVM [31], [32] or tree-based classification methods [33]–[35] treat the recognition task as a binary-classification-based problem. For example, the recognition methods based on SVM treat multi-classification as a one-vs-one or one-vs-rest (OvO or OvR) strategy, whereas the tree-based methods classify traffic signs using a binary classification tree.

A. Motivation and Contribution

The goal of this study is to establish a complete TSR system to detect and classify traffic signs in high-resolution images quickly. Based on the SFC-tree traffic sign detector described in [2], different traffic signs are able to be detected quickly in high-resolution images; yet, when complex recognition methods are utilized to classify traffic signs, the process of detection and recognition may be time-consuming. Hence, we need to find an effective ROI extraction method, and a fast and accurate recognition method.

The most popular previous ROI extraction methods [3]–[12] rely on thresholding of the input image in a particular color space; these methods are not robust to color variance in the input images. The other ROI extraction methods that do not rely on thresholding [13]–[15] or rely on extreme region detection [31] are often complex and time-consuming. To address these problems, a novel ROI extraction method, called the *High Contrast Region Extraction* (HCRE), is proposed. Motivated by the cascaded detection methods [24], [38], the HCRE utilizes integral image and rectangle features to quickly extract ROI regions; unlike the features used in cascaded detection methods [24], [38], the features used by the HCRE are not position-fixed in the detection window. The voting of neighboring features decides whether the corresponding region is a ROI region. Taking advantage of the observation that different types of traffic signs have relative high contrast in local regions, the HCRE can effectively remove non-interesting regions with small local contrast, such as sky, road and some buildings, and can boost the detection speed of the SFC-tree detector [2] from 5 frames per second to more than 10 frames per second in our experiments.

After the process of ROI extraction and traffic sign detection, multiclass traffic signs are detected and roughly classified into various categories. We design a verification method based on the OvR SVM classification method [36] to ensure the correct classification of the SFC-tree detector. This verification method is more timesaving than directly classifying these detected categories. After the verification process, the categories that contain different types of traffic signs have deeper classification performed.

For the traffic sign classification task, many TSR methods have been developed, such as the SVM-based methods [31], [32], the NN-based methods [26]–[28] and the tree-based methods [33]–[35]; yet, the SVM-based and tree-based methods are not robust at classifying signs with partial occlusions or isolated samples, and the NN-based methods are often too slow to be applied in real applications. Recently, some SRC-based TSR methods [29], [30] have been developed and employed for traffic sign classification. By introducing an identity occlusion dictionary, the SRC-based methods have robust recognition results against partial occlusion and isolated samples. The success of the existing SRC-based methods [29], [30] inspires us to design an occlusion-robust SRC-based TSR method; the existing problem is that the accuracy of the SRC-based methods [29], [30] largely relies on an over-complete dictionary, which often results in high computational complexity. We design a rapid occlusion-robust traffic sign recognition method based on the *Extended Sparse Representation Classification* (ESRC) [37] to improve the previous SRC-based methods in recognition time. The proposed ESRC-based method has two dictionaries: the content dictionary and the occlusion dictionary. The content dictionary can represent the inter-class differences, whereas the occlusion dictionary is shared by different objects and is utilized to represent the common occlusions of different signs. The introduction of the content dictionary and the common occlusion dictionary makes the sparse representation compact.

All of these methods are combined into our efficient multiclass TSR system, which can detect and recognize different types of traffic signs in high resolution images at approximately 8 to 12 frames per second on the GTSDB dataset [40]. The framework can reach accuracy as high as that of the-state-of-the-art methods, and the processing time is performed quickly in high-resolution images.

The remainder of this paper is organized as follows: In Section II, the ROI extraction method of the HCRE is introduced. Section III shows the TSD method based on the SFC-tree detector and the verification method. Section IV shows the occlusion-robust classification method based on the ESRC. Section V shows related experimental results, and Section VI presents the conclusion.

II. Region of Interest Extraction Method

Region of interest (ROI) extraction or non-interesting region removal is usually the first step before traffic sign detection and recognition. In this paper, we propose a new ROI extraction method called the *High Contrast Region Extraction* (HCRE). The purpose of this method is to reject non-interesting regions with low local contrast to save detection time. The HCRE ROI extraction method can combine with the SFC-tree detection method described in Section III into a very fast traffic sign detection method. The boosted cascade based detection methods [2], [23], [24] waste most of their computing time in the

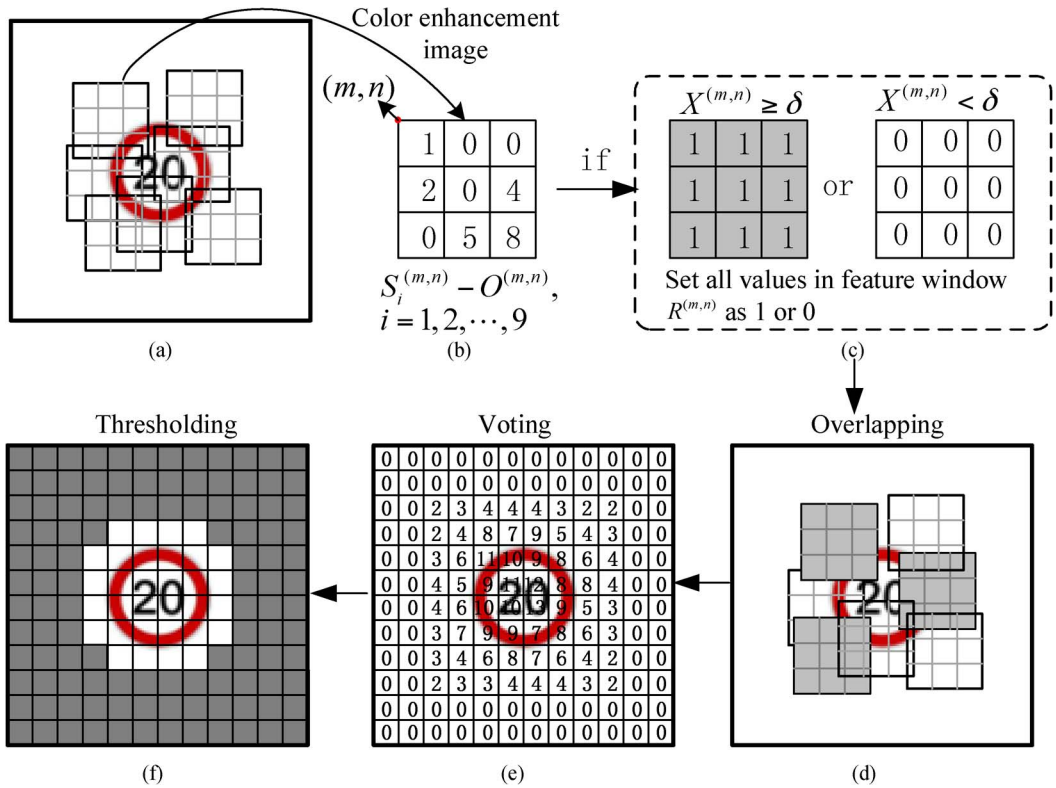

Fig. 1. The ROI extraction process of the HCRE method. The process of the ROI extraction method is shown from (a) to (f). (a) is to scan the image with a large amount of scanning feature windows at the step s; (b) is the normalization of every feature window; (c) is the thresholding of every feature window; (d) is to show the overlapping of the feature windows with value 1 or 0; (e) is the voting results of the overlapping feature windows, called the voting image; (f) is the thresholding of the voting image, called the mask image, which is the ROI extraction result.

first several stages of the cascaded detector [38]; the proposed HCRE can largely solve this problem. The HCRE can reject approximately 83.10% of the non-interesting regions on our test datasets and can effectively boost the detection speed of the SFC-tree detector [2] from 5 frames per second to more than 10 frames per second in our experiments.

Though the gray-level image or one channel in a color space can be used in this method, the normalized RGB space can be used to reach much better ROI extraction results than directly using one channel. After color normalization, the traffic signs with red and blue color are enhanced, and the yellow traffic signs still have high contrast and can be extracted. The color enhancement image is denoted as P:

$$P = \frac{\max(R, B)}{R + G + B} \tag{1}$$

where, R, G, and B are the red, green and blue channels respectively in RGB space.

Because the feature used in the HCRE is rectangle feature, the integral image need to be calculated. The integral image at location (x, y) contains the sum of the pixels above and to the left of (x, y) [38]:

$$I(x, y) = \sum_{x' \le x, y' \le y} P(x', y') \tag{2}$$

where $I(x, y)$ is the integral image.

The smallest window of the objects to be detected is set as w-by-h pixel. In these experiments, the smallest detected window is 24-by-24 pixel. A large amount of w-by-h pixel rectangular windows are applied to scan the image at step s. The scanning rectangular window is called *feature window* and denoted as $R^{(m,n)}$, where (m, n) are the top left coordinates of $R^{(m,n)}$. The coordinates of all pixels in $R^{(m,n)}$ belong to the set $E^{(m,n)}$. Each pixel in the image can be scanned by different feature windows, and be voted by its involved feature windows. We call the image after voting, the *voting image* V, and initialize it as $V(x, y) = 0$. The framework of the HCRE is shown in Fig. 1, and the process of the ROI extraction is shown in Fig. 2.

As shown in Fig. 1, the w-by-h rectangular feature window is divided into several small subwindows with the same sizes. For traffic signs, there are always different pixel distributions in these different small subwindows. The number of subwindows in each feature window is denoted as l, and the subwindow size is denoted as w'-by-h' pixel. In these experiments, like the MN-LBP feature in [2], there are 3-by-3 subwindows in each feature window and $l = 9$; this 3-by-3 feature window can successfully represent the distribution of multiclasss traffic signs in local regions. The sum of the pixel values in the ith subwindow of $R^{(m,n)}$ is denoted as $S_i^{(m,n)}$.

The average value $O^{(m,n)}$ of all subwindows in $R^{(m,n)}$ is calculated. We get the normalized value $T_i^{(m,n)}$ by subtracting the average value $O^{(m,n)}$ from the $S_i^{(m,n)}$, giving:

$$T_i^{(m,n)} = S_i^{(m,n)} - O^{(m,n)}, \ (i = 1, 2, \ldots, l). \tag{3}$$

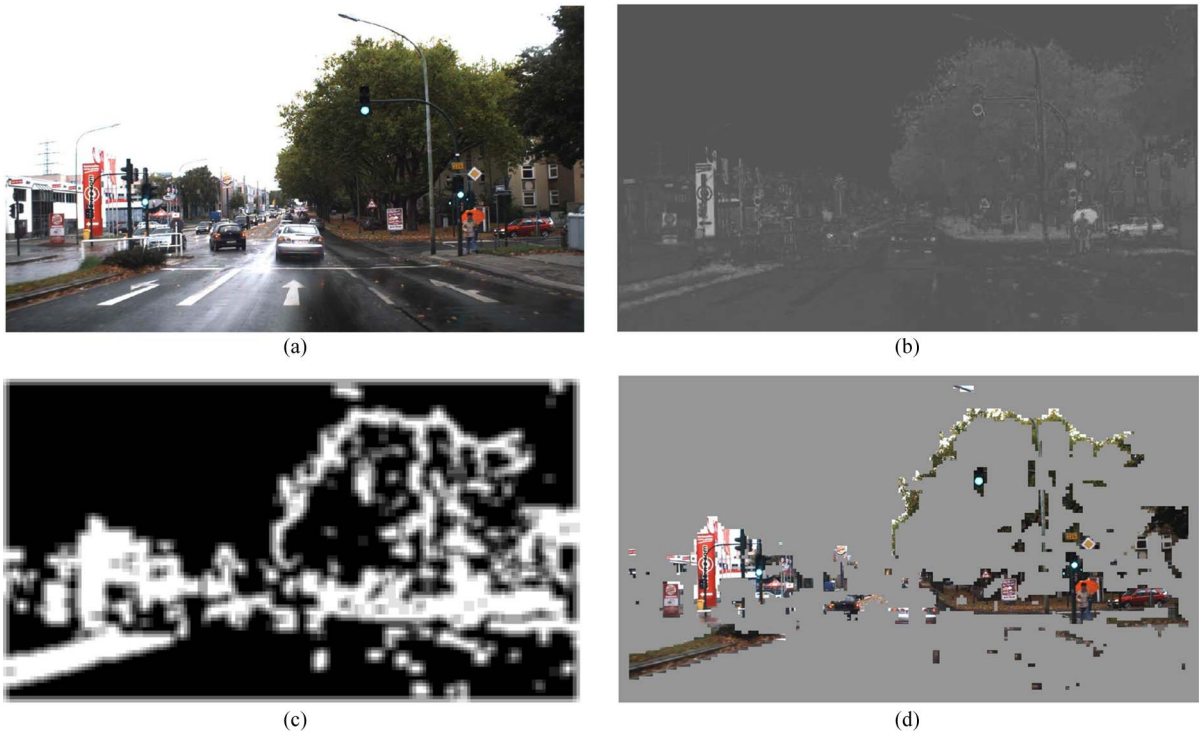

Fig. 2. The process of the ROI extraction. (a) is the input image. (b) is the color enhancement image. (c) is the voting image. (d) is to show the extraction regions using the mask; the gray region in (d) is the mask.

Then, the maximum value of the $T_i^{(m,n)}$ in $R^{(m,n)}$ is calculated, and divided by the pixel number in each subwindow to achieve that the maximum value is irrelevant to the feature window size. The calculated value is denoted as $X^{(m,n)}$,

$$X^{(m,n)} = \frac{\max\limits_{1\le i\le l}\left\{\left|T_i^{(m,n)}\right|\right\}}{(w'\cdot h')}. \tag{4}$$

When the maximum value of one feature window is large, the contrast in this feature window is large and this feature window can be a candidate interesting region. Only one feature window cannot determine whether the rectangular window in this feature window is a region of interest, because some signs are symmetrical and have little difference among their subwindows, or the distribution of the sign is similar to that of the environment around the sign. To avoid misjudgment of this case, we utilize the voting decision of the feature windows around the sign. Each pixel will be scanned by different rectangular feature windows. In the voting image $V(x,y)$, different pixels have different values, which reflect the possibility to be judged as candidate interesting regions. We set the threshold of $X^{(m,n)}$ as δ. In the scanning process, the voting image $V(x,y)$ is updated according to the thresholding result of $X^{(m,n)}$:

$$V(x,y) = \begin{cases} V(x,y)+1, & \text{if } X^{(m,n)} \ge \delta \\ V(x,y), & \text{if } X^{(m,n)} < \delta \end{cases} \quad \forall\,(x,y) \in E^{(m,n)}. \tag{5}$$

After the scanning process is over, the voting image $V(x,y)$ can be gotten and can be used to extract ROI. The threshold of the voting value is set as η. When the value of one pixel in $V(x,y)$ is greater than η, this pixel is classified as an interesting region. We can use this thresholding process to get the mask image and the mask image is denoted as $K(x,y)$,

$$K(x,y) = \begin{cases} 1, & \text{if } V(x,y) \ge \eta \\ 0, & \text{if } V(x,y) < \eta \end{cases} \tag{6}$$

where, $K(x,y)=1$ denotes that the pixel in (x,y) belongs to interesting regions, and $K(x,y)=0$ denotes that the pixel in (x,y) belongs to non-interesting regions.

After the ROI extraction process, we obtain the mask image K containing regions of interest; yet, parts of some sign regions may be removed as non-interesting regions, because the sign image is much larger than the scanning feature windows or the distribution of the sign is similar to that of the environment around the sign. To avoid this problem, we design the rule of traffic sign detection according to the mask image K. We denote the scanning subwindow of the detector as C, and $S(C)$ is the sum of the pixel values of the scanning subwindow C in the corresponding position of the mask image K. We define $\varphi(C)$ to determine whether the scanning detector C need to be detected,

$$\varphi(C) = \begin{cases} 0, & \text{if } S(C) < \xi \\ 1, & \text{if } S(C) \ge \xi \end{cases} \tag{7}$$

where, ξ is the threshold of $S(C)$. ξ is set as $w\cdot h/2$, because in our experiments at least 1/2 regions of different traffic signs are not covered in the mask image.

Then, the scanning detector can scan the image according to $\varphi(C)$. When $\varphi(C) = 0$, this subwindow does not need to be scanned, whereas when $\varphi(C) = 1$, this subwindow needs to be scanned and detected. This process can preserve nearly all traffic signs after ROI extraction, and largely reduce the scanning time of the traffic sign detector.

In this method, there are three parameters needed to be fixed: the scanning step s, the threshold of maximum value in the normalized feature window δ, and the threshold of the voting image η. We take the GTSDB database as an example to show the process of getting these parameters.

Step 1. We firstly train multiclass traffic signs to get the value of δ. Different traffic signs and background images are collected as the traffic sign set and the background set. Using the accuracy $c = 99.0\%$, we can get a value of δ that can ensure that more than 99.0% feature windows in the traffic sign set are in the range $X^{(m,n)} \geq \delta$.

Step 2. With chosen δ, the images in the traffic sign set are processed with different s and η to get the candidate parameters. In these experiments, w equals h, and the scanning step s is set as: $s \in \{s|0 \leq s \leq w\}$. Using different s values, we can get the corresponding η that can ensure that 99.0% of the traffic signs in the traffic sign set are classified as interesting regions. All the values s and η are put into the set $P = \{(s_1, \eta_1), (s_2, \eta_2), \ldots\}$.

Step 3. Using the parameters in set P to process the training images in GTSDB, the extraction rate a can be gotten for different parameters. The extraction rate a is the ratio of the number of the scanning detector subwindows according to $\varphi(C)$ in formula (7) to the number of the original scanning detector subwindows. When the extraction rate a is gotten, the expected processing time T, including the ROI extraction time and detection time, can be estimated using the following formula,

$$T = t_e + t_d \cdot a \tag{8}$$

where, t_e is the ROI extraction time, t_d is the detection time without using ROI extraction. Then, the minimum value of T can be easily gotten; in this case, the efficiency of the ROI extraction is judged to be the highest and the suitable parameters s, η can be gotten. After the three steps, the three parameters s, δ, and η can be fixed.

III. Traffic Sign Detection and Verification

A. Traffic Sign Detection Using the SFC-Tree Detector

After the segmentation of ROI, multiclass traffic sign detection based on the SFC-tree detector [2] is applied to detect different types of traffic signs, which are shown in Fig. 3. The SFC-tree detector is a fast and accurate multiclass TSD system, which can rapidly detect a wide variety of traffic signs without using any color information. There are three main parts of the multiclass TSD framework [2], including the *multi-block normalization LBP* (MN-LBP) and tilted multi-block normalization LBP (TMN-LBP) features, the *Split-Flow Cascade* and the *Common-Finder AdaBoost* algorithm (CF.AdaBoost).

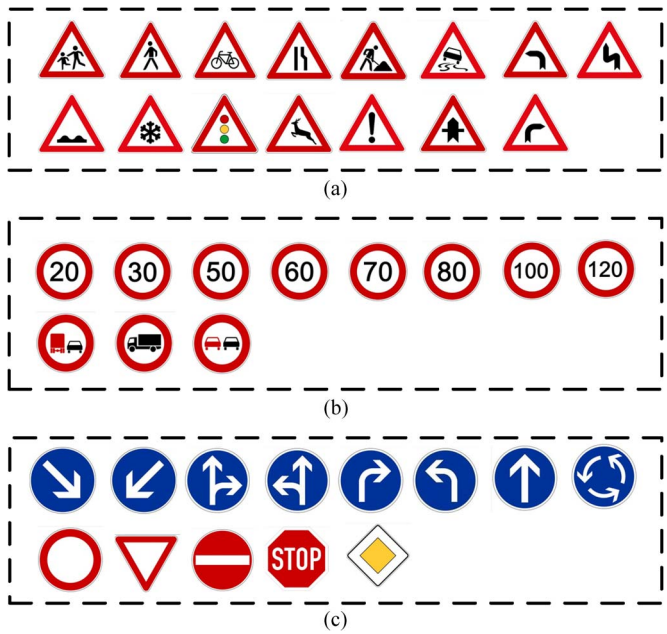

Fig. 3. All traffic signs. All triangular danger signs in (a) form a training set; All circular prohibitory signs in (b) can form a training set; The 13 different types of traffic signs in (c) form 13 independent training sets.

The MN-LBP features are proposed to express different types of traffic signs. The basic approach of MN-LBP is that the pixel sum values of eight rectangles around the center are compared with an average value to obtain a binary sequence. The proposed MN-LBP features are able to efficiently express very different patterns of multiclass traffic signs. In this work, we introduce threshold to MN-LBP features to improve the generalization ability of MN-LBP features, which is shown in Fig. 4(a).

$$t = (s(S_1 - A), \ldots, s(S_8 - A)) \tag{9}$$

where, t is a vector of MN-LBP with eight values, A is the average value:

$$A = \sum_{i=0}^{8} S_i / 9 \tag{10}$$

$$s(x) = \begin{cases} 1, & \text{if } x/K \geq \lambda \\ 0, & \text{if } x/K < \lambda \end{cases} \tag{11}$$

where, λ is the threshold of the MN-LBP features, K is the pixel number in the subwindow of MN-LBP, x/K is to eliminate the differences caused by scaling. The introducing of threshold can bring more candidate training features and improve the generalization ability of the features.

The SFC-tree is a coarse-to-fine pyramid structure, as shown in Fig. 4(b). The branching nodes in the tree are constructed from several stages. Each branching node is designed to detect various objects involved in that node while rejecting non-object subwindows as much as possible. The SFC-tree is able to reject non-objects in every stage of the branching nodes and leaf nodes, and the decision in each branching node is non-exclusive. These characteristics improve the detection rate and

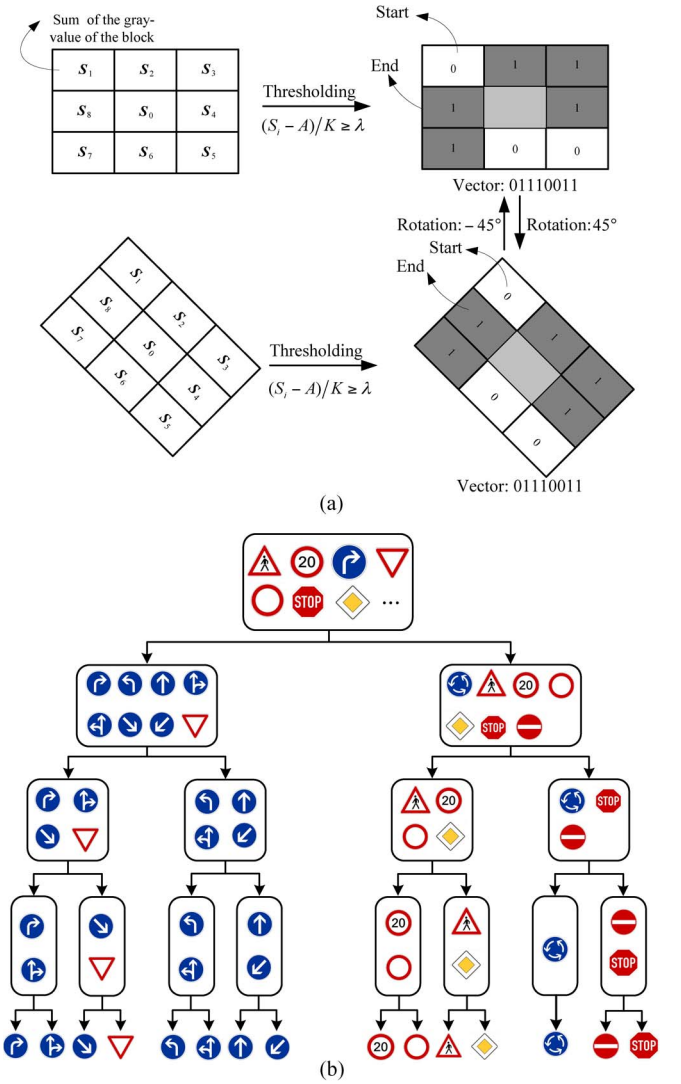

Fig. 4. The features and structure used in detection. (a) shows the MN-LBP features with threshold. (b) is the established Split-Flow Cascade structure for 39 different types of traffic signs.

detection time of the SFC-tree detector. The CF.AdaBoost is used to find common features of different training sets to build an SFC-tree, which is a learning method based on the classical AdaBoost algorithm [38]. All of these methods are combined together into this efficient multiclass TSD system. The detailed description of these methods can be found in [2].

In the design process of an SFC-tree for multiclass traffic signs, it is found that although different types of traffic signs appear different to each other, some of them share very similar appearances, such as the triangular danger category and the circular prohibitory category. Thus, before training, the 39 training sets are manually classified into 15 training sets to reduce the complexity of this TSD system. The classification result is shown in Fig. 3, in which (a) or (b) is treated as one training object, and (c) contains 13 different training objects. The rules of manual classification are that members in each large category must have a very similar appearance and that each category can be used to train a successful detector.

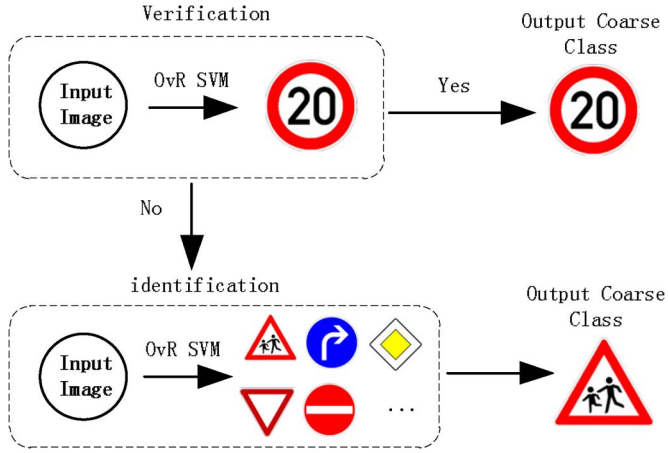

Fig. 5. The verification and identification process.

An SFC-tree is designed with one root node and 15 leaf nodes. The depth of this SFC-tree is five. With this established SFC-tree detector, an exhaustive search procedure is applied to detect traffic signs in an image. The SFC-tree detector is scanned across the image at multiple scales and locations.

In the nodes of the first and second levels of the SFC-tree detector, the cascade is not very efficient. There are two main reasons: first, there are so many different types of traffic signs; second, the features used in cascade are position-fixed in the detection window, which makes the features shared by different types of traffic signs not very efficient at rejecting non-sign subwindows. The ROI extraction method described in Section II can partly solve this problem. The scanning subwindow of the detector is denoted as C; in the scanning process, the $\varphi(C)$ value in formula (7) decides whether the subwindow needs to be scanned or not. If the $\varphi(C)$ value is nonzero, the detector will detect this subwindow; in other cases, this region will be rejected as a non-sign subwindow.

After the ROI extraction, the number of the detector subwindows has been largely reduced. After the detection of the SFC-tree detector, 39 different traffic signs in the GTSDB database [40] can be detected. The detection is robust to color changes, blurred edges and partial occlusions. The detailed verification experiments of the SFC-tree detector can be found in [2].

B. *Verification of Traffic Signs*

After the process of ROI extraction and traffic sign detection, multiclass traffic signs have been detected and roughly classified into various categories. These classification results of the SFC-tree are not absolutely reliable, and the rough classification accuracy is approximately 86.20% on the GTSDB database. This high accuracy means that instead of directly classifying the detected signs, we can verify the classification correctness of the detected signs. If the verification result of one sign is correct, this sign has been correctly recognized, whereas, in the other cases, this sign needs to be classified. This verification process is shown in Fig. 5. The verification of different categories uses much less time than directly classifying these categories.

The goal of verification is to build a model of a sign identity and then test a sign sample against this identity. The one-vs-rest SVM (OvR SVM) classification method [36] provides an effective tool for this verification problem, and the HOG features [43] are used in the SVM classification. Because the signs in the leaf nodes that need to be verified are very different from each other, the OvR SVM classification method can verify them at a high accuracy, reaching nearly 100% on the GTSDB database.

If the result of verification is correct, then the sign is in the correct category, whereas, if the result is in error, the sign then needs to be identified using the OvR SVM classification method. The computational complexity of the verification method in our experiments is much less than directly classifying these signs. Hence, this verification process is very time-saving and effective.

IV. Recognition of Traffic Signs

After the detection and verification process, traffic signs have been detected and classified into large categories. In this section, we introduce an occlusion-robust classification method based on the *Extended Sparse Representation Classification* (ESRC) [37] approach to classify traffic signs into finer classes. The ESRC was first introduced in [37] for undersampled face recognition. For the traffic sign recognition problem, some SRC-based TSR methods [29], [30] have been developed and employed for traffic sign classification. By introducing an identity occlusion dictionary, the SRC-based methods have robust recognition results against partial occlusion or isolated samples. However, the accuracy of existing SRC-based methods largely relies on an over-complete dictionary, which often results in high computational complexity. By analysis of the characteristics of multiclass traffic signs, we have developed a fast and accurate TSR method based on the ESRC. The proposed ESRC-based method treats every traffic sign as two separate parts: the content part and the occlusion part. Samples from these two parts form two dictionaries: the content dictionary and the occlusion dictionary. The content dictionary can represent the inter-class differences, whereas the occlusion dictionary is shared by different objects and is utilized to represent the common occlusions of different signs through linear combination. Compared with the previous SRC-based TSR methods, there are two main improvements.

Firstly, unlike classical SRC methods, the ESRC-based TSR method constructs two dictionaries, a content dictionary and an occlusion dictionary, shown in Fig. 6, to sparsely represent traffic signs. The content dictionary contains the contents of multiclass traffic signs. The occlusion dictionary contains occlusion parts, which are shared by multiclass traffic signs. Instead of using different types of signs with different occlusions in the classical occlusion-robust dictionary of SRC, the occlusion dictionary of ESRC can represent the different types of signs with different occlusions when combined with the content dictionary; this mechanism can largely reduce the atom numbers in the dictionaries of ESRC. For example, if some different types of signs with left occlusions need to be recognized, the dictionary of the classical SRC need contain all different types of traffic signs with similar left occlusions,

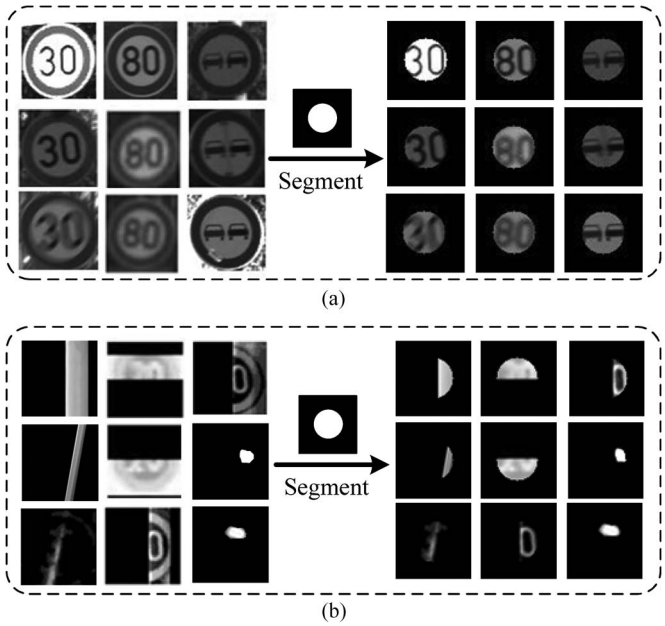

Fig. 6. The two dictionaries used in the ESRC. The atoms in the dictionaries are the values in the non-dark regions of the mask. The content dictionary contains the atoms without the frame and occlusions. The occlusion dictionary contains occlusion parts, and the values of the non-occlusion parts are 0. (a) The atoms in the content dictionary. (b) The atoms in the occlusion dictionary.

whereas, the dictionaries of the ESRC-based TSR method just need several left-occlusion samples in the common occlusion dictionary and several content samples in the content dictionary. Hence, the dictionary size of the ESRC-based TSR method is much smaller than the SRC-based TSR methods.

Secondly, apart from the introduction of these two dictionaries, the other improvement is utilizing the content without frame as the input vector. Unlike previous recognition methods that use the detected traffic sign as input vectors, we use the contents of traffic sign without frame as input vectors. This improvement can largely reduce the atom length in the dictionaries and reduce the noise effect from the frame. We do not need to design accurate segment method, and the segment parameters are based on the training results of the ESRC, which can be found in the experimental part. Using the segmented contents is more reliable than using the signs with frame.

The proposed ESRC recognition method classifies different types of traffic signs in the same leaf nodes, including the triangular danger signs and the circular prohibitory signs. As shown in Fig. 6, the content dictionary A contains content samples with content inside only, whereas the occlusion dictionary B contains the common occlusion structures that are shared by different types of traffic signs. The content regions of the prohibitory are circle. The content region of the danger sign is an equicrural triangle that is similar to the triangular shape of the danger signs. The classification processes of the classical SRC and the ESRC are shown in Fig. 7.

Firstly, the training samples of multiclass traffic signs are collected, aligned and resized into the same size. These samples are captured in different environments and have no occlusions. Then, we need to establish the training set for the occlusion dictionary B. The samples in the occlusion training set include

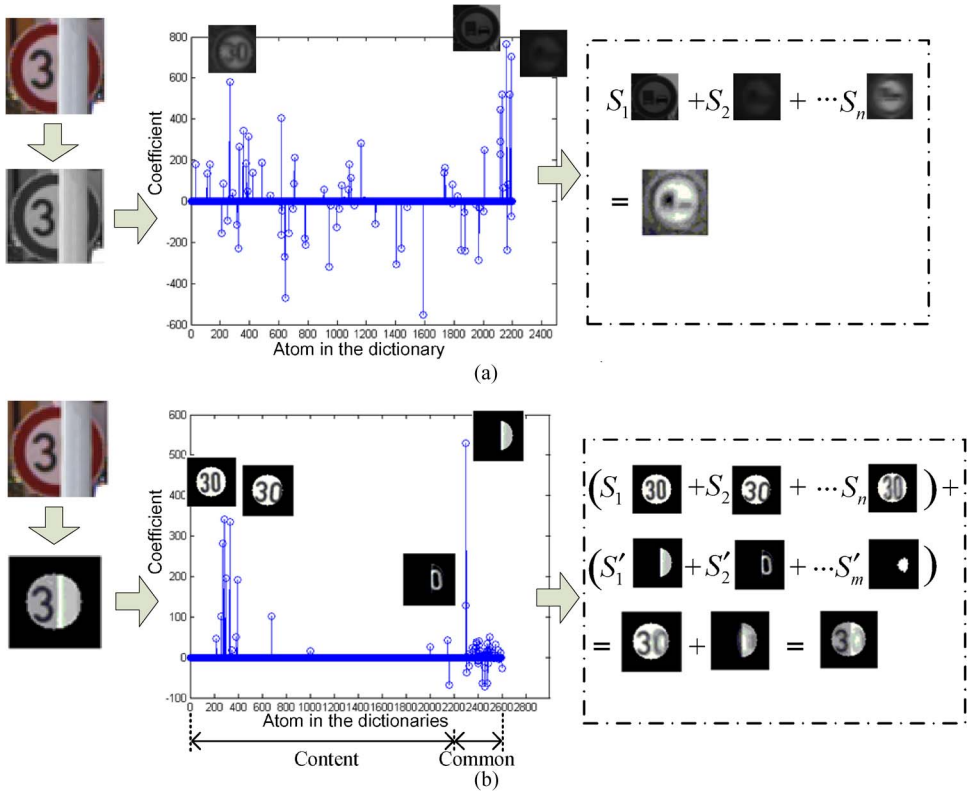

Fig. 7. The recognition process of the classical SRC-based and the ESRC-based TSR methods. Each column (atom) of the dictionary is a vector gotten from the corresponding training samples. For the classical SRC in (a), each atom is a one-dimensional vector containing the information of the whole traffic sign. For the ESRC in (b), each atom is a one-dimensional vector only containing the information in the content region. There are two dictionaries in the ESRC-based method; after sparsely representation, these two dictionaries can be used to reconstruct the input sign. (a) TSR based on classical SRC. (b) The proposed TSR method based on the Extended SRC.

three main parts: the common occlusions including trunk, telegraph pole, and light reflection, the signs with occlusions subtracting the original sign without occlusions, and the manually added small white patches. The first and second parts are the major occlusion parts, and the small white patches are used to complement different kinds of occlusions. In the occlusion training set, apart from the occlusion structures, the other values in the samples are 0. The samples in the occlusion dictionary can be linearly combined into different occlusions.

Then, we can use common dictionary learning methods to train the dictionaries A and B. In this work, the used dictionary learning method is K-SVD [44]. The selected samples after training are used to form the atoms in the dictionaries A and B. The set of content training samples of the ith object class is denoted by

$$A_i = [s_{i,1}, s_{i,2}, \ldots, s_{i,n_i}] \in \mathrm{IR}^{m \times n_i} \tag{12}$$

where the atom $s_{i,j}, j = 1, 2, \ldots, n_i$ is an m-dimensional vector stretched by the jth sample of the ith class.

The sample in the common occlusion dictionary is denoted by

$$B = [s'_1, s'_2, \ldots, s'_{n'}] \in \mathrm{IR}^{m \times n'} \tag{13}$$

where the atom $s'_j, j = 1, 2, \ldots, n'$, is an m-dimensional vector stretched by the jth sample of the common occlusion set.

For a test sample $y_0 \in \mathrm{IR}^m$ from class ith, intuitively, y_0 could be well approximated by the linear combination of the samples within A_i and B,

$$y_0 = [A_i, B] \begin{bmatrix} x_i \\ c \end{bmatrix} + z \tag{14}$$

where $x_i = [\alpha_{i,1}, \alpha_{i,2}, \ldots, \alpha_{i,n_i}]^T \in \mathrm{IR}^{n_i}$, and $c = [\beta_1, \ldots, \beta_{n'}]^T \in \mathrm{IR}^{n'}$ are the coefficients, and $z \in \mathrm{IR}^m$ is a noise term with bounded energy $\|z\|_2 < \varepsilon$.

Let $A = [A_1, A_2, \ldots, A_M]$ be the concatenation of the n training samples from all the M big classes, where $n = n_1 + n_2 + \cdots + n_M$, then the linear representation of y can be written in terms of all training samples as

$$y = [A, B] \begin{bmatrix} x \\ c \end{bmatrix} + z \tag{15}$$

where, $x \in \mathrm{IR}^n$, $c \in \mathrm{IR}^{n'}$.

Solve the $l1$-minimization problem,

$$\begin{bmatrix} \widehat{x} \\ \widehat{c} \end{bmatrix} = \arg\min \left\| \begin{bmatrix} x \\ c \end{bmatrix} \right\|_1, \text{ s.t. } \left\| [A, B] \begin{bmatrix} x \\ c \end{bmatrix} - y \right\|_2 \leq \varepsilon \tag{16}$$

where, $\widehat{x} \in \mathrm{IR}^n$, $\widehat{c} \in \mathrm{IR}^{n'}$, and ε is an optimal error tolerance.

The minimization of formula (16) can be achieved by some standard convex optimization technique. We use the $l1$-minimization method described in [42]. This optimization

method can always lead to sparse resolutions in these experiments.

For each class i, let ρ_i be the characteristic function that selects the coefficients associated with the ith class. For $x \in \mathrm{IR}^n$, $\rho_i(x) \in \mathrm{IR}^n$ is a new vector whose only nonzero entries are the entries in x that are associated with class i. In this process, the atoms in B are combined with each A_i in the content dictionary to reconstruct the input sample. The residual $r_i(y)$ is the difference between the reconstruction of input sample and the original input sample. Compute the residual:

$$r_i(y) = \left\| y - [A, B] \begin{bmatrix} \rho_i(\widehat{x}) \\ \widehat{c} \end{bmatrix} \right\|_2, \; i = 1, \ldots, M. \tag{17}$$

where, $\rho_i(x) = [0, \ldots, 0, \alpha_{i,1}, \alpha_{i,2}, \ldots, \alpha_{i,n_i}, 0, \ldots, 0]^T$.

The smallest residual $r_i(y)$ can determine which A_i combined with B lead to the smallest difference; in this case, the label of A_i is the identity:

$$Y = \arg\min r_i(y). \tag{18}$$

After the above steps, different types of traffic signs can be classified.

V. Experiments

To evaluate the presented methods for TSR, a multiclass TSR system is established to detect and recognize 39 different types of German traffic signs in the GTSDB database [40] and the GTSRB database [41]. The 39 different types of traffic signs are shown in Fig. 3 and there are both training and validation sets in the GTSDB and GTSRB databases. The evaluation images in the GTSDB database are all high-resolution images (1360-by-800 pixel), containing zero to six traffic signs. The GTSRB database contains more than 50,000 traffic sign images for training and evaluation of traffic sign recognition methods. The traffic signs in the GTSRB database may appear in every perspective and under every lighting condition. For different verification experiments, we have some other databases for different verification purposes, including the Swedish traffic sign database [39], the Chinese traffic sign database, and the partial-occlusion database. All of the methods are programmed with C++ and OpenCV, and the performances of these methods are tested on a conventional quad-core (i5-3470) 3.19 GHz PC.

A. Experiments on the Proposed Region of Interest Extraction Method and Traffic Sign Detection Method

The experiments in this part are designed to demonstrate that the proposed ROI extraction method of the HCRE can quickly and accurately extract ROI and that the SFC-tree traffic sign detector can achieve very quick detection time when combined with the HCRE.

Apart from the GTSDB database, we add two other databases, the Swedish traffic sign database [39] and the Chinese traffic sign database captured by our lab, to verify that the proposed HCRE can successfully extract ROI of the traffic signs from different countries. The traffic signs in these three datasets are captured with different cameras in different illuminations,

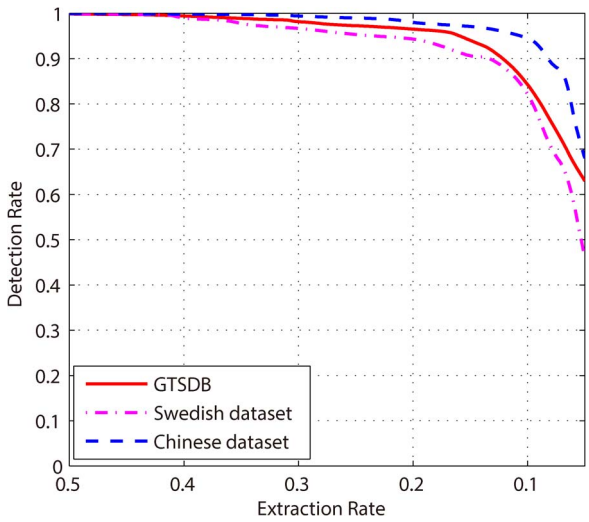

Fig. 8. The detailed performance of the HCRE on three datasets. Three datasets from different countries are test.

TABLE I
Performance of Different ROI Extraction Methods on the Three Datasets From Different Countries

Methods	DR (%)			ER (%)	Time (ms)
	R	B	Y		
RGBE [4]	70.56%	78.21%	36.84%	16.35%	5.0 ms
RGBNT [5]	85.43%	87.90%	52.63%	6.40%	7.3 ms
HST [5]	88.34%	92.34%	54.74%	50.91%	37.5 ms
CMSERs [31]	92.34%	93.53%	62.30%	9.68%	68.7 ms
HCRE	97.65%	98.80%	95.21%	16.90%	17.1 ms

and can be used to verify the robustness and generalization ability of the proposed ROI method. Four ROI extraction methods are used for comparison, including RGB enhancement (RGBE) [4], RGB normalized thresholding (RGBNT) [5], Hue and Saturation thresholding (HST) [5] and color MSERs (CMSERs) [31]. The extraction colors include red (R), blue (B) and yellow (Y). The detailed performance of the HCRE on three datasets is shown in Fig. 8. The statistical data of comparison are shown in Table I. The ROI extraction results are reflected in three parameters: the extraction rate (ER), the detection rate (DR), and extraction time. The DR is the ratio of the number of the extracted traffic signs to the number of all traffic signs. The ER is the ratio of the scanning subwindow number after ROI extraction to the number of the original scanning subwindows.

The statistical data in Table I show that the proposed HCRE method has much higher detection rates than the other methods in extracting traffic signs from different datasets, and both the time consumption and extraction rate of the proposed HCRE method are small enough for ROI extraction. The detection rates of RGBE [4], RGBNT [5] and HST [5] largely rely on the selected thresholds; these methods have low detection rates, especially in extracting yellow signs. The CMSERs [31] can partly achieve color-change invariance and has relative high detection rates; yet, it is limited in extracting yellow color and requires more than 60 ms extraction time. The proposed HCRE has the highest detection rates in extracting traffic signs with red, blue and yellow colors. Furthermore, both the extraction rate and extraction time of the HCRE are very low, which means the HCRE can largely reduce the detection time of the TSR

TABLE II
THE PERFORMANCES OF ROI EXTRACTION, TRAFFIC SIGN DETECTION AND VERIFICATION

Methods	Performance			
	DR (%)	FA (number)	VR (%)	Time (ms)
SFC-tree [2]	95.37%	210	99.72%	187 ms
HCRE+SFC-tree	95.10%	112	99.81%	75 ms

system. Hence, it can be concluded that the proposed HCRE has higher detection rates than the previous ROI extraction methods in [4], [5], and [31], and can achieve high detection rate with low extraction rate and small time consumption.

The traffic sign detection method based on the SFC-tree is described in detail in [2], and it can reach approximately 5 frames per second on the GTSDB database without using any ROI extraction methods. In this part, we test the improvement of the combination of the ROI extraction method of the HCRE and the SFC-tree traffic sign detector. The HCRE method is applied to remove 83.10% of the non-interesting regions on average, and then the SFC-tree detector is used to scan the regions of interest to detect signs; in this case, the extraction results of the HCRE has little effect on the detection rate of the SFC-tree detector because the missing signs of the HCRE are hard to be detected by the SFC-tree detector.

Most of the training samples of the SFC-tree detector are the collected samples from GTSRB [41], and the rest are captured by a vehicle-mounted camera. The training images have been resized to 24 by 24 pixels for training. The validation set is the 900 images in the GTSDB database, and the test signs are the signs that are larger than 24 by 24 pixels. The test parameters are the detection rate (DR), the false alarm number (FA), the verification rate (VR) and the detection time. The verification rate (VR) is the parameter that shows the results of the verification of the OvR SVM. The statistic results are shown in Table II.

From the data in Table II, it is observed that the combination of the HCRE and the SFC-tree detector can save more than half the time of the previous SFC-tree detector, and the verification rate can reach nearly 100%. The SFC-tree detector combined with the HCRE can reach 75 ms per frame on average, which means that there are approximately 110 ms saved for the following complex recognition methods. Furthermore, both the HCRE extraction method and the SFC-tree detector are robust to color changes and fuzzy edges. Besides the statistic data in Table II, the SFC-tree detector can roughly classify the signs into big categories with an accuracy of approximately 86.20% on the GTSDB database. This high accuracy means that instead of directly classifying the detected signs, the classification correctness of the detected signs can be verified using the proposed verification method based on SVM and HOG. As the verification data shown in Table II, the verification rates are nearly 100%; these high verification rates are mainly because the detected signs in the leaf nodes of the SFC-tree are quite different from each other, and the verification method based on SVM and HOG has strong verification and classification ability. This high verification rate means that the OvR SVM can successfully verify different traffic signs in the leaf nodes of the SFC-tree.

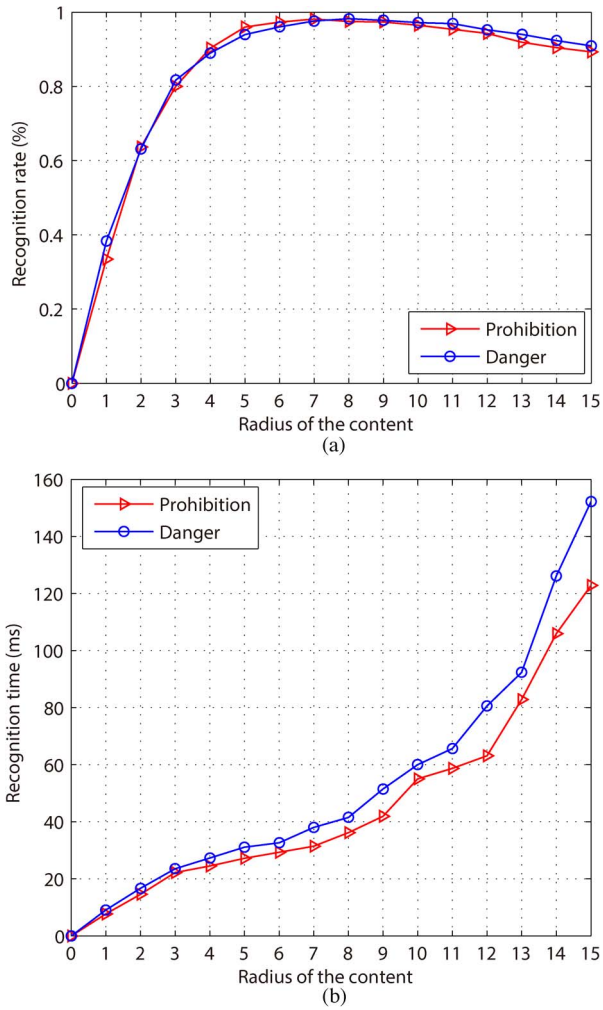

Fig. 9. The performance of the ESRC-based method using different radiuses of content. (a) shows the recognition rates of two different traffic sign categories as the radius of content increases. (b) shows the recognition times of two different traffic sign categories as the radius of content increases.

B. Experiments on the Proposed Traffic Sign Recognition Method

The experiment in this part is designed to test the hypothesises that given the same number of training images, the ESRC method can largely improve the generalization ability of the SRC and reduce occlusion-robust dictionary size. After the detection and verification, only the traffic signs in the circular prohibition sign class and triangular danger sign class need to be recognized. We use the images of these classes in the GTSRB database for training and testing. All the images are aligned and resized into a resolution of 30 $\times$ 30. In our experiments, we do not consider the traffic signs that can not be recognized by human.

We first get the suitable radiuses for the ESRC-based TSR method. As shown in Fig. 9, we test different radiuses of content to get the most suitable radiuses. All the training and verification images are from the training set in the GTSRB; half of these images are for training and the rest images are for testing. The cross-validation strategy is used to get average accuracies. The extreme points of the two curves in Fig. 9(a) are the most suitable radiuses, because the accuracies are the highest in these

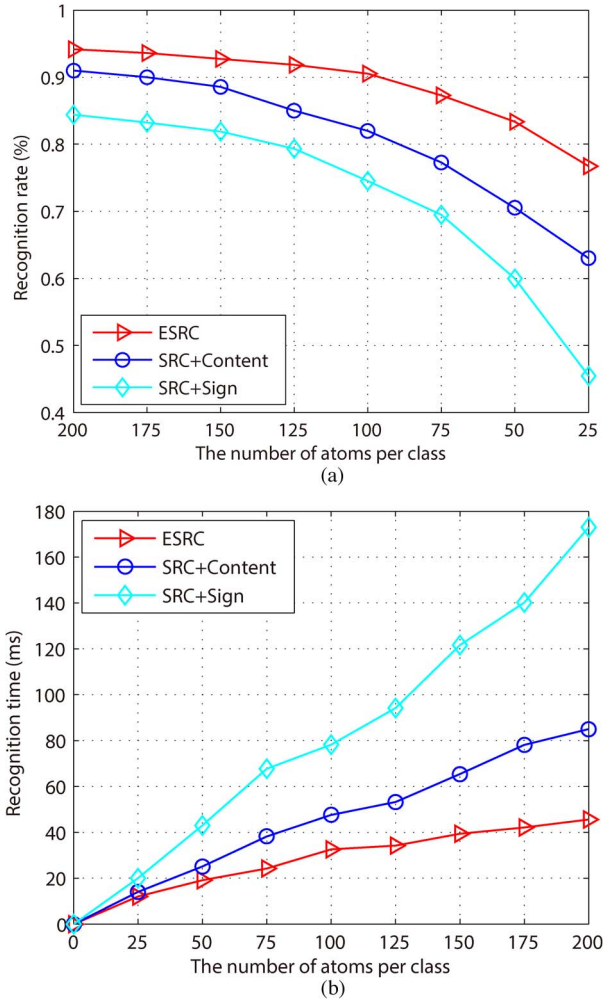

Fig. 10. The recognition results of the TSR methods in recognizing partial occlusion traffic signs. (a) shows the comparative recognition rates in recognizing partial occlusion traffic signs as the number of training images per class decreases. (b) shows the comparative recognition times in recognizing partial occlusion traffic signs as the number of training images per class increases.

points. The most suitable radiuses of prohibition and danger are seven and eight; the recognition times are 31.4 ms and 41.5 ms for the most suitable radiuses of prohibition and danger.

To demonstrate that the proposed ESRC-based traffic sign recognition method is faster and more accurate than the previous SRC methods in recognizing partially occluded traffic signs, we captured and collected 1520 partially occluded traffic signs. There are 18 different types of circular prohibitory traffic signs in this set. These occluded patches include telegraph pole, trunk, reflection of light, etc. The patches occupy ranging from approximately 1/10 to 1/2. The cross-validation strategy is used to get average accuracies. Half of these images are for training and the rest images are for testing; the rest training samples for the SRC-based methods are from the traffic signs with manually added occlusions. Comparisons are made among three approaches, including the proposed ESRC, the SRC with traffic signs as input vectors, and the SRC with contents as input vectors. To test the undersampled effect in recognizing partial occlusion signs, we reduce the number of training samples per class from 200 to 25. Fig. 10 shows the

TABLE III
COMPARISON OF DIFFERENT TSR METHODS

Methods	Classification rate (%)			Time
	S	P	Da	
SVM+ESRC	97.84	99.46	98.21	33 ms
C-CNNs [28]	99.47	99.93	99.07	690 ms*
MS-CNNs [27]	98.61	99.87	98.03	N/A
SRGE [30]	98.06	99.22	97.35	95 ms*

*The recognition time of [28] was gotten on a single GPU. The platform of [30] is a conventional quad-core (i7-950) 3.33 GHz PC.

comparative performance of different SRC-based approaches in recognition rate and recognition time.

As expected, the accuracy of the SRC-based methods deteriorates rapidly as the number of training images decreases, whereas, the accuracy of ESRC deteriorates more slowly than the other SRC-based methods. In the verification with partial occlusion signs, the ESRC performs better than the SRC-based methods in recognition rate and recognition time. In general, the superiority of the ESRC in recognition rate becomes more and more significant as the dictionary size decreases. The main reason is that the proposed ESRC method has small-size and compact dictionaries. In the ESRC-based method, the content without frames can largely reduce the atom length in the dictionaries and reduce the noise effect from the frame; this has been verified through the comparison of the ESRC and the SRC+content with the SRC+Sign in Fig. 10. Furthermore, the occlusion dictionary of ESRC can largely reduce the number of atoms in the dictionaries of ESRC. The curves in Fig. 10 can also verify that the ESRC need much smaller number of atoms to achieve the same accuracy with the SRC+content. Hence, the dictionaries used in ESRC are more compact and have stronger generalization ability than the occlusion dictionary used in classical SRC.

To show that the proposed ESRC-based TSR method is as accurate as the-state-of-the-art TSR methods and fast enough for TSR applications, we compare it with three other recognition methods that have the-state-of-the-art performance on the GTSRB validation database. The data in Table III show the comparison results in four big categories: the speed-limit category (S), the other prohibition category (P) and the danger category (Da). Because the ESRC is used to classify signs in two big categories, we need to classify the traffic signs into coarse categories before the ESRC-based classification. Like the verification method described in Section III, we use the classification method based on the HOG features and the SVM classification. Then, the ESRC-based TSR method classifies the coarse categories into fine categories. There are training and validation databases in the GTSRB database. The samples in the training database are used to form the dictionaries for the ESRC. For the ESRC-based method, there are 200 atoms for each object in the content dictionary and 400 atoms in the occlusion dictionary.

The data in Table III show that the proposed classification time is faster than the other methods, and the ESRC-based classification rates are very close to the reported results of the MS-CNNs [27] and the SRGE [30] and just a bit lower than the best results of the C-CNNs [28]. Having a small computing cost, the proposed classification method includes two parts: the

TABLE IV
PERFORMANCE OF THE PROPOSED TSR SYSTEMS

Systems	Performance		
	Accuracy (%)	FA (%)	Time (ms)
TSR System-A	94.81%	4.10%	115 ms
TSR System-B	91.67%	4.83%	106 ms
TSR System-C	88.42%	5.16%	84 ms

coarse classification based on SVM classification and HOG features, and the fine classification based on ESRC. In this experiment, the ESRC-based fine classification method classifies the traffic signs in coarse categories into fine classes with accuracy of 98.27% in approximately 25 ms on average. There are two main improvements that contribute to the achievements of high accuracy and fast classification time. One is that the proposed ESRC-based classification utilizes the content and occlusion dictionaries to achieve occlusion-robust classification with the generalized small-size dictionaries. The other is that the proposed ESRC-based classification uses the content to form dictionaries, which can largely reduce the atom length in the dictionaries and reduce the noise effect from the frame.

C. Performance of the Overall TSR System

The performance of the proposed multiclass TSR system is verified to detect and recognize 39 different types of traffic signs in the GTSDB dataset. All 900 images in the GTSDB dataset are used. The multiclass TSR system contains three parts: ROI extraction, traffic sign detection and traffic sign classification; we use three parameters to reflect the performance: the accuracy of the system, the false alarm rate (FA) and the average time consumption to process a 1360-by-800 pixel image.

In our system, higher accuracy often means more time consumption. For example, we can adjust the HCRE to detect nearly 100% of traffic signs with an extraction rate of approximately 45.81%, whereas a 96.30% detection rate can reach approximately 17.10% extraction rate, saving approximately an additional 55 ms of detection time. The SFC-tree detector can reach a detection rate of 95.37% and an average detection and verification time of 187 ms. After the ROI extraction of the HCRE, the detection and verification time of the SFC-tree detector has been reduced to 75 ms. For the ESRC-based recognition method, if we reduce the size of the dictionary, the recognition rate will be lower and the recognition time will be faster. Hence, the proposed ESRC-based recognition method also needs to keep a good balance of recognition rate and time consumption.

The statistical data of the proposed TSR system is shown in Table IV. The accuracy in Table IV is the accuracy of the TSR system including ROI extraction, detection and recognition. In Table IV, the TSR System-A is a TSR system with our highest recognition rate, and the TSR System-B and the TSR System-C are faster TSR systems with relative lower accuracies. Because there are different number of signs in one image, multicore programming can be used to improve the processing time.

From the statistical data in Table IV, it can be seen that the established TSR system-A can achieve 94.81% accuracy, 4.10% false alarm rate and 115 ms processing time per frame, which is our trained TSR system with highest accuracy. If we adjust the used parameters, the recognition time can be faster while achieving a relative lower accuracy. As Table IV shown, the TSR system-B and system-C can reach faster processing time and the accuracies of system-B and system-C are just a bit lower than that of the system-A. When some TSR applications need quick recognition time, the system-B or system-C can be used; when high accuracy is needed for some applications, system-A can meet this need. Hence, it is convenient to adjust the parameters to adapt this system to different applications.

VI. CONCLUSION

Several novel methods have been proposed to construct a traffic sign recognition (TSR) system. The presented TSR system can detect and recognize multiclass traffic signs in high-resolution images with high accuracy, and the detection and recognition are partly robust to color changes, blurred edges and partial occlusions.

To address the problem of multiclass traffic sign detection in high-resolution images with high speed and accuracy, we have developed a ROI extraction method called HCRE and the SFC-tree detector to rapidly detect different types of traffic signs. The proposed HCRE method can reduce the detection regions of the SFC-tree detector to save detection time of the TSR system. The traffic sign recognition method based on the ESRC utilizes a content dictionary and an occlusion dictionary to sparsely represent traffic signs, which can reach occlusion-robust recognition with compact dictionaries. The results of the validation experiments show that the presented methods are fast and robust to color changes. The TSR system can reach high accuracy at approximately 8 to 12 frames per second on the GTSDB dataset.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their constructive comments on the manuscript, and would like to thank Bin Wang for doing part of the experiments in the development of this work.

REFERENCES

[1] A. Møgelmose, M. M. Trivedi, and T. B. Moeslund, "Vision-based traffic sign detection and analysis for intelligent driver assistance systems: Perspectives and survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 4, pp. 1484–1497, Dec. 2012.

[2] C. Liu, F. Chang, and Z. Chen, "Rapid multiclass traffic sign detection in high-resolution images," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 6, pp. 2394–2403, Dec. 2014.

[3] A. Escalera, J. M. Armingol, J. M. Pastor, and F. J. Rodríguez, "Visual sign information extraction and identification by deformable models for intelligent vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 5, no. 2, pp. 57–68, Jun. 2004.

[4] A. Ruta, Y. Li, and X. Liu, "Real-time traffic sign recognition from video by class-specific discriminative features," *Pattern Recog.*, vol. 43, no. 1, pp. 416–430, Jan. 2010.

[5] H. Gómez-Moreno, S. Maldonado-Bascón, P. Gil-Jiménez, and S. Lafuente-Arroyo, "Goal evaluation of segmentation algorithms for

traffic sign recognition," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 4, pp. 917–930, Dec. 2010.
[6] F. Ren, J. Huang, R. Jiang, and R. Klette, "General traffic sign recognition by feature matching," in *Proc. Int. Conf. Image Vis. Comput.*, Wellington, New Zealand, 2009, pp. 409–414.
[7] H. Fleyeh, "Shadow and highlight invariant colour segmentation algorithm for traffic signs," in *Proc. IEEE Conf. Cybern. Intell. Syst.*, Bangkok, Thailand, 2006, pp. 1–7.
[8] J. Marinas, L. Salgado, J. Arróspide, and M. Nieto, "Detection and tracking of traffic signs using a recursive Bayesian decision framework," in *Proc. IEEE Conf. Intell. Veh. Symp.*, Washington, DC, USA, 2011, pp. 1942–1947.
[9] S. Maldonado-Bascon, S. Lafuente-Arroyo, P. Gil-Jimenez, H. Gomez-Moreno, and F. Lopez-Ferreras, "Road-sign detection and recognition based on support vector machines," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 2, pp. 264–278, Jun. 2007.
[10] U. L. Jau, C. S. Teh, and G. W. Ng, "A comparison of RGB and HSI color segmentation in real-time video images: A preliminary study on road sign detection," in *Proc. Int. Symp. Inf. Tech.*, Kuala Lumpur, Malaysia, 2008, pp. 1–6.
[11] J. F. Khan, S. M. A. Bhuiyan, and R. R. Adhami, "Image segmentation and shape analysis for road-sign detection," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 1, pp. 113–126, Mar. 2009.
[12] M. S. Prieto and A. R. Allen, "Using self-organising maps in the detection and recognition of road signs," *Image Vis. Comput.*, vol. 27, no. 6, pp. 673–683, May 2009.
[13] K. Zhang, Y. Sheng, and J. Li, "Automatic detection of road traffic signs from natural scene images based on pixel vector and central projected shape feature," *IET Intell. Transp. Syst.*, vol. 6, no. 3, pp. 282–291, Sep. 2012.
[14] A. Ruta, F. Porikli, S. Watanabe, and Y. Li, "In-vehicle camera traffic sign detection and recognition," *Mach. Vis. Appl.*, vol. 22, no. 2, pp. 359–375, Mar. 2011.
[15] R. Kastner, T. Michalke, T. Burbach, J. Fritsch, and C. Goerick, "Attention-based traffic sign recognition with an array of weak classifiers," in *Proc. IEEE Intell. Veh. Symp.*, San Diego, CA, USA, 2010, pp. 333–339.
[16] M. A. Garcia-Garrido *et al.*, "Robust traffic signs detection by means of vision and V2I communications," in *Proc. 14th Int. IEEE ITSC*, Oct. 2011, pp. 1003–1008.
[17] A. Gonzalez *et al.*, "Automatic traffic signs and panels inspection system using computer vision," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 2, pp. 485–499, Jun. 2011.
[18] F. Moutarde, A. Bargeton, A. Herbin, and L. Chanussot, "Robust onvehicle real-time visual detection of American and European speed limit signs, with a modular traffic signs recognition system," in *Proc. IEEE Intell. Veh. Symp.*, Istanbul, Turkey, 2007, pp. 1122–1126.
[19] N. Barnes, A. Zelinsky, and L. S. Fletcher, "Real-time speed sign detection using the radial symmetry detector," *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 2, pp. 322–332, Jun. 2008.
[20] G. Loy and N. Barnes, "Fast shape-based road sign detection for a driver assistance system," in *Proc. IEEE Conf. Int. Robots Syst.*, 2004, pp. 70–75.
[21] G. Wang, G. Ren, Z. Wu, Y. Zhao, and L. Jiang, "A robust, coarse-to-fine traffic sign detection method," in *Proc. Int. Joint Conf. Neural Netw.*, Dallas, TX, USA, 2013, pp. 754–758.
[22] F. Zaklouta and B. Stanciulescu, "Segmentation masks for real-time traffic sign recognition using weighted HOG-based trees," in *Proc. Int. Conf. Intell. Transp. Syst.*, Washington, DC, USA, 2011, pp. 1975–1959.
[23] K. Doman *et al.*, "Construction of cascaded traffic sign detector using generative learning," in *Proc. Int. Conf. Innovative Comput., Inf. Control*, Dec. 2009, pp. 889–892.
[24] X. Baró, S. Escalera, J. Vitri, O. Pujol, and P. Radeva, "Traffic sign recognition using evolutionary AdaBoost detection and forest-ECOC classification," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 1, pp. 113–126, Mar. 2009.
[25] H. Fleyeh and E. Davamia, "Eigen-based traffic sign recognition," *IET Intell. Transp. Syst.*, vol. 5, no. 3, pp. 190–196, Sep. 2011.
[26] Y. Wu, Y. Liu, J. Li, H. Liu, and X. Hu, "Traffic sign detection based on convolutional neural networks," in *Proc. Int. Joint Conf. Neural Netw.*, Dallas, TX, USA, 2013, pp. 747–753.
[27] P. Sermanent and Y. LeCun, "Traffic sign detection with multi-scale convolutional networks," in *Proc. Int. Joint Conf. Neural Netw.*, San Jose, CA, USA, 2011, pp. 2809–2813.
[28] D. Ciresan, U. Masci, J. Schmidhuber, and J. Schmidhuber, "Multi-column deep neural network for traffic sign classification," *Neural Netw.*, vol. 32, pp. 333–338, Aug. 2011.
[29] B. M. Chandrasekhar, V. S. Babu, and S. S. Medasani, "Traffic sign representation using sparse-representations," in *Proc. Int. Conf. Intell. Syst. Signal Process.*, Anand, India, 2013, pp. 369–372.
[30] K. Lu, Z. Ding, and S. Ge, "Sparse-representation-based graph embedding for traffic sign recognition," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 4, pp. 1515–1524, Dec. 2012.
[31] J. Greenhalgh and M. Mirmehdi, "Real-time detection and recognition of road traffic signs," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 4, pp. 1498–1506, Dec. 2012.
[32] X. Yuan, X. Hao, H. Chen, and X. Wei, "Robust traffic sign recognition based on color global and local oriented edge magnitude patterns," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 4, pp. 1466–1477, Aug. 2014.
[33] F. Zaklouta and B. Stanciulescu, "Real-time traffic-sign recognition using tree classifiers," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 4, pp. 1507–1514, Dec. 2012.
[34] M. Meuter, C. Nunn, S. M. Görmer, S. Müller-Schneiders, and A. Kummert, "A decision fusion and reasoning module for a traffic sign recognition system," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 4, pp. 1126–1134, Dec. 2011.
[35] A. Ruta, Y. Li, and X. Liu, "Robust class similarity measure for traffic sign recognition," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 4, pp. 846–855, Dec. 2010.
[36] K. Duan and S. Keerthi, "Which is the best multiclass SVM method? An empirical study," in *Proc. Int. Conf. Multiple Classifier Syst.*, Berlin, German, 2005, pp. 278–285.
[37] W. Deng, J. Hu, and J. Guo, "Extended SRC: Undersampled face recognition via intraclass variant dictionary," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 9, pp. 1864–1870, Apr. 2012.
[38] P. Viola and M. J. Jones, "Robust real-time face detection," *Int. J. Comput. Vis.*, vol. 57, no. 2, pp. 137–154, May 2004.
[39] F. Larsson and M. Felsberg, "Using Fourier descriptors and spatial models for traffic sign recognition," in *Proc. 17th Scandinavian Conf. Image Anal.*, Helsinki, Finland, 2011, pp. 238–249.
[40] S. Houben, J. Stallkamp, J. Salmen, M. Schlipsing, and C. Igel, "Detection of traffic signs in real-world Images: The German traffic sign detection benchmark," in *Proc. Int. Joint Conf. Neural Netw.*, Dallas, TX, USA, 2013, pp. 1–8.
[41] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Legel, "Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition," *Neural Netw.*, vol. 32, pp. 323–332, Aug. 2012.
[42] A. Yang, A. Ganesh, S. Sastry, and Y. Ma, "Fast l1-Minimization algorithms and an application in robust face recognition: A review," in *Proc. Int. Conf. ICIP*, Hong Kong, China, 2010, pp. 1849–1852.
[43] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. CVPR*, 2005, pp. 886–893.
[44] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.

Chunsheng Liu received the B.S. degree in industrial automation and the M.S. degree in pattern recognition and machine intelligence from Shandong University, Jinan, China, in 2009 and 2012, respectively. He is currently working toward the Ph.D. degree in pattern recognition and machine intelligence with the School of Control Science and Engineering, Shandong University. His research interests include pattern recognition, machine learning, and intelligent transportation systems.

Faliang Chang received the B.S. and M.S. degrees from Shandong Polytechnic University, Jinan, China, in 1986 and 1989, respectively, and the Ph.D. degree in pattern recognition and intelligence systems from Shandong University, Jinan, in 2003. Since 2003, he has been a Professor of pattern recognition and machine intelligence with the School of Control Science and Engineering, Shandong University. His current research interests include computer vision, image processing, intelligent transportation systems, and multicamera tracking methodology.

Zhenxue Chen received the M.S. degree in information science and engineering from Wuhan University of Science and Technology, Wuhan, China, in 2003 and the Ph.D. degree in pattern recognition and intelligent systems from Huazhong University of Science and Technology, Wuhan, in 2007. Since 2007, he has been an Associate Professor with the School of Control Science and Engineering, Shandong University. His current research interests include image processing and analysis, pattern recognition, computer vision, and information safety.

Dongmei Liu received the B.S. degree in electronic information science and technology from Shandong Normal University, Jinan, China, in 2010 and the M.S. degree in computer science and technology from Fujian Normal University, Fuzhou, China, in 2013. She is currently working toward the Ph.D. degree in pattern recognition and machine intelligence with the School of Control Science and Engineering, Shandong University, Jinan. Her research interests include pattern recognition, cognitive science, and intelligent transportation systems.