CrossMark

An overview of traffic sign detection and classification methods

Yassmina Saadna[1] · Ali Behloul[1]

Abstract Over the last few years, different traffic sign recognition systems were proposed. The present paper introduces an overview of some recent and efficient methods in the traffic sign detection and classification. Indeed, the main goal of detection methods is localizing regions of interest containing traffic sign, and we divide detection methods into three main categories: color-based (classified according to the color space), shape-based, and learning-based methods (including deep learning). In addition, we also divide classification methods into two categories: learning methods based on hand-crafted features (HOG, LBP, SIFT, SURF, BRISK) and deep learning methods. For easy reference, the different detection and classification methods are summarized in tables along with the different datasets. Furthermore, future research directions and recommendations are given in order to boost TSR's performance.

1 Introduction

The United Nations estimates that between 2010 and 2020 the number of road deaths will increase by up to 50%, that is, about 1.9 million people. To reverse this trend, the UN established, in 2011, the 1st "Decade of Action for Road Safety."

✉ Yassmina Saadna
yassmina.saadna@gmail.com

Ali Behloul
a.behloul@univ-batna2.dz

[1] LaSTIC Laboratory, Department of Computer Science, University of Batna 2, Batna, Algeria

Driver Assistance Systems can help reduce the number of accidents by automating tasks such as lane departure warning systems, traffic sign recognition.

The recognition of traffic signs has received increasing attention in recent years; it is even considered as a highly important feature of intelligent vehicles. Traffic signs carry substantial useful information that might be disregarded by drivers due to driving fatigue or searching for an address reasons. These drivers are also likely to pay less attention to traffic signs on driving in threatening weather. Therefore, making enhancement initiatives, like increasing driving safety along with improving automatic detection and road sign recognition system, is becoming indispensable to help decrease road death toll. These enhancements, however beneficial they may seem, meet several external non-technical challenges such as lighting variations, scale and weather conditions changes, occlusions and rotations, which may eventually decrease the traffic sign recognition systems performance.

The main issue of the problem in the traffic sign recognition system is not how to detect or recognize, with high recall, a traffic sign in a fixed image. It is rather about how to obtain a high precision in videos big data. To illustrate the problem of false alarms, a traffic sign recognition system installed in a smart phone, with 30 shot frames per s, (108,000 frames in a 1-h video), was considered. If we suppose that in every 4 min we detect a sign and that—with reference to the speed of a car—every sign spans 2 s, this means that in the course of 1 h we will find a total of 15 signs and that every sign will display in 60 frames (900 frames that contain signs while 107,100 does not). Supposing also that the system has a 1% false positive accuracy rate, this means that there are 17 detected false alarms in 1 min (1071 in 1 h) and 1 true positive and 68 false alarms in 4 min by consequent. This may eventually lead to most users disabling their applications.

 Springer

Fig. 1 Traffic signs types: **a** mandatory sign, **b** temporary sign, **c** warning sign, **d** prohibition sign, **e** reservation sign

Traffic sign recognition systems consist of three main stages: localization, detection, and classification. In the case of any false alarm in the detection stage, performance will be lower in the classification one; this is due to the fact that the classifier is not usually trained on false alarms.

Road signs have many discriminating features on the basis of which they are classified. According to their shapes and colors, these are five main classes: warning signs (red triangle), prohibition signs (red circular), reservation signs (rectangular blue), mandatory signs (circular blue), and temporary signs (yellow triangle). Examples of traffic signs for each of the categories mentioned above are shown in Fig. 1.

The aim of this paper is to present an overview of some recent and efficient traffic sign detection and classification methods; some authors like [1–4,15,36] are also preferred to make a study on this domain.

In Sect. 2, traffic sign detection methods are presented; they have been divided into three categories: color-based, shape-based, and learning-based methods, including deep learning methods. In Sect. 3, traffic sign classification methods are stressed, firstly we cite learning methods based on hand-crafted features, and then, we mention deep learning methods. Moreover, different publicly available traffic sign detection and classification datasets are also presented to help meet the goal of this paper. Section 4 describes the future research directions which can incorporate with researchers in their future works. Finally, a conclusion is expected.

2 Detection methods

As mentioned above, we can classify detection or localization methods into three fundamental classes: color-based, shape-based, and learning-based methods. According to the nature of the problem and system requirements, we can decide upon the best method to apply; for example, methods based on color information can used with high-resolution dataset, however, not with grayscale images.

2.1 Color-based methods

The dominant color-based segmentation is applied to detect regions of interest. There are specific characteristic colors of traffic signs: red, blue, and yellow. These characteristics, however, indicate sensitivity to various factors, such as the age of signs and the variation of light, which make the segmentation an arduous process. In order to overcome this problem, authors are working on different color spaces among which we mention the following:

2.1.1 RGB space

De La Escalera et al. [5] adopt RGB space by reason of HSI formulas are nonlinear. The authors use the relation between the components as it is presented in the following expression:

$$g(x, y) = k_1 \begin{cases} R_a \leq f_r(x, y) \leq R_b \\ \\ TG_a \leq \dfrac{f_g(x,y)}{f_r(x,y)} \leq TG_b \\ \\ TB_a \leq \dfrac{f_g(x,y)}{f_b(x,y)} \leq TB_b \end{cases} \tag{1}$$

$$g(x, y) = k_2 \quad \text{in other case}$$

$f_r(x, y)$, $f_g(x, y)$, and $f_b(x, y)$ are, respectively, the functions that provide the red, green, and blue levels of each point of the image.

Thresholding is used by several authors as [6–8]. Their approaches, however, highly related to the selected thresholds, which make the comparison of their performances a difficult task.

Ruta et al. [7] applied filtering for each pixel: $X = [x_R, x_G, x_B]$ and $S = x_R + x_G + x_B$)

$$f_R(X) = \max(0, \min(x_R - x_G, x_R - x_B)/S) \tag{2}$$
$$f_B(X) = \max(0, \min(x_B - x_R, x_B - x_G)/S)) \tag{3}$$
$$f_Y(X) = \max(0, \min(x_R - x_B, x_G - x_B)/S)) \tag{4}$$

In this approach, they generate three maps red, blue, and yellow for each RGB image. The dominant color has a high intensity while deteriorated signals have low intensities. King et al. [9] prefer the R'G'B' space. At first, they normalize the three RGB channels by intensity I:

$$I = \frac{R' + G' + B'}{3} \tag{5}$$

$$r = \frac{R'}{I}, \quad g = \frac{G'}{I}, \quad b = \frac{B'}{I} \tag{6}$$

Then, they construct four new images according to the equation proposed by [10]:

$$R = r - \frac{(g + b)}{2} \tag{7}$$

$$G = g - \frac{(r + b)}{2} \tag{8}$$

$$B = b - \frac{(r+g)}{2} \qquad (9)$$

$$Y = \frac{r+g}{2} - \frac{|r-g|}{2} - b \qquad (10)$$

The dominant color has a great intensity that facilitates the extraction of the panels. Thresholding is used to binarize the four images (R, G, B, Y); morphological operations are then applied to remove the unwanted pixels. It is worth highlighting that this approach is capable of detecting up to 93.63% of the panels.

King et al.'s approach is adopted in [11]. A filter is proposed to eliminate undesirable pixels with a view to reduce the execution time.

2.1.2 HSV space

Yakimov [12] considered that is not possible to detect traffic signs in real images by applying a simple threshold directly in the RGB space due to lighting variations; this is what urged them to choose the HSV space. They used experimental method to determine the optimal threshold values for red color as it is presented in the following expression:

$$(0.0 H < 23) \, \| \, (350 < H < 360)$$
$$(0.85 < S \le 1) \qquad (11)$$
$$(0.85 < V \le 1)$$

After segmentation, they used a modified algorithm presented in [13] to denoise segmented images. The advantage of the denoising algorithm is that only noise will be removed and the regions of interest stay unfiltered.

Wang et al. [14] also choose HSV space, and they found that the classical thresholding method gives good results in many different lighting conditions except for the cases of color cast or poor lighting condition. They proposed a new thresholding method by using the color information of neighboring pixels. Firstly, the red degree of each point c is calculated to get a new image $f_R(c)$ with the following equation:

$$f_R(c) = \begin{cases} S(c) \frac{\sin(H(c) - 300°)}{\sin(60°)} & \text{if } H(C) \in [300°, 360°] \\ S(c) \frac{\sin(60°) - H(c)}{\sin(60°)} & \text{if } H(C) \in [0°, 60°] \\ 0 & \text{others} \end{cases} \qquad (12)$$

Secondly, the normalized red degree $f_{NR}(x)$ is calculated as the following:

$$f_{NR}(x) = \frac{(f_R(x) - \mu_R(\omega_x))}{(\sigma_R(\omega_x))} \qquad (13)$$

$\mu_R(\omega_x)$ and $\sigma_R(\omega_x)$ are the mean and the variance of the red degrees of the pixels in the window ω_x centered on x.

Thirdly, the normalized intensity f_{NI} is calculated with the following equation:

$$f_{NI}(x) = \frac{(f_I(x) - \mu_I(\omega_x))}{(\sigma_I(\omega_x))} \qquad (14)$$

$f_I(x)$ is the intensity of the pixel x; $\mu_I(\omega_x)$ and $\sigma_I(\omega_x)$ are the mean and the variance of the intensities of the pixels in the window ω_x. Finally, the red bitmap B is given as the following:

$$B(x) = \begin{cases} 1 & \text{if } f_{NR}(x) > \max(THR1, f_{NI} + THR2) \\ 0 & \text{others} \end{cases} \qquad (15)$$

Basconand et al. [16] combine thresholding on H and S components with the achromatic decomposition. This method, however simple and fast, is not robust to signal deterioration and illumination changes. Fleyeh et al. [17] use thresholding on H, S, and V components; this method is resistant to lighting changes, but is costly in computation time. Vitabile et al. [18] on the other side use dynamic aggregation technique of pixels to segment the image.

2.1.3 HSI space

Several authors have chosen to use HSI space because its hue component is invariant to changes in the luminance.

Escalera et al. [19] have chosen HSI space to detect the signs. only H and S components are used to compensate for the problem of brightness variations. The authors constructed two look-up tables (LUTs): one for the hand and the other for the S component. The idea is that each LUT makes up for the other, i.e., if a component has false values, the other can compensate. Once the LUTs are applied, the resulting images are multiplied and compared with the conventional logical AND.

Fang et al. [20] assume that each particular color of a sign can be represented by a hue value distributed with a Gaussian manner variance σ^2. The set of all these hue values is denoted as $[h_1, h_2, \ldots \ldots, h_q]$; then, a z degree of similarity between the color of a pixel h and color sign hues h_k is calculated as the following:

$$z = \max_{(k=1,.q)}$$
$$z_k = \frac{1}{\sqrt{2\pi}\sigma} \exp((h - h_k)^2 / 2\sigma^2) \qquad (16)$$

They will not get a segmented image, but an image where each pixel represents the similarity between the pixel color

and standard one. One of the drawbacks of this method is that its calculations are not linear.

2.1.4 YUV space

The YUV space is a three-component model based on the separation of luminance and chrominance information:

$$
\begin{aligned}
Y &= 0,299R + 0,587G + 0,114B \\
U &= 0,493(B - Y) \\
V &= 0,877(R - Y)
\end{aligned}
\tag{17}
$$

Rectangular information panels being detected in [21] by a colorimetric thresholding in YUV are followed by a horizontal and vertical projection of the gradient on a recognition of Kanji (Japanese writing system). In this approach, the performance is only illustrated by some examples. The choice in [22] fells also on the YUV space after previous color correction given the pixel values of the theoretically gray floor ($R = G = B$).

In [23], the authors compared different segmentation methods in order to find the best method in the field of road signs recognition. They classified the methods of segmentation into three main categories: segmentation with binarization, chromatic/ achromatic decomposition, and edge detection and then proposed a new segmentation method in which they combined SVM with LUT (Look Up Table).

After implementing the different methods and conducting an extensive research to find the best method, they concluded that:

- In the single images, the best results were obtained with the RGB method; however, in the videos, the best results were seen when applying LUT SVM method;
- The edge detection can be used as a complement to the segmentation method but cannot be used alone;
- The standardization of RGB color space proves good performance with less operations; on the other hand, HSV space, though it gives higher results, takes a long time of execution, which makes it a less efficient method. Thus, why would we use a nonlinear transformation if just simple normalization is good enough.

Unlike the color thresholding and extreme region extraction methods used in previous approaches, the recent approach [24] uses High-Contrast Region Extraction (HCRE), motivated by the cascaded detection methods, to extract ROI with high local contrast. This can keep a compromise between the detection and extraction rates. Taking advantage of the observation that different types of traffic signs have relative high contrast in local regions, the HCRE can reject approximately 83.10% of the non-

interesting regions with low local contrast such as the sky, roads and some buildings, and thus boosting the detection speed of the SFC-tree detector from 5 to more than 10 frames per s in their experiments.

2.2 Shape-based methods

The authors, in this approach, do not absolutely consider color segmentation as a discriminative feature due to its sensitivity to various factors such as the distance of the target, weather conditions, time of the day, and reflection of the signs. Conversely, detection of the signs is made from the edges of the image analyzed by structural or comprehensive approaches. Shape-based methods are generally robust than colorimetric methods by reason they can process images in grayscale and treat their gradients. However, they are costly in computation time, given the fact that the rate of treatment depends largely on the number of detected edges. However, shape-based methods can treat grayscale images; in some countries, such as Japan, there are pairs of different signs in the highway code which, when converted to grayscale, appear exactly the same. To be able to distinguish them, an amount of color information is absolutely needed [7]. On the other hand, some authors adopt the color feature to localize the region of interest and complete with shape methods in order to detect the signs position and recognize its geometric form.

Vitabile et al. [18] use the totality of pixels to recognize the geometric form of the sign, where each form of a road sign is represented by a binary image of fixed size (36 × 36 pixels). After detecting the regions of interest with the dynamic aggregation technique of pixels, they resize them on the same scale of binary image to calculate a measure of similarity with all forms of road signs using the Tanimoto coefficient: The value of this coefficient is normalized, i.e., the more it is close to 1, the more the model and region of interest are similar. If X and Y are considered sets of pixels of binary images to be compared, then the Tanimoto coefficient S will be defined as follows:

$$
S = \frac{|x \cap y|}{|x \cup y|}
\tag{18}
$$

The results obtained show that more than 86% of signs are detected among 620 images of 24 classes. Authors in [25] used template matching to filter out the regions which do not contain traffic signs. A sliding window, having the same size of the template, slides on the image to search its most similar region. The function of similarity used is the mean square error (MSE).

$$
MSE = \frac{1}{MN} \sum_{y=1}^{N} [T(x, y) - I(x, y)]^2
\tag{19}
$$

Table 1 Results achieved by Youssef et al. [81]

Dataset	Prohibitive		Danger		Mandatory	
	Accuracy (%)	Time (ms)	Accuracy (%)	Time (ms)	Accuracy (%)	Time (ms)
GTSDB	98.67	231	96.01	234	90.43	243
DITS	97.87	198	98.12	197	89.71	200

- $T(x, y)$: the intensity value of the pattern image at the position (x, y);
- $I(x, y)$: the intensity of the input image at the position (x, y);
- M and N: the width and height of the image, respectively.

A Hough transform has been used by [26] to detect edges of the panels and select the closed contours which make their approach sensitive to noise and occlusion. This approach is able to detect 97% of the 435 speed limit signs and 94% of 312 danger signs in a time ranging between 20 and 200 ms/img according to the number of processed outline. It is worth mentioning that Miura et al. [26] have also applied the transformed Hough for circular signals.

Youssef et al. [81] also choose to use HOG descriptor with a 40×40 detection window (block of 10×10 and 2×2 striding). The GTSRB dataset is used for first round training, DITS (The Data Set of Italian Traffic Signs) is used to the second round, and the both are used in the third round. In the aims to reduce both the false positive and time execution, a color segmentation step with improved HSV is used before the detection step. Established results are illustrated in Table 1.

In [27], the authors preferred to use Radial Symmetry Transformation to detect speed limit signs. This method, a variant of the circular Hough transform, is particularly made to detect possible occurrences of circular signs. The authors reached a detection rate of 96% for 10% false positives in a 50 ms/img for a 320–240 image. Authors in [28] are also using Radial Symmetry Transform to detect other geometric shapes like octagon, square, and triangle. However, they reached a detection rate of 100% for triangle shape, and the major drawback is that it can only find the size and position of forms: It cannot distinguish a Yielding passing panel from an intersection one.

In [29], a process of extracting contours with their associated tangents accuracy is made, and they also propose three algorithms: RANSAC type of quadrilateral, ellipse, and triangle detection. The final shape will be chosen based on compatibility degrees provided by each algorithm. The method presented in [30] is adopted to estimate the center of the shape using only three points with their tangents. Over 80% of 1400 test images are properly detected, with 5% false positive, in a processing time that ranges between 15 and 20 s for 1980 1024 image.
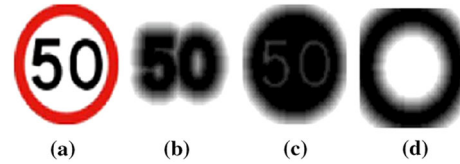

Fig. 2 CDT (*Color* Distance Transform) normalized: **a** original image in discrete *color*, **b** black CDT, **c** white CDT, **d** *red CDT*. Shadows regions mean a small distance, extract of [8] (color figure online)

Qin et al. [31] exploit Fourier descriptors (FDs) to describe the contour, the main reason of using FDs is their robustness to the rotation, scaling, and translation, and then, a process of matching FDs is applied. A database containing over 20000 images has been created by recording sequences from over 350 km of Swedish highways and city roads; this database is used to evaluate the proposed approach. The average detection rate achieved is 77.08% with 641 images in total.

Several works, like [8], preferred to use the distance transform which converts a binary image to an image where each pixel value represents the distance from the pixel to the nearest pixel feature. The researchers propose a variant of DT called Color Distance Transform (CDT) in which they applied a comparison between the real image and the template one in a representation of the discrete color. To facilitate the comparison, they calculate a distance transform DT for each discrete color. Then, pixels having a discrete color are considered as feature pixels while other pixels are not. The result of this process is illustrated in Fig. 2.

Qin et al. [32] use the vector of the distance to border (DTB): It is the distance between the outer contour and the bounding box in which they calculate, for each segmented blob, the four DTB vectors (left, right, top, and bottom) as shown in Fig. 3. This distance is robust to translations and rotations. Since the blobs should be zoomed to 36×36, the proposed approach is not robust to scale changes. DTB vectors will be then chosen as a feature to classify blobs with SVM. The use of four distance vectors for each segmented blob makes the DTB robust to occlusions, that is why it is also used by [63] as a feature to recognize the geometric form of signs.

Another approach is proposed by [11] to recognize the shape of the sign candidate; the idea is to compare the detected pattern with the BoxOut rectangle that encompasses. A score of intersections is calculated between the contour of

Fig. 3 Distance to border (DTB) **a** segmented blob, **b** binary image of **a**, **c** DTB of blob [32]
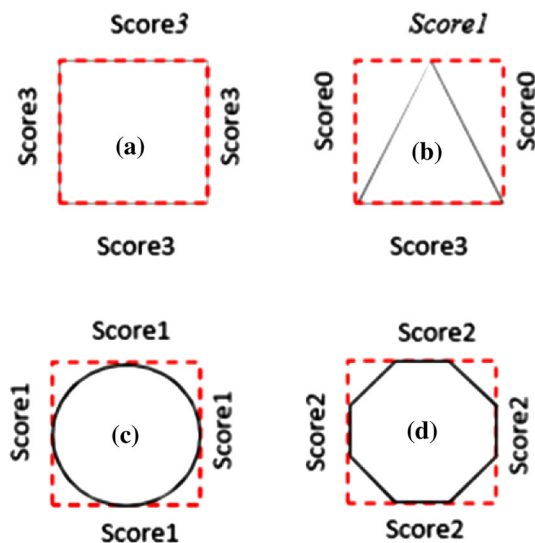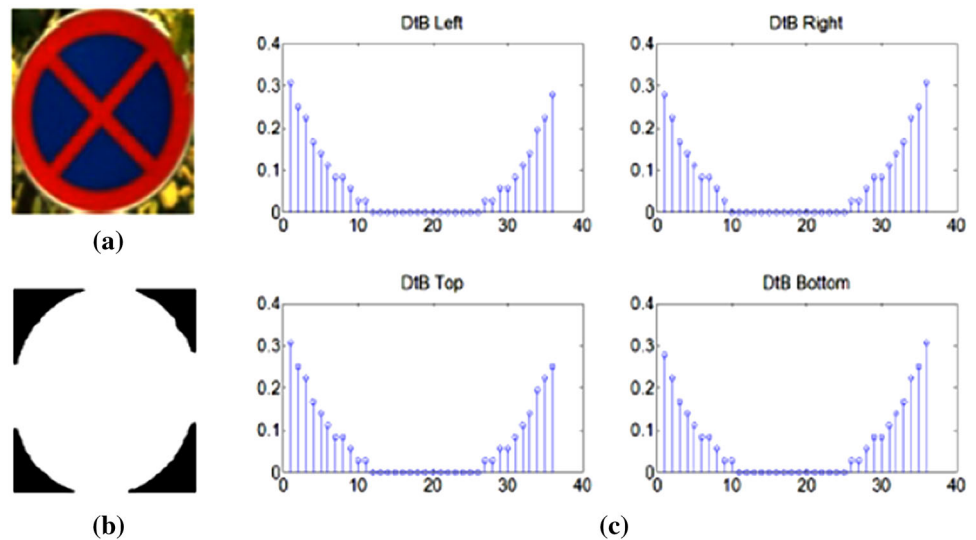

(a)

(b)

(c)

Fig. 4 The proposed detection approach of [11] **a** rectangle, **b** triangle, **c** circle, **d** octagon

the pattern and the four lines of the BoxOut; it is shown in Fig. 4. This approach is capable of detecting 95.65% with 2.17 % false alarms. The used dataset consists of 48 images with 360×270 pixels containing three different traffic signs. The disadvantage of this approach is its weakness to occlusions and noises.

Recently, a new circle detection algorithm EDCircles [75] is used by [76] to detect circular traffic signs. The algorithm consists to, firstly, use Edge Drawing Parameter Free (EDPF) to detect edges in grayscale images, then extract circular arcs from edges and combine arcs having similar radius together, and finally, the candidate circles are validated. The accuracy rate detection achieved by GTSDB dataset is 93.78% with 0.99% false positive for prohibitive signs and 75.51% with 2.04% false positive for mandatory signs.

Table 2 summarizes shape-based methods; it is clear from the table that the previous detection methods do not use the same database, which prevents us from comparing them to one another. Authors in [29] use a big dataset with large images, reached an acceptable detection rate that is still very far from the real-time application. The method used by [11], however having real time, its dataset is small, which keeps us from considering it a highly efficient method.

2.3 Learning-based methods

The previous methods share a common weakness in several factors such as lighting changes, occlusions, scale change, rotation, and translations. However, these problems could also be treated using machine learning, but it requires a large database of annotated data.

Increasing complexity detectors cascade is used by Viola and Jones [33], each detector is a set of classifiers based on Haar wavelet, and these classifiers use a learning algorithm AdaBoost.

Authors in [34] used the Viola–Jones detector to detect triangular traffic sign. The detector was trained using about 1000 images of relatively poor quality. The obtained detector achieved a very high true positive rate (ranging from 90 to 96%) depending on the training set and the configuration of the detector. In their experiments [34,35], they observed two main weaknesses of the Viola–Jones detector:

- the requirement of a large number of training images;
- false positive rates.

Nevertheless, their research also indicates that the Viola–Jones detector is robust to noise and low-quality training data [36].

Table 2 Shape-based detection methods

Paper	Detection method	Detection rate (%)	False alarms	Used dataset	Time
Barnes et al. [27]	Radial symmetry transformation	100	10	Speed limit signs with 320 × 240 pixels	50 ms/img
Behloul et al. [11]	Score of intersection with boxout	95.65	2.17%	48 images with 360 × 270 pixels1	Not exceed 1 ms/image
Berkaya et al. [76]	EDCircles	93.78 for prohibitive signs / 75.51 for mandatory signs	0.99% for prohibitive signs / 2.04% for mandatory signs	GTSDB	36 ms/image
Garcia-Garrido et al. [26]	Hough transformation	97 of speed limit / 94 of panels danger	– / –	435 speed limit signs / 312 danger signs	20 until 200 ms/img
Soheilian et al. [29]	3 Proposed algorithms	80	5	1400 images with 1980 × 1024 pixels	15 until 20 s/img
Vitabile et al. [18]	Distance of similarity	86	–	620 img with 24 classes	–
Youssef et al. [81]	HOG	Prohibitive: 98.67 / Danger: 96.01 / Mandatory: 90.43 / Prohibitive: 97.87 / Danger: 98.12 / Mandatory: 89.71		GTSDB / DITS	231ms / 234 ms / 243 ms / 198 ms / 197 ms / 200 ms

Bario et al. [37] proposed an attentional cascade which is composed of a set of classifiers where each entry of the cascade is the region of interest detected by the previous classifier, and Adaboost algorithm was used to learn the classifiers. The researchers also proposed a classification strategy Forest-ECOC (Error Correcting Output Codes) to overcome the multiclass problem, and the idea is to integrate several trees in the Framework ECOC. The authors obtained the following results:

- Prohibition circle: 70% of detections with 3,65% false positives per image;
- Obligation circle: 60% of detections with 0,95% false positives per image;
- Danger triangle: 65% of detections with 2,25% false positives per image;
- Triangle Right of way: 75% of detections with 2,8% false positives per image.

The rate of treatment is not given because the algorithm is operated offline. The case of rectangular panels is not addressed, and the color information is not used.

Priscariu et al. [38] used Adaboost classifier based on viola and Jones detector, followed by an SVM operating on normalized RGB channels. The system is robust to motion blur through 3D region-based tracking.

Chen et al. [85] explore both exploring Adaboost and support vector regression (SVR) together to detect traffic signs. The proposed approach is evaluated over three datasets (GTSD, BTSD, STSD) using an Intel Core-i7 4770 with 8G RAM. This approach is not real time, the detection time varied from 0.05 to 0.5 s, and the training time is 16 min. The recall obtained on STSD is 80.85% with a precision of 94.52%, and Table 3 shows the results achieved on GTSD and BTSD.

Researchers in [19] used genetic algorithms in the detection step. The authors applied a parallel search in different directions following by an optimization process that mimics natural evolution and selection. However, their approach is robust to scale changes, rotation, weather conditions, and partial occlusion, and it is not a real-time application. The neural networks are used by [39] to recognize the shape of the panels, but this process is not linear with time 2 s/img.

Table 3 Results achieved on GTSD and BTSD by Chen et al. [85]

Categories	GTSD (%)	BTSD (%)
Mandatory	99.87	97.78
Danger	100	99.88
Prohibitory	100	93.45

Table 4 Detection rate of algorithms cited in Houben et al. [43]

Algorithm	Prohibition (%)	Warning (%)	Mandatory (%)
HOG+LDA	91.3	90.7	69.2
Hough-like	55.3	65.1	34.7
Viola-Jones	98.8	74.6	67.3

Zaklouta et al. [40] use the histogram of oriented gradients, proposed by Dalal and Triggs [41] for pedestrian detection due to its scale-invariance, local contrast normalization, coarse spatial binning, and weighted gradient orientations. The HOG descriptors are computed and used as a feature to train a linear SVM classifier. To improve the precision of the SVM detector, they use a morphological operator (blackhat) to filter the detected candidates. The blackhat transform is defined as the difference between the closing and the input image. A large part of the image is eliminated using this lter, and the number of false alarms is reduced.

HOG transformer is also used by Wang et al. [42], and they use LDA and SVM as a classifier. The proposed approach achieves high recall and precision ratios in GtSDB dataset, it is robust to bad lighting condition, partial occlusion, low quality, and small projective deformation, but this method is not real time.

In [43], the authors use the German Traffic Sign Detection Benchmark presented as a competition at IJCNN 2013 (International Joint Conference on Neural Networks) to evaluate some of the most popular detection approaches such as the Viola–Jones detector based on Haar features and a linear classifier relying on HOG descriptors, and they also evaluate a recent algorithm exploiting shape and color in a model-based Hough-like voting scheme presented in [44]. The detection rate of the three algorithms is presented in Table 4.

Salti et al. [84] propose an approach based on interest regions extraction rather than sliding window detection. The authors test their approach on ground-truth dataset which contains 6580 images. The dataset is created by using 5 cameras to correctly geo-referenced the signs; however, using 5 cameras is not practical for a real-time application because it increases the execution time (Instead of processing 1 image, 5 images will be processed). They achieved 78.21% for prohibitory signs, 82.13% for danger signs and 72.78% for mandatory signs.

Integral Channel Features detector built on HOG features and boosted decision trees is used by [45], on BTSD dataset they obtained 97.96% for mandatory signs, 97.40% for warning signs and 94.44% for prohibitory signs, on GTSDB dataset they have 96.98% for mandatory signs, 100% for warning signs and 100% for prohibitory signs.

To address the multiclass traffic sign detection (TSD) problem, [48] presented traffic sign localization framework that is capable to detect multiclass traffic sign rapidly in high-resolution image. They proposed three new ideas: first, multi-block normalized local binary pattern (MN-LBP) and tilted MN-LBP (TMN-LBP) are used as discriminant features to express multiclass traffic signs effectively. Second, a tree structure called Split-Flow Cascade is designed. It uses the common features of multiple classes to build a coarse-to-fine pyramid structure named SFC tree. Third, to design an efficient SFC-tree a Common-Finder AdaBoost (CF AdaBoost) is developed to find common features among different training sets. All these new contributions make the system work in real time with high recall, and they have good results on GTSDB dataset: 100% for prohibitory signs, 99.2% for warning signs, 98.57% for mandatory signs, and 97.24% for other signs.

In [47], a good run time has been achieved (6–8 fps on video sequences). The solution presents a novel approach, called Categories-First-Assigned Tree (CFA-Tree) where they integrate the detection and the classification phase in one module, this novel system has high accuracy about 93.5%, and however, this search tree can only detect three categories and has low efficiency in handling high-resolution images [48].

Due to the success of CNN in traffic sign classification, the authors in [77] propose a lightweight and optimized ConvNet with sliding window to detect traffic signs in high-resolution images. The accuracy rate detection achieved is 99.89% on the German Traffic Sign Detection Benchmark dataset. Time execution on GPU (GeForce GTX 980) is 26.506 ms which is equal to processing 3772 frames per s. Obtained results make this approach a real-time application.

Wu et al. [46] use convolutional neural networks CNN to localize and recognize traffic sign, firstly they use support vector machine to transform the original image from RGB to grayscale to avoid falling into the problem of sensitivity to color difference due to various lighting conditions, secondly they use the fixed layer in the CNN to localize region of interest which are similar to traffic sign, and the learnable layers are used to extract discriminant features for classification. They use GTSDB as a dataset and they obtained 99.73% in warning signs and 97.62% in mandatory signs, but it is too far from a real-time application.

Cascaded convolutional neural networks (CNNs) are recently used by [83] to reduce false positive regions detected using the local binary pattern (LBP) feature detector combined with the AdaBoost classifier. Results achieved on GTSDB using an Intel Core 2 Duo 2.2 GHz are illustrated in Table 5.

In Table 7, we resume detection methods based on machine learning. Best accuracies (>99%) are achieved by [84], [45] and [42], and however, can really these methods be a commercial application and have a high rate and precision in ground truth with other complex conditions? However,

Table 5 Results achieved by Zang et al. [83]

Categories	Rate (%)	Time (ms)	False positive
PROHIBITORY	99.45	35	1
MANDATORY	96.50	34	0
DANGER	98.33	35	0

these methods give best results, and they are not real-time application. On the other hand, [82] achieve good results with real-time execution (35 ms). Ordinarily, most learning-based detection methods achieve a detection rate higher than 95%, so it is time to create new datasets more complicated because current dataset is saturated.

2.4 Publicly available detection datasets

- The German Traffic Sign Detection Benchmark—GTSDB dataset [43]: The GTSDB is a single-image traffic sign detection, it consists of 900 images 1360×800 pixels divided into 600 training images and 300 evaluation images, and it is classified into three classes mandatory, warning, and prohibitory. It allows an online evaluation system with immediate analysis and ranking of the submitted results.
- The Belgium Traffic Sign Dataset—BTSD dataset: The BTSD [49]: It consists of more than 10,000 annotations, and images are divided into three categories mandatory, warning, and prohibitory. It contains four video sequences captured in Belgium which can be used for tracking experiments.
- Laboratory for Intelligent and Safe Automobiles—LISA dataset [50]: It contains videos and annotated frames, it is composed of 7855 images containing 47 categories of traffic signs, and only 6610 images are annotated, and the size of images is varied from 640×480 to 1024×522.
- STSD dataset (Swedish Traffic Signs Dataset) [31]: It consists of more than 20,000 images that are created by taking recordings on over 350 km of Swedish highways and city roads, and every fifth frame from the sequence is manually annotated. Sequence of STSD can be used for tracking application.
- DITS data (Data Set of Italian Traffic Signs) [81]: It is a recent dataset generated from 43,289 images extracted from 14h of videos (1280×720 at 10 fps) captured in Italy at date and night time. The detection dataset is composed of 1416 training images and 471 test images, it contains also text file with annotations, and three shape-based super-classes are defined: Prohibitive, Indication, and Warning.

Traffic sign detection datasets are summarized in Table 6.

Table 6 Publicly available traffic sign detection datasets

Dataset	*Images*	Properties
GTSDB [43]	900	Image's size: 1360×800 pixels Country: Germany
BTSD [49]	10,000	Image's size: 1628×1236 pixels Contain 4 videos With annotations Country: Belgium
LISA [50]	7855	Contain videos With annotations Image's size: 640×480 to 1024x522 Country: USA
MASTIF [51]	10,000	Image's size: $720 \times$ pixels Composed of 3 datasets With annotations Country: Croatia
STSD [31]	20,000	Image's size: 1280×960 pixels With annotations Country: Sweden
DITS [81]	1887	With annotation Country: Italy

3 Classification methods

In this section, we highlight some recent and efficient methods in traffic sign classification. Firstly, we describe some methods use hand-crafted features such as HOG, LBP, SIFT, and BRISK; secondly, deep learning methods surpassed the human performance are cited (Table 7).

3.1 Learning methods based on hand-crafted features

Fatin Zaklouta et al. [52] used different size histogram of oriented gradients (HOG) descriptors and Distance Transforms to evaluate the performance of K-d trees and random forests, and the random forests are more robust to variations in the background than the K–d trees. The rate classification achieved for random forest is 97.2% with HOG descriptors and 81.8% with Distance Transform, K–d trees improve 92.9 and 67%, respectively.

Ellahyani et al. [86] calculate the histogram of oriented gradients (HOG) features in the HSI color space; then, it combined with the local self-similarity (LSS) features. The authors prefer to use random forest as a classifier. The recognition rate achieved is 97.43% on the GTSDB and 94.21% for the entire system in 8–10 frame/s.

Authors in [53] compared traffic sign recognition performance of human and machine learning methods; they also show results of a linear classifier trained by linear discriminant analysis (LDA). The performance of LDA was dependent on the feature representation. The authors achieve best results with HOG2 representation by an accuracy of 95.68, 93.18% for HOG1 and 92.34% for HOG3.

The analysis of training data carried out by [62] demonstrates that there is an imbalance in the distribution of samples in the traffic sign classes. The biggest class can contain more

Table 7 Learning-based detection methods

Methods	Dataset used	Prohibition signs	Warning signs	Mandatory signs	All signs	Time
Aghdam et al. [77]	GTSDB	–	–	–	99.89%	26.506 ms
Bar et al. [37]	–	70% with 3,65% false positives	65% with 2,25% FP	60% with 0,95% FP	–	–
Brkic et al. [34]	1000 images	–	90% to 96%	–	–	–
Chen et al. [85]	STSD				80.85%	0.05 to 0.5 s
	GTSDB	100%	100%	99.87%		
	BTSD	97.78%	99.88%	93.45%		
Houben et al. [43]						
HOG+LDA	GTSDB	91.3%	90.7%	69.2%	–	–
Hough-like	GTSDB	55.3%	65.1%	34.7%	–	–
Viola–Jones	GTSDB	98.8%	74.6%	67.3%	–	–
Liu et al. [47]	GTSDB	–	–	–	93.5%	Real time
Liu [48]	GTSDB	100%	92.2%	98.57%	–	192 ms
Mathias et al. [45]	BTSD	94.44%	97.40%	97.96%	–	–
	GTSDB	100%	100%	96.98%	–	–
Salti et al. [84]	Ground-truth	78.21%	82.13%	72.78%	–	–
Wang et al. [42]	GTSDB	99.88%	99.85%	99.85%	–	1.12 to 1.48 s
Zaklouta et al. [40]	820 images contains 11 classes	–	–	–	95%	247 ms
Zang et al. [83]	GTSDB	99.45%	98.33%	96.50%	–	35 ms

than 1000 images while the smallest class can contain only several images. This imbalance can negatively impact on the classification performance; to overcome this problem, the authors proposed a hierarchical classification method for traffic sign recognition, and the classification tree is composed of two layers. In the first layer, the Adaboost classifier combined with Aggregate Channel Features (ACF) is used to classified sings into three categories according to their geometric shape. The ACF is used for feature representation where 10 channels are used (three color channels of RGB color space, the gradient magnitudes, the six oriented gradient maps: horizontal, vertical, 30, 60, 120, and 150), and then, these features are used for training Adaboost classifier. In the second layer, a traffic sign is identified by a random forest classifier which is trained on three features: histogram of oriented gradients (HOG), local binary pattern (LBP), and HSV; they achieved as accuracy 95.97% with GTSRB dataset and 97.94% with STSD dataset.

Yakimov et al. [61] achieved a real-time traffic sign recognition using multithreaded programming technology CUDA on a mobile GPU Nvidia Tegra K1, which contains 192 graphics cores and 4 CPU cores of ARM architecture. The authors proposed a modified generalized Hough transform (GHT) algorithm to classified traffic signs, and they show that the algorithm achieved a good compromise between execution time and accuracy with preprocessed images comparing to result obtained in [43] as it is illustrated in Table 8. The German traffic sign dataset was used, but only 9987 among 50,000 images were taken into account.

LBP and HOG features are used by Li et al. [74] to recognize traffic signs. The accuracy rate achieved is 95.16% for HOG and 95.38% for LBP with GTSRB dataset. Authors in [76] use three categories of feature descriptor HOG, LBP, and Gabor filter as input features to SVM, results obtained are shown in Table 9.

He and Dai [72] propose a new variant of local binary pattern (LBP) named multiscale center symmetric local binary pattern (MS-CSLBP) used as a local feature and the low frequency coefficients of discrete wavelet transform (DWT) as a global feature to classified traffic signs. The great difference between LBP and CSLBP is that the latter replace the center pixel value with the pixel value that is symmetric to

Table 9 Results achieved by Li et al. [74]

approach	Rate (%)	Executing time (ms)
LBP	93.36	1.21
Gabor	93.90	2.31
HOG	94.56	0.02
HOG+LBP	95.24	1.24
HOG + Gabor	97.00	2.40
Gabor + LBP	95.17	3.49
Gabor + LBP + HOG	97.04	3.60

the center pixel which reduce the dimension of feature vectors to $2^{N/2}$ rather than 2^N in LBP as it is illustrated in the following equation:

$$CSLBP_R^N = \sum_{i=0}^{\frac{N}{2}-1} S(g_i - g_{i+p/2})2^i$$

$$S(x) = \begin{cases} 1 & x > \tau \\ 0 & x < \tau \end{cases} \tag{20}$$

A threshold τ is used to reduce the influence of the noise. CSLBP can reduce the dimension of the feature vector and computes very simple compared to HOG and SIFT; however, CSLBP feature of single scale is not enough a discriminate feature to represent traffic signs. To solve the problem, the authors propose to calculate CSLBP in multiscale (radius = 1, 2 and 3). The MS-CSLBP is used then as input feature to classified traffic signs with SVM; the results achieved by [72] with GTSRB dataset are compared with two algorithms of [73] of as it is shown in Table 10

SIFT feature matching algorithm is used by [66], and the authors test their approach on eight videos captured in India. The range of the rate achieved is between 75 to 100% with a false positive does not exceed 2%. Hua et al. [67] also used SIFT combined with the bag of visual to construct codebook then classifying with SVM. The dataset used for evaluation consists of 130 images (50×50 pixels), and they obtained an accuracy rate 93% with time execution which consists of 0.098ms per image. To exploit the intrinsic structure of the

Table 8 Results achieved by Yakimov [61]

Algorithm	Accuracy (%)	time (FPS)
Sliding window + SVM [43]	100	1
GHT with preprocessing [61]	97.3	43
GHT without preprocessing [61]	89.3	25
Viola–Jones [43]	90.81	15
HOG [43]	70.33	20

Table 10 Results achieved by He et al. [72] compared to results of Tang et al. [73]

Method	Rate (%)	Execution time (ms)
HOG [73]	95.92	134
DCT+LBP [73]	92.50	350
DWT+MSCSLBP [72]	97.67	121

pre-learned visual codebook, a new feature approach using group sparse coding is proposed by [71]. The rate of recognition achieved by the proposed approach is 97.83% with GTSRB dataset.

[63] use Speeded Up Robust Feature Descriptor (SURF) with Artificial Neural Network (ANN) classifier to recognize traffic signs. They create a new dataset containing 200 images which have captured from highway roads of Bangladesh in different weather and illumination conditions. The true positive rate achieved is 97% with 3% false positive rate; however, they have good results, the time execution has not mentioned, and their dataset does not contain all traffic sign types.

[64] combine SURF with K-nearest neighbor (K-NN) search method, and they propose a novel feature selection strategy. Not useful interest points are eliminated by applying a threshold of the determinant of Hessian matrix, only interest points with larger determinant are considered. In order to find good matches, they propose a novel feature selection strategy by calculating the first and the second minimum distances. Then, a good match is given as the following:

$$\frac{d_1}{d_2} < t_1 \quad and \quad d_1 < t_2 \tag{21}$$

d_1, d_2: the first and the second minimum distances. t_1, t_2: relative and absolute threshold, respectively.

To evaluate their approach, they create a new dataset including more than 1200 images. They have as total false recognition 12.81% and 0.99% of false classification rate. The proposed approach is not robust to blur, and they obtained a total false recognition 4% after eliminating blur images from the dataset.

Chen et al. [69] use SURF as a feature to classify traffic signs; firstly, they divide template signs into eight categories based on the color and the trained Adaboost classifiers to reduce time processing. Then approximate nearest neighbor (ANN) algorithm is used for matching step. The recognition accuracy achieved is 92.7% with 200 images containing 281 traffic signs.

SURF descriptor is also chosen by [11] due to the efficient of SURF in execution time and robustness to changes in lighting comparing to SIFT and PCA-SIFT. The authors create for each class of signs a template model to eliminate the interest points detected in the background and keep only the points inside the sign. The recognition rate obtained is 97.72% with 48 images.

Hoferlin et al. [70] present an architecture for recognition of circular traffic signs; it consists of two multilayer perceptrons (MLP). The first layer uses SIFT as input feature, and the second layer uses SURF. System performance is evaluated in 30 min sequence which contains 133 traffic signs, and they achieved a rate of 96.4%.

Table 11 Results achieved by Malik et al. [65]

Method	Recognition rate (%)		Matching time(s)
	Scenario 1 (%)	Scenario (%) 2	
SIFT	100	93.75	9.03
SURF	93.75	81.25	6.02
BRISK	93.75	87.5%	4.3

A comparative analysis of three feature matching techniques SIFT, SURF and Binary Robust Invariant Scalable Key (BRISK) points is presented by [65]. The authors create a new dataset containing 172 images classified to 32 categories; after evaluation, they observed that SIFT is outperformed SURF and BRISK. In execution time, however, BRISK is almost twice as fast as SIFT, but it is not in real time. The authors evaluate the system in two different scenarios: in the first scenario, signs are manually segmented, while in the second one, signs are the result of segmentation step. Table 11 shows results obtained by the authors which demonstrate that classification performance depends on results of segmentation step.

Recently, SURF was chosen by [68] because they consider that SURF is a method that achieved the best results in a reasonable time. The accuracy rate achieved is 94.28% with 179 images of the created dataset, and execution time obtained is 0.04 ms for a single SURF keypoint and 5 ms per image. In this approach, objects detected in a similar area of the scene are merged to one object, so that two superimposed signs can be considered as a single sign which can influence the matching performances.

Learning methods based on hand-crafted features are less accurate than ConvNets; however, both methods are not scalable and they cannot classify signs of other region or novel dataset. Due to the scalability of visual attributes, they are used by [89], combined with Bayesian networks which can estimate the probable class of a novel input using the observations. Rate classification achieved for each class of GTSRB is 97.01% for the speed limit, 97.09% for mandatory, 96.31% for danger and other class so in average they achieved 98.04%.

we summarize classification methods based on hand-crafted features in Table 12. However, all methods achieve a good rate classification, and there is not a method with high rate classification and less false positive process in real time it still an open field of research.

3.2 Deep learning methods

The traditional hand-crafted features have a limited representation power and they are strongly related to expert knowledge; consequently, they cannot be discriminative with

Table 12 Traffic sign classification methods based on hand-crafted features

Authors	Dataset used	Feature	Rate (%)
Abedin et al. [63]	200 images	SURF+ANN	97
Aghdam et al. [89]	GTSRB	Visual attributes with Bayesian network	98.04
Ali et al. [11]	48 images	SURF	97.72
Berkaya et al. [76]	GTSRB	GABOR	93.90
		LBP	93.36
		HOG	94.56
		HOG+LBP	95.24
		HOG+GABOR	97.00
		Gabor+LBP	95.17
		HOG+LBP+GABOR	97.04
Chen et al. [69]	200 images	SURF+ANN	92.7
Ellahyani et al. [86]	GTSRB	HOG in HSI+LSS with random forest	97.43
Han et al. [64]	1200 images	SURF+KNN	96
He and Dai [72]	GTSRB	MS-CSLBP+DWT with SVM	97.67
Hoferlin et al. [70]	30 min sequence	SIFT+SURF with MLP	96.4
Houben et al. [43]	GTSRB	Sliding window with SVM	100
		Viola–Jones	90.81
		HOG	70.33
Hua et al. [67]	130 images	SIFT+bag of visual	93
Lasota et al. [68]	179 images	SURF	94.28
Li et al. [74]	GTSRB	HOG	95.16
		LBP	95.38
Liu et al. [71]	GTSRB	Group sparse coding	97.83
Malik et al. [65]	172 images	SIFT	93.75
		SURF	81.25
		BRISK	87.5
Qu et al. [62]	GTSRB	ACF with Adaboost	95.97
	STSD		97.94
Sathish et al. [66]	Videos captured in India	SIFT	75 to 100
Stallkamp et al. [53]	GTSRB	LDA+HOG1	93.18
Tang et al. [73]	GTSRB	HOG	95.92
Yakimov [61]	GTSRB	GHT with preprocessing	97.3
		GHT without preprocessing	89.3

a very large dataset. To overcome this problem and push the recognition performance, deep features are necessary. Pierre Sermanet and Yann Le Cun [54] use Convolutional Networks (ConvNets) to learn invariant features of traffic sign in a supervised way using 32x32 color input images of GTSRB dataset, and they reached an accuracy of 98.97%, which is above then the human performance (98.81%). Moreover, they increased their networks capacity and depth by ignoring color information and they established a new record of 99.17%.

The authors obtained the best result without color information, so they suspected that normalized color channels may be more informative than raw color, and this method is still far from a real-time application.

Fully connected layers in convolutional neural network (CNN) are trained through back-propagation, and the sensitivity of back-propagation and the over training of fully connected layers can make the generalization performance of CNN sub-optimal. Ordinarily, authors use CNN as a fea-

Table 13 Accuracies and time processing obtained by Aghdam et al. [58] compared with results of [60]

Method	Accuracy (%)	CPU time execution (ms)	GPU time execution (ms)
1 ConvNet [58]	–	12.96	1.06
1 compact ConvNet [58]	–	12.47	1.03
1 ConvNet [60]	–	14.47	1.45
5 ConvNet [58]	99.23	64.8	5.30
2 Compact ConvNet [58]	99.61	24.94	2.06
20 ConvNet [60]	99.65	289.4	29.0

ture extractor and a classifier, it is true, and they are getting impressive results, however, with a vast and complex network on a huge dataset. On the other hand, the authors in [57] propose a new approach where CNN works as a deep feature extractor, which means that only the first eight layers are retained and eliminating the fully connected layers. Extreme learning machine (ELM) is used then as a classifier due to its performance generalization, and it receives the input from CNN. The proposed method takes 5–6 h in training without GPU implementation and achieves a recognition rate of 99.40% without any data augmentation and preprocessing like in [56], but this method is not robust to motion blur.

Qian et al. [87] also use CNN as a feature extractor and multilayer perceptron (MLP) as a classifier. Comparing with the classical ConvNet, in the max pooling layer the authors do not use the max values, conversely, their position. The max pooling positions (MPPs) consist to encoded each max value position to a binary value of 4 bits, then concatenating all max positions values to obtain the MPPs feature. The accuracy achieved using MPPs is boosted to 98.86% on GTSRB.

Xie et al. [88] observe that 80% of misclassified signs have the same color, shape, and pictogram. To overcome this problem, they propose a two-stage cascaded CNN. The first stage CNN is trained on the class label, while the second stage is trained on super-classes separately according to the shape and pictogram. The accuracy of the proposed method is 97.94% on GTSRB, and they decrease the misclassified number with the cascaded CNN from 430 to 202. The time execution is not mentioned.

A new ConvNet architecture proposed by [58] can reduce the number of parameters 27, 22 and 3% compared with ConvNets used by [54,59,60], respectively; then they propose a compact ConvNet which reduces the number of parameters 52% compared to the ConvNet proposed. To improve the accuracy of classification, they also proposed a new method for creating an optimal ConvNets by selecting an optimal number of ConvNets with the highest possible accuracy and with less number of arithmetic operations 88 and 73% compared with [59,60], respectively. The accuracy achieved by their proposed method on GTSRB dataset is 99.23% with

2 ConvNets (compact ConvNet) and 99.61% with only 5 ConvNets which greatly reduces the execution time as it is illustrated in Table 13. On the other hand, [54] have as accuracy 99.46% with an ensemble of 25 ConvNets, and [60] achieve an accuracy 99.65% using 20 ConvNets. To test the scalability and the cross-dataset performance of the new ConvNet proposed by [58], they use the ConvNet already trained on GTSRB to identify the traffic signs in the BTSC dataset, and the accuracy obtained is 92.12%.

A simple architecture of deep neural network is used by [80] to recognize circular traffic signs, and they achieved a recognition rate of 97.5% on GTSRB. Two other architectures of CNN are proposed by [81], and single-scale architecture consists of two stages of convolutional layers and two local fully connected layers followed by a softmax classifier. In multiscale architecture, the output of the first convolutional layer is considered as input for both the second convolutional layer and the first fully connected one. The two architectures are evaluated on GTSRB dataset and their novel dataset DITS (Data Set of Italian Traffic Signs). The accuracy rates achieved on GTSRB are 97.2% for single-scale architecture and 98.2% for the multiscale one, and for DITS dataset the accuracy rate achieved is 93.1% for single scale and 95.0% for multiscale.

Dan Cirean et al. [55] used a GPU implementation of a convolutional neural network and they applied preprocessing on input images by resizing all images to 48x48 pixels and they test three types of normalization to overcome high contrast variation and they obtain a recognition rate of 99.15% by using a committee of multilayer perceptrons (MLPs) trained with HOG feature descriptors and CNNs trained on raw pixel intensities to boost recognition performance. After they won the final phase of the German traffic sign recognition benchmark by a recognition rate 99.46% in [56], they use a multi-column deep neural networks (MCDNN) by averaging the output activations of several DNN columns.

Aghdam et al. [78] propose a new architecture of CNN which sets a new record of classification accuracy 99.51%. The proposed architecture reduces 85% of the number of the parameters and 88% of multiplications comparing to the winner network reported in the German Traffic Sign Benchmark

Table 14 Deep learning classification methods

Authors	Dataset used	Method	Rate (%)
Aghdam et al. [58]	GTSRB	New ConvNet	99.23
		Compact ConvNet	99.61
	BTSC	ConvNet trained on GTSRB	92.12
Aghdam et al. [78]	GTSRB	New CNN	99.51
Aghdam et al. [77]	GTSRB	Single CNN	99.55
		3 CNN	99.70
Cirean et al. [56]	GTSRB	MCDNN	99.46
Cirean et al. [55]	GTSRB	MLPs+HOG+CNNs	99.15
Eickeler [80]	GTSRB	New DNN	97.5
Jin [60]	GTSRB	20 ConvNet	99.65
Qian et al. [87]	GTSRB	CNN+MLP	98.86
Qu et al. [62]	GTSRB	ACF with Adaboost	95.97
	STSD		97.94
Sermanet et al. [54]	GTSRB	Convnets	98.97
		Human performance	98.8
		Convnets without color	99.17
Xie et al. [88]	GTSRB	Cascaded CNN	97.94
Youssef [81]	GTSRB	Single-scale CNN	97.2
Zeng et al. [57]	GTSRB	CNN+ELM	99.40

Table 15 Available classification datasets

Dataset	Classes	Images	Properties
GTSRB [53]	43	50000	Image's size: 15×15 to 222×193 pixels Country: Germany
BTSC [49]	62	6000	Image's size: NA Country: Belgium
Revised MASTIF [51]	30	6000	Image's size: 64×64 pixels Country: Croatia
DITS [81]	58	9254	Country: Italy

competition [56]. This reduction is achieved by using Leaky Rectified Linear Units (ReLU) [79] as an activation function that use only one comparison and one multiplication to compute the output.

A new record of traffic sign classification is achieved again by Aghdam et al. [77], and they propose a new variant of their previous CNN proposed in [78]. The authors replace color images with that grayscale, and they remove the linear transformation layer. And to increase the flexibility a fully connected layer is added to the network and they reduce the size of the first and the middle kernels also the input images are resized to 44×44 pixels to reduce time processing. The new best accuracy achieved is 99.55% with single CNN and 99.70% with an ensemble of 3 CNN. The new CNN is real time with time processing of 0.7 ms per image.

In Table 14, we summarize deep learning methods.

3.3 Publicly available classification datasets

- German Traffic Sign Recognition Benchmark—GTSRB dataset [53]: The GTSRB traffic sign classification dataset is composed of 43 classes containing more than 50000 images. The minimum of traffic signs in each class is 9 traffic signs with a size varied between 15×15 to 222×193 pixels. To allow searchers who do not have a background in image processing to participate in the competition, three pre-calculated features are provided: three configuration of HOG features (HOG1, HOG2, HOG3), Haar-like features and hue histograms (Table 15).
- Belgium Traffic Sign Classification–BTSC dataset [49]: The BTSC dataset is just an extraction of regions of interest containing traffic sign in BTSD dataset, and it is composed of more than 4000 training images classified in 62 classes and more than 2000 testing images.
- Revised MASTIF dataset: The revised MASTIF [51]: It is a subset of MASTIF dataset obtained by extracting traffic sign examples from regions of interest in MASTIF dataset. It is composed of 4028 training images and 1644 testing images. Images are distributed on 30 classes.
- DITS data (Data Set of Italian Traffic Signs) [81]: The classification subset is composed of 8048 images for

training and 1206 for testing. It contains in total 58 classes of signs with different sizes.

4 Future research directions

The problem with the current state-of-art is the lack of ground truth and universal dataset that contains sign from the different region (including regions not adhering to The Vienna Convention on road signs and signals) and captured in different conditions. Since available datasets reached saturation, we regard that a new universal dataset and more complex is necessary. Another problem of the datasets currently available is the imbalance in the distribution of samples in the traffic sign class which can negatively impact on the classification performance; to overcome this problem, a new balanced dataset is required.

We suggest for future research using the high-resolution image in detection dataset, if we consider that a speed of the car is 100 km/h and there is a sign away from the car about 27 m; therefore, this sign will be passed in 1 s. For this reason, dataset's images should be in high resolution to clearly detect distant signs. Researchers can focus more on tracking module to follow signs for the reason that if the system uses a camera with 30 shot frames per s, it should be able to detect and recognize all signs of 30 frames in 1 s; however, with tracking module, sings detected and tracked will not be reclassified in each frame captured but only once until the appearance of a new signs.

Traffic sign recognition systems are composed of detection and classification stages. Since classification's performance depends on detection results, this makes the determination of the best solution a tedious process. As it is demonstrated in the introduction, the core of the problem is not detecting traffic signs with high recall; however, obtaining a high precision is more crucial. We suggest focusing research on how decreasing false alarms in traffic sign detection.

To enhance the accuracy of classification stage, researchers should focus more on searching and analyzing more discriminant features which can better represent different classes of traffic signs. Currently, deep features are more discriminant than hand-crafted one; nevertheless, there are no studies on learning methods, proving their scalability for new dataset which can pave way for future research.

5 Conclusion

In this paper, we have presented an overview of some recent and efficient traffic sign detection and classification methods. Detection methods are divided into three categories: color based that are classified according to the color space, shape based, and learning based that include deep learning methods. The recent detection methods achieve a detection rate varied from 90 to 100% with available dataset described briefly in the paper. Nevertheless, it is arduous to decide which method is the best.

To obtain a high classification rate, it is necessary to adopt discriminative features and a powerful classifier. Acceptable results achieved by learning methods using hand-crafted features, and furthermore, classification methods performances boosted with deep learning methods such as CNN and they achieved a high accuracy rate $> 99\%$. Since available datasets reached saturation, a new universal dataset more complex is indispensable.

However, detection and classification methods achieved a high accuracy rate, and they are still far from a real-time ADAS application where the sign should be detected and classified in real time.

The imposed question is: Can recent traffic sign detection and classification methods prove the same performances in real-world application or with other ground truth datasets? Can they prove with smart phone devices the same real-time execution obtained under cpu and gpu environment? Finally, a universal traffic sign recognition system is still an open field of research.

References

1. Gudigar A, Chokkadi S, Raghavendra U (2016) A review on automatic detection and recognition of traffic sign. Multimed Tools Appl 75(1):333–364
2. Fu MY, Huang YS (2010) A survey of traffic sign recognition. In: International conference on wavelet analysis and pattern recognition (ICWAPR), 2010, IEEE, pp 119–124
3. Wali SB, Hannan MA, Hussain A, Samad SA (2015) Comparative survey on traffic sign detection and recognition: a review. Przegld Elektrotechniczny, ISSN: 0033-2097
4. Escalera S, Bar X, Pujol O, Vitri J, Radeva P (2011) Background on traffic sign detection and recognition. In: Traffic-sign recognition systems. Springer, London, pp 5–13
5. De La Escalera A, Moreno LE, Salichs MA, Armingol JM (1997) Road traffic sign detection and classification. IEEE Trans Indus Electron 44(6):848–859
6. Benallal M, Meunier J (2003) Real-time color segmentation of road signs. In: Canadian conference on electrical and computer engineering, vol 3, 2003, IEEE CCECE 2003, IEEE, pp 1823–1826
7. Ruta A, Li Y, Liu X (2010) Real-time traffic sign recognition from video by class-specific discriminative features. Pattern Recogn 43(1):416–430
8. Ruta A, Porikli F, Watanabe S, Li Y (2011) In-vehicle camera traffic sign detection and recognition. Mach Vis Appl 22(2):359–375
9. Lim KH, Ang LM, Seng KP (2009) New hybrid technique for traffic sign recognition. In: International symposium on intelligent signal processing and communications systems, 2008. ISPACS 2008, IEEE, pp 1–4
10. Itti L, Koch C, Niebur E (1998) A model of saliency-based visual attention for rapid scene analysis. IEEE Trans Pattern Anal Mach Intell 20(11):1254–1259
11. Behloul A, Saadna Y (2014) A Fast and Robust Traffic Sign Recognition. Int J Innov Appl Stud 5(2):139
12. Yakimov P (2015) Traffic signs detection using tracking with prediction. In: International conference on E-business and telecommunications Colmar, France, Springer, pp 454–467

13. Yakimov PY (2015) Preprocessing digital images for quickly and reliably detecting road signs. Pattern Recogn Image Anal 25(4):729–732

14. Wang G, Ren G, Jiang L, Quan T (2014) Hole-based traffic sign detection method for traffic signs with red rim. Vis Comput 30(5):539–551

15. Mogelmose A, Trivedi MM, Moeslund TB (2012) Vision-based traffic sign detection and analysis for intelligent driver assistance systems: Perspectives and survey. IEEE Trans Intell Transp Syst 13(4):1484–1497

16. Maldonado-Bascon S, Lafuente-Arroyo S, Gil-Jimenez P, Gomez-Moreno H, Lpez-Ferreras F (2007) Road-sign detection and recognition based on support vector machines. IEEE Trans Intell Transp Syst 8(2):264–278

17. Fleyeh H (2006) Shadow and highlight invariant color segmentation algorithm for traffic signs. In: IEEE conference on cybernetics and intelligent systems, 2006, IEEE, pp 1–7

18. Vitabile S, Pollaccia G, Pilato G, Sorbello F (2001) Road signs recognition using a dynamic pixel aggregation technique in the HSV color space. In: Proceedings of the 11th international conference on image analysis and processing, 2001, IEEE, pp 572–577

19. De la Escalera A, Armingol JM, Mata M (2003) Traffic sign recognition and analysis for intelligent vehicles. Image Vis Comput 21(3):247–258

20. Fang CY, Fuh CS, Yen PS, Cherng S, Chen SW (2004) An automatic road sign recognition system based on a computational model of human recognition processing. Comput Vis Image Underst 96(2):237–268

21. Miura J, Kanda T, Nakatani S, Shirai Y (2002) An active vision system for on-line traffic sign recognition. IEICE TRANSACTIONS on Inf Syst 85(11):1784–1792

22. Broggi A, Cerri P, Medici P, Porta PP, Ghisio G (2007) Real time road signs recognition. In: IEEE intelligent vehicles symposium, 2007, IEEE, pp 981–986

23. Gmez-Moreno H, Maldonado-Bascn S, Gil-Jimnez P, Lafuente-Arroyo S (2010) Goal evaluation of segmentation algorithms for traffic sign recognition. IEEE Trans Intell Transp Syst 11(4):917–930

24. Liu C, Chang F, Chen Z, Liu D (2016) Fast traffic sign recognition via high-contrast region extraction and extended sparse representation. IEEE Trans Intell Transp Syst 17(1):79–92

25. Hechri A, Hmida R, Mtibaa A (2015) Robust road lanes and traffic signs recognition for driver assistance system. Int J Comput Sci Eng 10(1–2):202–209

26. Garcia-Garrido MA, Sotelo MA, Martin-Gorostiza E (2006) Fast traffic sign detection and recognition under changing lighting conditions. In: IEEE intelligent transportation systems conference, 2006 ITSC'06, IEEE, pp. 811–816

27. Barnes N, Zelinsky A (2004) Real-time radial symmetry for speed sign detection. In: IEEE intelligent vehicles symposium, 2004, IEEE, pp 566–571

28. Loy G, Barnes N (2004) Fast shape-based road sign detection for a driver assistance system. In: Proceedings of the 2004 IEEE/RSJ international conference on intelligent robots and systems, 2004 (IROS 2004) Vol 1, IEEE, pp 70-75

29. Soheilian B, Arlicot A, Paparoditis N (2010) Extraction de panneaux de signalisation routire dans des images couleurs. In: Reconnaissance des Formes et Intelligence Artificielle, pp 1–8

30. Zhang SC, Liu ZQ (2005) A robust, real-time ellipse detector. Pattern Recogn 38(2):273–287

31. Larsson F, Felsberg M (2011) Using Fourier descriptors and spatial models for traffic sign recognition. In: Scandinavian conference on image analysis, Springer, Berlin, Heidelberg, pp 238–249

32. Qin F, Fang B, Zhao H (2010) Traffic sign segmentation and recognition in scene images. In: Chinese conference on pattern recognition (CCPR), pp 1–5

33. Viola P, Jones M (2001) Rapid object detection using a boosted cascade of simple features. In: Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition, CVPR 2001, vol 1, IEEE, p I

34. Brkic K, Pinz A, Šegvic S (2009) Traffic sign detection as a component of an automated traffic infrastructure inventory system. Stainz, Austria

35. Brki K, Šegvi S, Kalafati Z, Sikiri I, Pinz A (2010) Generative modeling of spatio-temporal traffic sign trajectories. In: IEEE computer society conference on computer vision and pattern recognition workshops (CVPRW), 2010, IEEE, pp 25–31

36. Brkic K (2010) An overview of traffic sign detection methods. Department of Electronics, Microelectronics, Computer and Intelligent Systems Faculty of Electrical Engineering and Computing Unska, 3, 10000

37. Bar X, Escalera S, Vitri J, Pujol O, Radeva P (2009) Traffic sign recognition using evolutionary adaboost detection and forest-ECOC classification. IEEE Trans Intell Transp Syst 10(1):113–126

38. Prisacariu VA, Timofte R, Zimmermann K, Reid I, Van Gool L (2010) Integrating object detection with 3d tracking towards a better driver assistance system. In: 20th International conference on pattern recognition (ICPR), 2010, IEEE, pp 3344–3347

39. Fang CY, Chen SW, Fuh CS (2003) Road-sign detection and tracking. IEEE Trans Veh Technol 52(5):1329–1341

40. Zaklouta F, Stanciulescu B (2011) Warning traffic sign recognition using a HOG-based Kd tree. In: IEEE intelligent vehicles symposium (IV), 2011. IEEE, pp 1019–1024

41. Dalal N, Triggs B (2005) Histograms of oriented gradients for human detection. In: IEEE computer society conference on computer vision and pattern recognition, CVPR 2005, vol 1, IEEE pp 886–893

42. Wang G, Ren G, Wu Z, Zhao Y, Jiang L (2013) A robust, coarse-to-fine traffic sign detection method. In: The 2013 international joint conference on neural networks (IJCNN), IEEE, pp 1–5

43. Houben S, Stallkamp J, Salmen J, Schlipsing M, Igel C (2013) Detection of traffic signs in real-world images: The German Traffic Sign Detection Benchmark. In: The 2013 international joint conference on neural networks (IJCNN), IEEE, pp 1–8

44. Houben S (2011) A single target voting scheme for traffic sign detection. In: IEEE intelligent vehicles symposium (IV), 2011, IEEE, pp 124–129

45. Mathias M, Timofte R, Benenson R, Van Gool L (2013) Traffic sign recognitionHow far are we from the solution?. In: The 2013 international joint conference on neural networks (IJCNN), IEEE, pp 1–8

46. Wu Y, Liu Y, Li J, Liu H, Hu X (2013) Traffic sign detection based on convolutional neural networks. In: The 2013 international joint conference on neural networks (IJCNN), pp 1–7, IEEE

47. Liu C, Chang F, Chen Z, Li S (2013) Rapid traffic sign detection and classification using categories-first-assigned tree. J Comput Inf Syst 9(18):7461–7468

48. Liu C, Chang F, Chen Z (2014) Rapid multiclass traffic sign detection in high-resolution images. IEEE Trans Intell Transp Syst 15(6):2394–2403

49. Timofte R, Zimmermann K, Van Gool L (2009) Multi-view traffic sign detection, recognition, and 3d localisation. In: Workshop on applications of computer vision (WACV), 2009, IEEE, pp 1–8

50. Mogelmose A, Trivedi MM, Moeslund TB (2012) Learning to detect traffic signs: comparative evaluation of synthetic and real-world datasets. In: 21st International conference on pattern recognition (ICPR), 2012, IEEE, pp 3452–3455

51. Šegvic S, Brki K, Kalafati Z, Stanisavljevi V, evrovi M, Budimir D, Dadi I (2010) A computer vision assisted geoinformation inventory for traffic infrastructure. In: 13th International IEEE conference on intelligent transportation systems (ITSC), 2010, IEEE, pp 66–73

52. Zaklouta F, Stanciulescu, B., Hamdoun, O. (2011) Traffic sign classification using kd trees and random forests. In: The 2011 international joint conference on neural networks (IJCNN), pp 2151–2155, IEEE

53. Stallkamp J, Schlipsing M, Salmen J, Igel C (2012) Man vs. computer: benchmarking machine learning algorithms for traffic sign recognition. Neural networks 32:323–332

54. Sermanet P, LeCun Y (2011) Traffic sign recognition with multi-scale convolutional networks. In The 2011 international joint conference on neural networks (IJCNN), IEEE, pp 2809–2813

55. Cirean D, Meier U, Masci J, Schmidhuber J (2011) A committee of neural networks for traffic sign classification. In: The 2011 international joint conference on neural networks (IJCNN), IEEE, pp 1918–1921

56. CireAn D, Meier U, Masci J, Schmidhuber J (2012) Multi-column deep neural network for traffic sign classification. Neural Netw 32:333–338

57. Zeng Y, Xu X, Fang Y, Zhao K (2015) Traffic sign recognition using extreme learning classifier with deep convolutional features. In: The 2015 international conference on intelligence science and big data engineering (IScIDE 2015), Suzhou, China

58. Aghdam HH, Heravi EJ, Puig D (2017) A practical and highly optimized convolutional neural network for classifying traffic signs in real-time. Int J Comput Vis 122(2):246–269

59. Ciregan D, Meier U, Schmidhuber J (2012) Multi-column deep neural networks for image classification. In IEEE conference on computer vision and pattern recognition (CVPR), 2012, IEEE, pp 3642–3649

60. Jin J, Fu K, Zhang C (2014) Traffic sign recognition with hinge loss trained convolutional neural networks. IEEE Trans Intell Transp Syst 15(5):1991–2000

61. Yakimov PY (2016) Real-time road signs recognition using mobile GPU. In: CEUR workshop proceedings, vol 1638, pp 477–483

62. Qu Y, Yang S, Wu W, Lin L (2016) Hierarchical traffic sign recognition. In: Pacific rim conference on multimedia, Springer, pp 200–209

63. Abedin MZ, Dhar P, Deb K (2016) Traffic Sign Recognition using SURF: speeded up robust feature descriptor and artificial neural network classifier. In: 9th International conference on electrical and computer engineering (ICECE), 2016, IEEE, pp 198–201

64. Han Y, Virupakshappa K, Oruklu E (2015) Robust traffic sign recognition with feature extraction and k-NN classification methods. In: IEEE international conference on electro/information technology (EIT), 2015, IEEE, pp 484–488

65. Malik Z, Siddiqi I (2014) Detection and recognition of traffic signs from road scene images. In: 12th International conference on frontiers of information technology (FIT), 2014, IEEE, pp 330–335

66. Sathish P, Bharathi D (2016) Automatic road sign detection and recognition based on SIFT feature matching algorithm. In: Proceedings of the international conference on soft computing systems. Springer India, pp 421–431

67. Hua X, Zhua X, Lia D, Li H (2010) Traffic sign recognition using Scale invariant feature transform and SVM. In: A special joint symposium of ISPRS technical commission IV and AutoCarto in conjunction with ASPRS/CaGIS fall specialty conference November, pp 15–19

68. Lasota M, Skoczylas M (2016) Recognition of multiple traffic signs using keypoints feature detectors. In: International conference and exposition on electrical and power engineering (EPE), 2016, IEEE, pp 535–540

69. Chen L, Li Q, Li M, Mao Q (2011) Traffic sign detection and recognition for intelligent vehicle. In: IEEE intelligent vehicles symposium (IV), 2011, IEEE, pp 908–913

70. Hoferlin B, Zimmermann K (2009) Towards reliable traffic sign recognition. In: IEEE Intelligent vehicles symposium, 2009, IEEE, pp 324–329

71. Liu H, Liu Y, Sun F (2014) Traffic sign recognition using group sparse coding. Inf Sci 266:75–89

72. He X, Dai B (2016) A new traffic signs classification approach based on local and global features extraction. In: International conference on information communication and management (ICICM), IEEE, pp 121–125

73. Tang S, Huang LL (2013) Traffic sign recognition using complementary features. In: 2nd IAPR Asian conference on pattern recognition (ACPR), 2013, IEEE, pp 210–214

74. Li C, Yang C (2016) The research on traffic sign recognition based on deep learning. In: 16th International symposium on communications and information technologies (ISCIT), 2016, IEEE, pp 156–161

75. Akinlar C, Topal C (2013) EDCircles: a real-time circle detector with a false detection control. Pattern Recogn 46(3):725–740

76. Berkaya SK, Gunduz H, Ozsen O, Akinlar C, Gunal S (2016) On circular traffic sign detection and recognition. Expert Syst Appl 48:67–75

77. Aghdam HH, Heravi EJ, Puig D (2016) A practical approach for detection and classification of traffic signs using Convolutional Neural Networks. Robot Auton Syst 84:97–112

78. Aghdam HH, Heravi EJ, Puig D (2016) Recognizing traffic signs using a practical deep neural network. In: Robot 2015: 2nd Iberian robotics conference, Springer, pp 399–410

79. Maas AL, Hannun AY, Ng AY (2013) Rectifier nonlinearities improve neural network acoustic models. In: Proceedings of the ICML, vol 30, No 1

80. Eickeler S, Valdenegro M, Werner T, Kieninger M (2016) Future computer vision algorithms for traffic sign recognition systems. In: Advanced microsystems for automotive applications 2015, Springer, pp 69–77

81. Youssef A, Albani D, Nardi D, Bloisi DD (2016) Fast traffic sign recognition using color segmentation and deep convolutional networks. In: International conference on advanced concepts for intelligent vision systems, Springer, pp 205–216

82. Huang Z, Yu Y, Gu J, Liu H (2016) An efficient method for traffic sign recognition based on extreme learning machine. IEEE Trans Cybern 47(4):920–933

83. Zang D, Zhang J, Zhang D, Bao M, Cheng J, Tang K (2016) Traffic sign detection based on cascaded convolutional neural networks. In: 17th IEEE/ACIS international conference on software engineering, artificial intelligence, networking and parallel/distributed computing (SNPD), 2016, IEEE, pp 201–206

84. Salti S, Petrelli A, Tombari F, Fioraio N, Di Stefano L (2015) Traffic sign detection via interest region extraction. Pattern Recogn 48(4):1039–1049

85. Chen T, Lu S (2016) Accurate and efficient traffic sign detection using discriminative adaboost and support vector regression. IEEE Trans Veh Technol 65(6):4006–4015

86. Ellahyani A, El Ansari M, El Jaafari I (2016) Traffic sign detection and recognition based on random forests. Appl Soft Comput 46:805–815

87. Qian R, Yue Y, Coenen F, Zhang B (2016) Traffic sign recognition with convolutional neural network based on max pooling positions. In: 12th International conference on natural computation, fuzzy systems and knowledge discovery (ICNC-FSKD), 2016, IEEE, pp 578–582

88. Xie K, Ge S, Ye Q, Luo Z (2016) Traffic sign recognition based on attribute-refinement cascaded convolutional neural networks. In: Pacific rim conference on multimedia, Springer, pp 201–210

89. Aghdam HH, Heravi EJ, Puig D (2015) Traffic sign recognition using visual attributes and Bayesian network. In: International joint conference on computer vision, imaging and computer graphics, Springer, pp 295–315