



An effective automatic traffic sign classification and recognition deep convolutional networks

Jayant Mishra¹ · Sachin Goyal¹

Received: 5 April 2021 / Revised: 14 July 2021 / Accepted: 25 January 2022 /
Published online: 10 March 2022

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract

Automatic traffic sign recognition is essential for autonomous driving, assisted driving, and driver safety. Convolutional neural networks (CNNs) are the most widely used deep learning algorithm for traffic signs recognition. This research presents an effective technique for automatically recognizing traffic signs images. This technique mainly uses four GTSRB, GTSDDB, BTSC, and TSRD standard traffic signs images datasets of different traffic sign images and provides the best result using our CNN model architecture. It helps to assist the driver in driving the motor vehicle safely. Drivers devote too much attention and effort to recognizing traffic signs by manually analyzing and recognizing their aspects. This study presents an automatic traffic sign recognition system to minimize motor vehicle accidents using representation to identify signs-this task utilizing a deep convolutional neural network. Here, our work presents a novel CNN architecture (.001) with Adam optimizer, batch size 128, and multi-interconnect layers to improve the performance of traffic signs detection. A Convolutional Neural Network (CNN) achieved more accuracy, results based on a complex network. Our model learns from the GTSDDB dataset, which contains 43 traffic classes, and uses this information to predict the proper class of an anonymous traffic sign with 99.81% accuracy and minimum losses. Contradictory, the result is improved than the earlier study, which examined 98.20% accuracy that the approach can still detect traffic signs with extreme weather conditions and blur image conditions.

Keywords Automatic traffic sign recognition · Convolutional neural networks (CNN) · Deep learning algorithms · GTSRB, GTSDDB, BTSC, and TSRD · Standard traffic sign image datasets

✉ Jayant Mishra
rgtu.jayant@gmail.com

Sachin Goyal
sachingoyal@rgtu.net

¹ UIT-RGPV, Bhopal, MP, India

1 Introduction

Traffic signs use images, characters, and shapes with meaningful colors to indicate set, prohibitive and dangerous. While driving a motor vehicle, it is typical for most drivers to misinterpreted and misread traffic signs. In terms of speed, the intelligent vehicle also misinterpreted traffic signs which cause a major accident [19]. In terms of the survey, more than 1.24 million deaths and critical injuries from vehicle accidents are interpreted every year in different age groups and vehicles due to the reflection of light, lack of attention, unknown problems, and a lot of further reasons for terrible accidents. Cameras detect most objects in automated driving, including traffic signals, routes, individual vehicles, and traffic police gestures. Temporarily, radar is inherently incapable of identifying signs such as speed limits and stop signs. Many autonomous vehicles and driver assistance systems have cameras mounted on their dashboards. Cameras can collect real-time traffic sign images or videos integrated into the vehicle system for machine learning models [39]. The model's deep learning method must be reliable for recording traffic signs in different angles and positions. Even so, the vehicle's speeds and geographic position are constantly changing as it passes through various backgrounds and lights [11]. The camera mounted on the car captures images of traffic signals in their natural environment. While deep neural networks perform well in traffic signal detection experiments, real applications still suffer time and space constraints [50]. In an image, its distance from the camera defines the image size. The identification of traffic signs is most important for the responses and safety of intelligent vehicles. The limited scale of the signs, on the other hand, significant difficulty in traffic signs recognition. In comparison, identifying traffic signs is more important for accuracy than the label of traffic signs [7, 21]. Convolutional Neural Networks (CNN) [22] is a form of deep neural network that can learn more discriminative features and is close to the visual processing of human vision [1, 52]. Compared to the current best-performing approaches, CNN performs better in traffic sign identification algorithms [31]. For traffic sign classification, traditional research methods like simple machine learning models and support vector machines [27, 49, 50]. However, if traffic signs are blurred, the identification rate of traditional methods can decrease (Figs. 1 and 2).

The German traffic sign database, which consists of the German Traffic Sign Recognition Benchmark (GTSRB) and the German Traffic Sign Detection Benchmark (GTSDB) [40], is frequently used to research traffic sign detection and classification. GTSRB and GTSDB, conversely, are not required to represent real-world driving circumstances, as GTSRB sign images consist of a considerable part of the image. Compared to GTSDB signs, real-world traffic sign images frequently have a smaller image area [39].

This paper suggests a multi-neural network strategy for identifying traffic signs using deep learning-our proposed methodology is a newly constructed neural network architecture for successful traffic sign recognition. CNN architecture has been designed to detect traffic signs. Here different convolutional layers, pooling layers, max pool, and dropout layers are defined that show the novelty of our work. Furthermore, the proposed model is also applied in the different standard datasets such as GTSRB, GTSDB [40], BTSC [51], and TSRD datasets. Finally, analyzes the result in terms of precision, recall, and f1 score representing the best accuracy under the validation of datasets. Finally, our experimental evaluation demonstrates that the dataset's traffic sign identification has significantly greater accuracy efficiency. Here, Adam optimizer is applied [5] to optimize the overall training objective and set $lr = 0.001$ (Table 1).

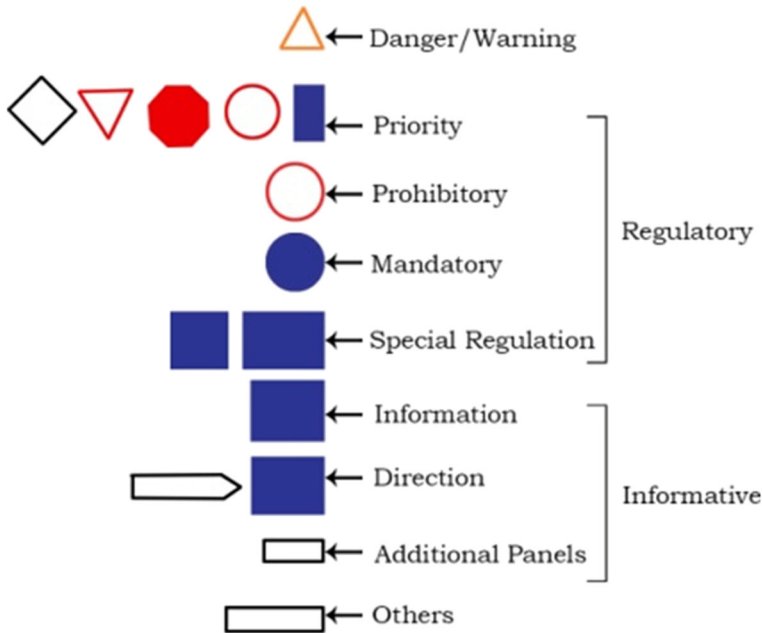


Fig. 1 Traffic Sign categories [34]

2 Related works

S. He et al. [11] proposed CapsNet for the detection of different traffic signs. Here study addresses the misunderstanding issues caused by CNN’s use in traffic sign identification and the loss of maximum pooling. Visual characteristics, position information such as the location and shape of the input image, and the spatial relationship are all kept. The network has the advantage of being robust and generalizable. This study uses CapsNets in traffic sign identification, emerging in deep learning for analyzing traffic scenes. In this article, CapsNet solves traffic sign recognition by modifying the parameters and weights. In 2021, Ghosh R. [9] presents a method for detecting and monitoring on-road vehicles in various weather circumstances by utilizes many Faster R-CNN region proposal networks (RPNs). The application of several RPNs in Faster RCNN relatively unexplored area of research in this field. Since the typical Faster R-CNN produces regions of interest (ROIs) with a single fixed RPN, it could not distinguish vehicles of different sizes. In comparison, the existing study provides an end-to-end method for detecting on-road cars that generates ROIs using numerous varying-sized RPNs and thereby detects various vehicle sizes. The suggested approach is novel in which it

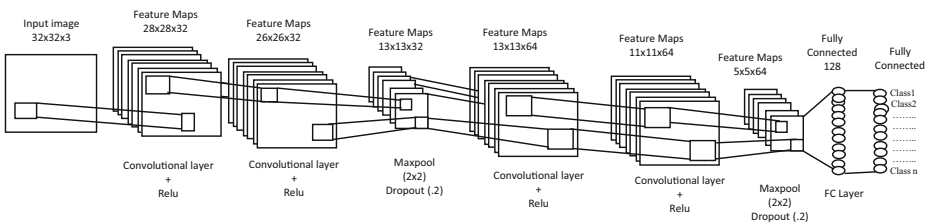


Fig. 2 Design of our CNN Model

Table 1 Classes of the traffic signs

Categories	Subcategories
Danger Sign	42 classes
Prohibitory Sign [23]	42 classes
Mandatory Sign [23, 24]	29 classes
Wayfinding Sign	62 classes
Tourist Information Signs	17 classes
Security Sign	26 classes
Auxiliary Sign	—

includes many RPNs with varying sizes into a standard Faster R-CNN. Three distinct public datasets to evaluate the proposed system's performance: DAWN, CDNet 2014, and LISA. The DAWN, CDNet 2014, and LISA datasets attained an average precision of 89.48%, 91.20%, and 95.16%. A study conducted by Gupta H. et al. [10] developed an algorithm that automatically differentiates various kinds of vehicles in aerial footage using deep learning techniques. UAV dataset has unbalanced classes. This work added 500 images to the cleaned and preprocessed dataset using web mining. Faster R-CNN, SSD, YOLOv3, and YOLOv4 were the four most unique object identification algorithms with enhanced generalization capabilities and flexibility in real-world scenarios. The traffic recognition model constructed with YOLOv4 does better than by at least 88%, 13%, and 25%, respectively. Dewi C. et al. [6] proposed Spatial Pyramid Pooling (SPP) that used the Yolo V3 model. SPP considered input images of various dimensions during training. The experimental results indicated that SPP improves recognition of Taiwan's prohibitory sign. All YOLO V3 models evaluated using mAP.

Y. Wu et al. [46] provided two-level detection architecture location modules (RPM) and the classification module (CM) that seeks to categorize objects in this article. Additionally, the model also demonstrated that the data augmentation technique is based on the logo addresses missing categories efficiently. Fu J. et al. [7] suggested an object detection approach based on detecting a specific vehicle and its wheels. Due to the standard SSD's incompetence in differentiating small objects, completing vehicle and wheel detection tasks offers problems. To address this issue, they develop a new benchmark dataset with five separate categories and propose a novel SSD-based approach that uses many multi-concatenation modules and SEBlocks to enhance recognition accuracy for small objects. The proposed technique illustrates through trials on the Pascal VOC2007 dataset, the KITTI dataset, and the benchmark dataset. Using visualization tests on the Pascal VOC2007 exam and wheel detection, they demonstrate that MSSD extensively boosts small object recognition. Furthermore, it is extensible to different models with a large number of prediction layers. According to Said Y. et al. [30], identify a country's flags in the absence of a localization requirement. The suggested local context network design created area suggestions by vertically expanding tested anchors to encompass the flag under challenging conditions such as distortion and occlusion. The projected collection will include 20,000 images of actual flags from 200 countries-the proposed methods mean average precision of 89.5%.

The classification of traffic signs [43] examine for decades, and the German Traffic Sign Recognition (GTSRB) [28, 35, 45] advocated that an analogous study as well. In this article, they proposed the DC-GCNN model. The performance of DCGCNN is compared to the standard deep structure on ten separate datasets in the first experiment set. On the same

convolutional network, DC-GCNN improves DCNN using conventional fully connected techniques. On application datasets, DC-GCNN increases performance by 44.45%, recall by 39.69%, and F1-score by 42.57%. The performance of DC-GCNN compared to that of other CNN-based classifiers in the second experiment set. On the CIFAR-10 and MNIST datasets, the proposed structure outperformed competing techniques with classification accuracies of 89.12% and 99.28%, respectively [36]. The CNN-ELM model implemented integrating the excellent ability of CNN feature learning with ELM efficiency. CNN-ELM also uses ELM as a classifier after extracting features from CNNs, combining the advantages of deep learning with conventional machine learning [46]. Then ELM is used as a classifier to perform a fast and exceptional classification. CNN-HLSGD trains a hinge loss convolution neural network recognition rate was higher than most GTSRB dataset approaches [15]. The DP-KELM algorithm introduced different techniques that classify deep perceptual features through kernel-based extreme learning machines for different traffic sign images (KELM in the deep perceptual Lab color space instead of base RGB color space). The approach uses a basic architecture that lowers computing costs and increases comparative recognition [49]. Graphics processing units (GPUs) optimize time complexity computation for the significant and deeper network [27]. In this paper, Sun C. et al. [37] suggested a technique for recognizing minor traffic signs (Dense-RefineDet) based on RefineDet. They presented a novel anchor-design method for placing small traffic signs at feature-map cell corners or next. A Dense-TCB communicates semantic data from all higher-level levels to the target lower-level layer, resulting in rich contextual data for small-sized traffic signs. Dense-RefineDet was also faster than previous deep-learning-based algorithms due to its single-stage layout [11].

Publicly available standard traffic sign image datasets containing a variety of traffic-sign categories:

- The German Traffic-Sign Recognition Benchmark (GTSRB) Dataset [2, 4, 35, 37, 51]: There are 43 different classes or categories, The training set contains 39,209 images, with the remaining 12,630 images for testing purposes.
- The German Traffic-Sign Detection Benchmark (GTSDb) Dataset [28, 35, 37, 45]: There are 43 different classes or groups, with 900 images. The training set contains 600 images, whereas the testing set contains 300 images.
- The Belgium Traffic Signs (BTS) dataset [22]: There are 62 classes or categories for detection and recognition, 4533 training, and 2562 testing images.
- TSRD dataset [17]: The TSRD contains 6164 traffic sign images divided into 58 sign categories. The images separate into two sub-databases: training and testing. The training database has 4170 images, and the testing database contains 1994 images.

In 2020, Liu Z. et al. [21] introduced improving the small size of traffic sign images for identification. DR-CNN recommended leveraging the deconvolution to the deep layer and the low layer. Huang W et al. [12] proposed the MS COCO technique, which evaluated two-phase adaptive loss function classification to separate complex negative samples from clear positive total loss samples. They looked at ways to speed up the distribution of high-resolution traffic images to achieve on-time productivity for automobile applications in this research, includes driver assistance, near-road mapping, repair stock signs, automatic driving, and automated traffic. A hinge loss stochastic gradient descent (HLSGD) technique for training convolutionary neural networks is proposed (CNNs). Here, with 11,62,284 trainable parameters, CNN consists of three phases (70–110–180). They suggested an HLSGD strategy to

teach CNNs on the German Benchmark for traffic sign identification and contrasted our findings [15]. They introduced DP-KELM, a kernel-based revolutionary learning approach. They used perceptual color space and demonstrated that it was more beneficial for identifying CNN-based traffic signs [49]. Two back propagation neural network (BPNN) models constructed in this article for early yield prediction, namely for immature, small, green fruitlet, and mature red fruits, by combining fruit attributes with (four) tree canopy variables. The four canopy variables chosen are suitable for early yield prediction and provide an elegant method for forecasting fruit, such as apples and maybe other fruit crops, using backpropagation neural network prediction [5]. In this article, they suggested a novel approach to traffic sign identification and recognition. The ResNet-50 is the backbone in the detection phase to construct pyramidal feature networks to improve the semantic representation of small objects and track focal failure [20, 23]. The technique proposed by Xing J. et al. [48], Faster R-CNN for traffic sign detection and identification, and comparison of object detection and recognition results using different networks such as VGGNet, GoogleNet, and ResNet used in this study, achieving 92.6% recall and 93.4% precision. The Adam stands for Adaptive Moment Prediction, which, with hyper-parameters, predicts adaptive learning rates. The Adam optimization algorithm is simple to use, needs little memory, adequate for computer-efficient and sparse gradient issues [5].

3 Proposed methods

The traffic sign dataset uses preprocess traffic sign images and trains a deep CNN traffic sign identification system [38]. It recognizes and detections various signs and symbols [4, 8, 14, 52]. Initially, images transform into an acceptable shape from which feature extraction becomes more accessible. Following that, image preprocessing methods extract features from different layers of CNN and finally classify traffic signs using these features [18].

3.1 Input layer: Input layer takes images as input

3.1.1 CNN

CNN is a neural network of several layers composed of several similar building blocks [1, 52]. Primarily, the machine read images that consist of three simple colors, known as RGB colors, such as Red, Green, and Blue [5, 24, 28]. Each color has its pixel values corresponding to it. The binary image has only two black and white colors.

Here consider the different architecture of CNN or adjust different CNN layers, pooling layer, max pool with dropouts at various levels. Our results do better than with adam optimizer, previous studies on the same datasets in the GTSRB, GTSDB, BTSC, and TSRD datasets.

In convolution layers, a CNN combines numerous kernels for converting both the whole object and the best feature maps, resulting in many characteristics [32]. In the majority of earlier work, fully connected networks classify images. Thus, suppose firstly take input images with 32x32x3 and 200x200x3 pixels. In the first hidden layer, the total number of weights required will be 3072 and 1,20,000, respectively. As a result of dealing with such a large number of parameters, processing a whole complicated image collection provides a more significant number of neurons that could result in overfitting or is impossible. As a result,

rather than fully connected networks, convolutional neural networks are used for image classification. It is composed of neurons with learnable weights and biases. It is similar to a feed-forward neural network. These neurons learn and transform inputs such as images to the corresponding output label during training. CNN, A neuron in a layer is connected to a small number of neurons (such as a local region), needing fewer neurons and processing less weight than a fully connected network. A single neuron connects to all neurons. Each neuron takes in a large number of inputs one at a time and performs a weighted sum on them. When it exceeds it using an activation function, it responds with an output. A CNN, in general, is made from four layers that help extract information from images: convolution, ReLU, pooling, and fully connected. CNN takes inputs as an image of a traffic sign, constructs various damaged images, and classifies them. CNN classifies images by converting them into a matrix of numbers, one for each pixel [3].

The convolution layer consists of many feature filters performing the convolution function [26]. These features evaluate two small bits of larger images and evaluate if they match. This layer involves four steps: first, align the image feature filter and multiply each image pixel by matching the feature pixel. The feature filter crosses the image, discontinues each position on the image, and repeats the previous phases. For each feature filter, this technique performs to obtain convolution output. The other is the rectified linear unit (ReLU layer), which activates a node only if the inputs are more significant than a specific value. Output is 0, as seen in Fig. 6. When the input value passes a specific threshold, there is a linear relationship between it and the dependent variable. Removes all negative values from the equation and converts them to zero. It results by applying it to all the feature images. An additional activation function is sigmoid, seen as S-shape in Fig. 7. The result seems predictable as a probability, suggesting the output is between 0 and 1.

3.1.2 Pooling layer

A pooling layer is preferable to minimize the complexity of feature maps and network parameters following a convolutional approach. Pooling layers, like convolutional layers, interpretable invariantly because their computations take into consideration surrounding pixels. The two most common methods are average and maximum pooling.

3.1.3 Nonlinear layer

The CNN adjusts the input non-linearly in classifying the features contained inside each hidden layer. In the CNN framework, ReLU is used. ReLUs is a frequently used nonlinear transform. This layer performs a fundamental action with a threshold, setting any input value less than zero.

3.1.4 Fully connected layer

After many rounds of the preceding layers, the data reaches the final layer of the CNN, which is the fully connected node. The neurons in the two adjacent layers communicate directly with the neurons in the fully connected network. Our suggested explanation makes advantage of a batch normalization layer. Any channel can be normalized in a mini-batch using the batch normalization layer.

3.1.5 Softmax

To interpret the network's performance is a challenge. In classification [19] problems, using a softmax function to finish the CNN is typical. Each phase will select a traffic sign class based on the features extracted from the previous layers through which the images of traffic signs pass. This layer makes use of the softmax to identify the correct sign class.

3.1.6 Training

Network training by identifying kernels and weights in convolutional layers and fully connected layers minimizes the inconsistency between output predictions and provides ground truth labels on a training dataset [41]. Our model uses the cross-entropy loss as the objective function and used 70% of the data in our study for training.

3.1.7 Testing

The testing dataset evaluates the final design based on the dataset.

3.2 Image preprocessing and data augmentation

To train our CNN, GTSRB [35, 51], GTSDB [51], TSRD [17], and BTSC [17] datasets. Despite this, datasets contain classes such as 43, 43, 62, and 58. The GTSRB database is also another name for the Traffic Sign Recognition Benchmark test images dataset. To train and test the traffic sign recognition and classification model. A large number of convolutional neural network (CNN) approaches were used [29, 44]. Figure 3 also shows images of gray road signs from the GTSRB [14] dataset.

3.2.1 Image preprocessing

The primary objective of image preprocessing in the traffic sign recognition and detection method [13, 35, 53] is to reduce low-frequency background noise, normalize particle image



Fig. 3 Samples of different categories in GTSRB image dataset [50]

amplitude, suppress reflections, and mask image parts [14, 18]. The input image is classified separately into channels R, G, B [3].

3.2.2 Data augmentation

It essential to do some primary data augmentation to prevent overfitting and boost generalization [3]. For this reason, it is using measurements of degrees across its geometric center for angles in the range $[-20^\circ, 20^\circ]$ rotated. Here are some data augmentation parameters as shown in Table 2, after applying data augmentation parameters on images and shows Fig. 4 after augmented images.

3.3 Datasets

The GTSRB dataset has several categories, image data collection that presents difficulties in classifying traffic sign tasks. There are 51,839 samples in the dataset, varying in scale from 15×15 to 250×250 and noncircular. In this dataset, there are 43 classifications, each containing 1000 ~ 1000 images. The GTSRB dataset includes 39,209 training and 12,630 test images. According to view adjustments, shade, color loss, lighting conditions, it can be complicated even for human eyes to recognize all of these signs [2]. In addition, in the Belgian Traffic Sign Classification dataset (BTSC), 4637 training images and 2534 test images are separated into 62 classes. The BTSC dataset includes more distinct forms but fewer training examples of traffic signs than the GTSRB dataset, increasing the complexity of proper classification as shown in Table 3. The GTSDDB dataset includes 900 high-Resolution natural condition images of traffic signals. GTSDDB dataset makes use of 600 training and 300 test images. Finally, combine GTSRB and TSRD datasets to test our research performance on 101 classes of the dataset, as shown in Table 4.

Table 2 Data augmentation parameters

Transformation parameters	Properties
Image rotate	20 ° both sides (clock and anti-clockwise)
Image width modify	10% or 0.1 portion of total image width
Image Height Modify	10% or 0.1 portions of total image height
Image zoom range	20% or 0.2% smaller or more extensive of the original image
Training set	70% or 0.7 of the dataset.
Validation split	30% or 0.3 of the dataset.
Rescale	1/255 multiplied with image

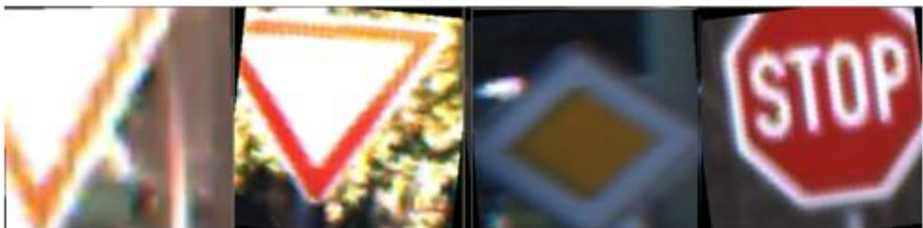


Fig. 4 After Data augmentation

Table 3 Comparison between different publicly available standard datasets

Datasets	Total Number of Images	Total Training images	Test image	Number of Classes
GTSRB	51,839	39,209	12,630	43
GTSDDB	900	600	300	43
BTSC	7171	4637	2534	62
TSRD+GTSRB	58,003	43,379	14,624	101 (58+43)

Table 4 Comparison between different publicly available standard datasets in terms of performance

Dataset	Performance	Number of Classes
GTSRB	99.76%	43
GTSDDB	99.81%	43
BTSC	99.79%	62
TSRD+GTSRB	98.37%	101

3.4 Experimental results

The proposed automatic traffic sign identification system has been tested in this work on the GTSRB, GTSDDB, BTSC, and TSRD datasets, which contained traffic sign images from 43, 43, 62, and 58 classes, respectively. Cross-validation is utilized in datasets where 70% of the dataset from each class is used as the training set and 30% as the validation set. The training dataset is used to train the model, and it improves the model in learning about traffic sign images. A validation dataset is used to offer an unbiased evaluation of a model that has been fitted to training data.

Operating system (OS): Windows 10; Development platform: Python3.8 + OpenCV+ Jupyter notebook; CPU: Intel(R) Core (TM) i7, Memory: 8 GB; Disk: 1 TB.

The image data collection for traffic signs in Belgium has two comprehensive datasets [2, 35]. The comparatively well-established publicly available standard traffic sign identification dataset consists of mainly the GTSDDB dataset [51], the GTSRB dataset [2]. Here different numerical values like precision, recall, f1-score of the GTSDDB dataset shown in Table 5, and Fig. 5 shows its confusion matrix.

3.4.1 Simultaneous classification and detection

The test has four classifications: True Positive (TP), False Positive (FP), False Negative (FN), and True Negative (TN). Precision (P) is the probability of adequately predicting a positive sample. The recall (R) value for positive core samples represents the number of positive samples predicted effectively [42]. As a result of this computation, the following accuracy and recall values determine:

$$P = \frac{TP}{(TP+FP)}$$

Table 5 Report of the confusion matrix

Class ID	Class Name	Precision	Recall	F1-score	Support
0	Speed limit (20 km/h)	1.00	0.97	0.99	112
1	Speed limit (30 km/h)	1.00	0.99	0.99	924
2	Speed limit (50 km/h)	1.00	0.99	1.00	776
3	Speed limit (60 km/h)	0.99	0.99	0.99	508
4	Speed limit (70 km/h)	0.99	1.00	1.00	806
5	Speed limit (80 km/h)	1.00	1.00	1.00	806
6	End of speed limit (80 km/h)	1.00	1.00	1.00	364
7	Speed limit (100 km/h)	1.00	1.00	1.00	216
8	Speed limit (120 km/h)	1.00	1.00	1.00	162
9	No passing	1.00	1.00	1.00	209
10	No passing for vehicles over 3.5 metric tons	0.99	1.00	0.99	474
11	Right of way at the next intersection	0.99	1.00	0.99	79
12	Priority road	1.00	0.99	0.99	476
13	Yield	0.99	0.98	0.99	116
14	Stop	1.00	1.00	1.00	112
15	No vehicles	1.00	1.00	1.00	124
16	Vehicles over 3.5 metric tons prohibited	0.98	0.99	0.99	192
17	No entry	0.98	0.99	0.99	146
18	General caution	1.00	1.00	1.00	598
19	Dangerous curve to the left	1.00	0.97	0.98	196
20	Dangerous curve to the right	1.00	1.00	1.00	88
21	Double curve	1.00	0.98	0.99	224
22	Bumpy road	0.99	1.00	0.99	96
23	Slippery road	0.98	1.00	0.99	534
24	Road narrows on the right	0.99	1.00	0.99	176
25	Road work	1.00	1.00	1.00	297
26	Traffic signals	0.99	1.00	0.99	94
27	Pedestrians	1.00	1.00	1.00	276
28	Children crossing	1.00	0.98	0.99	196
29	Bicycles crossing	1.00	1.00	1.00	484
30	Beware of ice/snow	1.00	0.99	0.99	166
31	Wild animals crossing	1.00	1.00	1.00	87
32	End of all speed and passing limits	0.99	1.00	1.00	776
33	Turn right ahead	1.00	1.00	1.00	116
34	Turn left ahead	0.99	1.00	1.00	766
35	Ahead only	1.00	0.97	0.98	124
36	Go straight or right	1.00	0.99	0.99	94
37	Go straight or left	1.00	1.00	1.00	96
38	Keep right	0.99	0.98	0.99	734
39	Keep left	0.99	1.00	1.00	152
40	Roundabout mandatory	0.99	0.99	0.99	596
41	End of no passing	0.99	1.00	1.00	515
42	No vehicles over 3.5 metric tons	0.99	0.99	0.99	545
	Accuracy	1.00			14,628
	Macro avg	1.00	0.99	0.99	14,628
	Weighted avg	1.00	1.00	1.00	14,628

$$R = \frac{TP}{(TP+FN)}$$

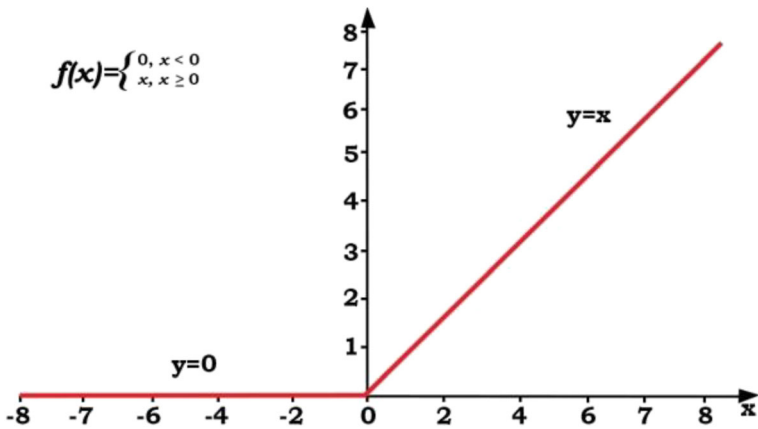


Fig. 6 A Relu activation function

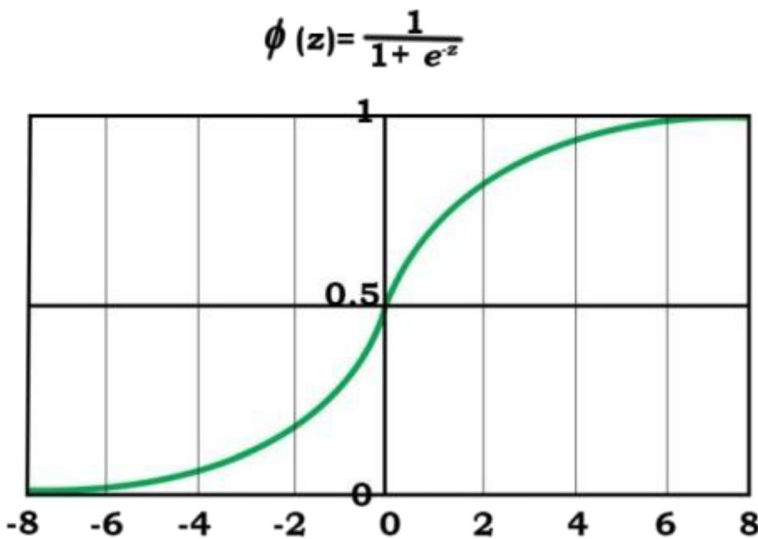


Fig. 7 Sigmoid function

3.4.3 Softmax

The softmax activation feature is a feature in the output layer for classification purposes [36, 47]. This activation function is superior to others in classification because it constrains the outputs of each segment to values between 0 and 1. A softmax activation portion can be denoted mathematically by an equation.

$$\sigma(x)_i = \frac{e^{x_i}}{\sum_{j=1}^J (e^{x_j})}$$

Table 6 Comparison metrics for related work and the proposed method

Paper	Year	Methods	Datasets	Country of region	Number of Classes in Datasets	Dataset ratio for training and testing	Detection Method	Accuracy
Sun Y. et al. [33]	2019	Supervised [16]	GTSRB	Germany	43	90:10	CNN	98.2%
Nartey O. T et al. [21]	2020	Semi-Supervised [25]	GTSRB, BTSC	Germany Belgium	43 62	70:30	CNN	99.27% 98.97%
S. He et al. [11]	2021	Supervised [16]	LISA	United State	47	90:10	CNN, SVM, ResNet 101	95.77%
Our Model	–	Supervised [16]	GTSRB,	Germany	43	70:30	CNN	99.76%
			GTSDDB, BTSC,	Germany Belgium	43 62			99.81% 99.79%
			TSRD+GTSRB	Spain + Germany	101			98.37%

3.4.4 Evaluations on GTSRB, GTSDDB, BTSC, and TSRD

Here our work tested on GTSRB, GTSDDB, BTSC, and TSRD. Table 6 shows the comparison metrics for related work and the proposed method.

Our CNN architecture achieved accuracy on GTSRB, GTSDDB, BTSC, and TSRD + GSRB datasets is 99.76%, 99.81, 99.79, and 98.37%. The training and testing of value accuracy and value loss graphs below in Figs. 8, 9, 10, 11, 12, 13, 14, and 15.

4 Conclusion

In today's motor vehicles, advanced driving assistance systems are helpful to assist. One of the essential functions of an ADAS is to assist autonomous vehicles and drivers in traffic signs. This research proposed a CNN model for traffic TSRD based on the deep learning technique.

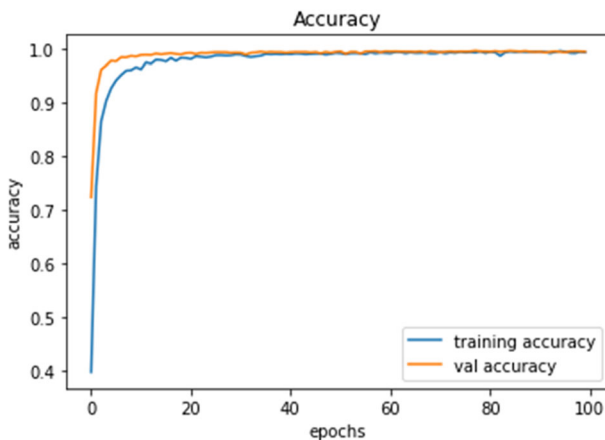


Fig. 8 Training accuracy result of GTSRB

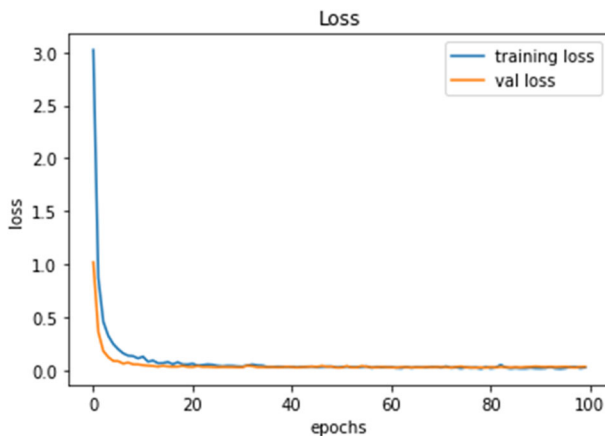


Fig. 9 Training loss result of GTSRB

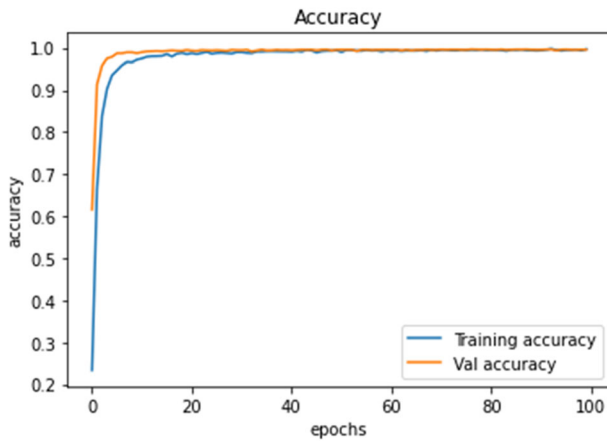


Fig. 10 Training accuracy result of GTSDB

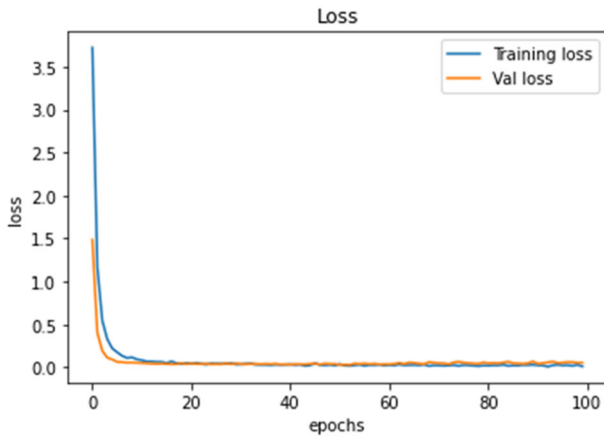


Fig. 11 Training loss result of GTSDB

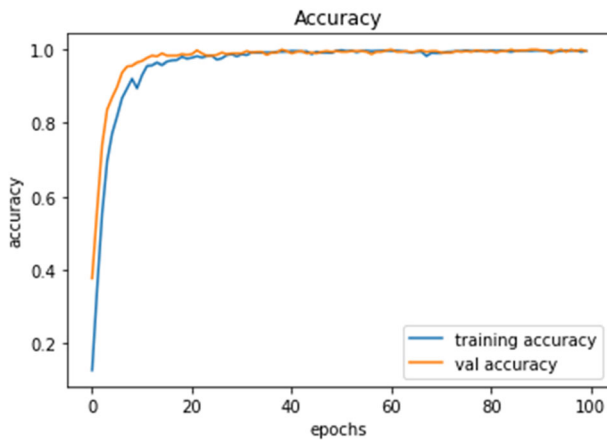


Fig. 12 Training accuracy result of BTSC

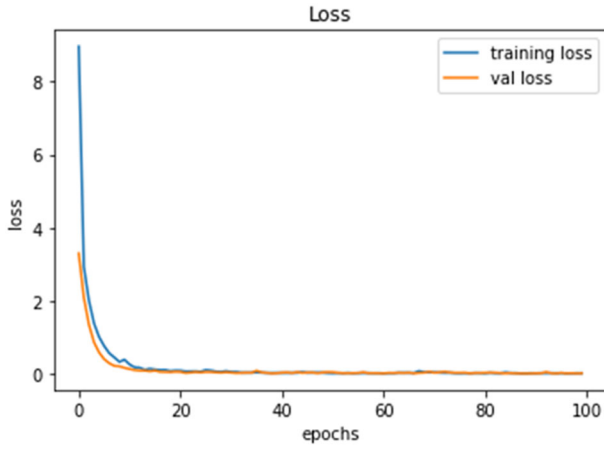


Fig. 13 Training loss result of BTSC

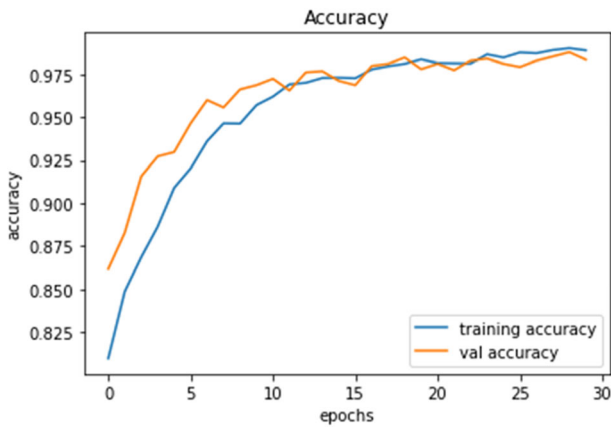


Fig. 14 Training accuracy result of TSRD+GTSRB

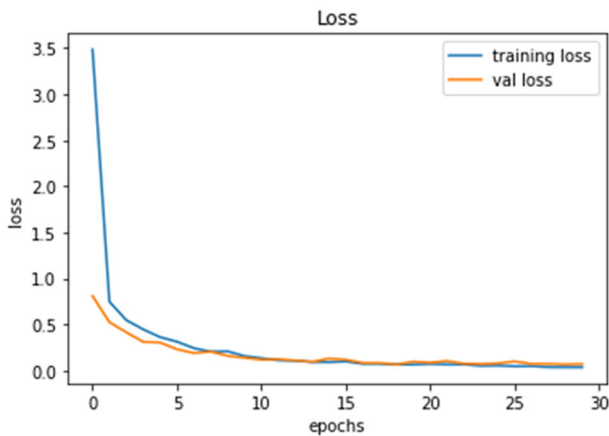


Fig. 15 Training loss result of TSRD+GTSRB

Traffic sign detection is a difficult task. Above all, our model could consistently learn the data, and its accuracy would increase as it trained with data. Our deep CNN network is better CNNs for reliability and accuracy, even in blurred, rotated, and distorted images, by correctly performing image classification and recognition tasks. Our future work will consist of two components. One is to collect traffic signs as tests under difficult weather situations to create our dataset. The second one is Generative Adversarial Network (GANs) to optimize the approach further to create a traffic sign classification and detection system.

Acknowledgments The authors would like to express their gratitude to the anonymous reviewers for their insightful and productive feedback.

References

1. Alghmgham DA, Latif G, Alghazo J, Alzubaidi L (2019) Autonomous traffic sign (ATSR) detection and recognition using deep CNN. *Proc Comput Sci* 163:266–274
2. Arcos-García Á, Álvarez-García JA, Soria-Morillo LM (2018) Deep neural network for traffic sign recognition systems: an analysis of spatial transformers and stochastic optimization methods. *Neural Netw* 99:158–165. <https://doi.org/10.1016/j.neunet.2018.01.005>
3. Bisen D (2021) Deep convolutional neural network based plant species recognition through features of leaf. *Multimed Tools Appl* 80:6443–6456. <https://doi.org/10.1007/s11042-020-10038-w>
4. Cao J, Song C, Peng S, Xiao F, Song S (2019) Improved traffic sign detection and recognition algorithm for intelligent vehicles. *Sensors*
5. Cheng H, Damerow L, Sun Y, Blanke M (2017) Early yield prediction using image analysis of apple fruit and tree canopy features with neural networks. *J Imaging* 3:2–13
6. Dewi C, Chen RC, Yu H (2020) Weight analysis for various prohibitory sign detection and recognition using deep learning. *Multimed Tools Appl* 79:32897–32915. <https://doi.org/10.1007/s11042-020-09509-x>
7. Fu J, Zhao C, Xia Y, Liu W (2020) Vehicle and wheel detection: a novel SSD-based approach and associated large-scale benchmark dataset. Springer Science Business Media, LLC, part of Springer Nature 2020
8. Garcia-Garrido MA, Sotelo MA, Martin-Gorostiza E (2006) Fast traffic sign detection and recognition under changing lighting conditions. *IEEE Intelligent Transportation Systems Conference*
9. Ghosh R (2021) On-road vehicle detection in varying weather conditions using faster R-CNN with several region proposal networks. *Multimed Tools Appl* 80:25985–25999. <https://doi.org/10.1007/s11042-021-10954-5>
10. Gupta H, Verma OP (2021) Monitoring and surveillance of urban road traffic using low altitude drone images: a deep learning approach. *Multimedia Tools Appl*. <https://doi.org/10.1007/s11042-021-11146-x>
11. He S et al (2021) Automatic recognition of traffic signs based on visual inspection. *IEEE Access* 9:43253–43261. <https://doi.org/10.1109/ACCESS.2021.3059052>
12. Huang W, Huang M, Zhang Y (2018) Detection of traffic signs based on combination of GAN and faster-RCNN. *IOP Conf. Ser J Phys Conf Ser*
13. James Deva Koresh H (2019) Computer vision based traffic sign sensing for smart transport. *J Innov Image Process (JIIP)* 01(01):11–19
14. Jiang Y, Li C (2020) Convolutional neural networks for image-based high-throughput plant phenotyping: a review. *Plant Phenomics*
15. Jin J, Fu K, Zhang C (2014) Traffic sign recognition with hinge loss trained convolutional neural networks. *IEEE Trans Intell Transp Syst* 15:1991–2000
16. Khan MA, Mittal M, Goyal LM, Roy S (2021) A deep survey on supervised learning based human detection and activity classification methods. *Multimed Tools Appl* 80:27867–27923. <https://doi.org/10.1007/s11042-021-10811-5>
17. Kherarba M, Abbas MT, Boumerdassi S, Meddah M, Benhamada A, Senouci M (2021) Road sign identification with convolutional neural network using tensorflow. In: Renault É, Boumerdassi S, Mühlenthaler P (eds) *Machine learning for networking. MLN 2020. Lecture notes in computer science*, vol 12629. Springer, Cham. https://doi.org/10.1007/978-3-030-70866-5_17

18. Laguna R, Barrientos R, Blázquez LF, Miguel LJ (2014) Traffic sign recognition application based on image processing techniques. *IFAC Proce* Vol 47(3):104–109
19. Li W, Qu F, Liu J, Sun F, Wang Y (2020) A lane detection network based on IBN and attention. *Multimed Tools Appl* 79:16473–16486. <https://doi.org/10.1007/s11042-019-7475-x>
20. Liang Z, Shao J, Zhang D, Gao L (2019) Traffic sign detection and recognition based on pyramidal convolutional networks. *Neural Computing and Applications. Multi-Source Data Understanding (MSDU)* Springer
21. Liu Z, Li D, Ge S, Tian F (2019) Small traffic sign detection from the large image. *Applied intelligence, springer science+business media, LLC, part of springer nature*
22. Loukmane A, Graña M, Mestari M (2020) A model for classification of traffic signs using improved convolutional neural network and image enhancement. *Fourth International Conference On Intelligent Computing in Data Sciences (ICDS)*, pp 1–8. <https://doi.org/10.1109/ICDS50568.2020.9268761>
23. Lu Y, Lu J, Zhang S, Hall P (2018) Traffic signal detection and classification in street views using an attention model. *Vol 4, No 3*, pp 253–266
24. Mishra J, Sharma A, Chaturvedi K (2011) An unsupervised cluster-based image retrieval algorithm using relevance feedback. *Int J Manag Inf Technol (IJMIT)* 3(2):9.16
25. Nartey OT, Yang G, Asare SK, Wu J, Frempong LN (2020) Robust semi-supervised traffic sign recognition via self-training and weakly-supervised learning. *Sensors*. 20(9):2684. <https://doi.org/10.3390/s20092684>
26. Natarajan S, Annamraju AK, Baradkar CS (2018) Traffic sign recognition using weighted multi-convolutional neural network. *IET Intell Transp Syst* 12:1396–1405. <https://doi.org/10.1049/iet-its.2018.5171>
27. Ngiam J, Coates A, Lahiri A, Prochnow B, Ng A (2011) On optimization methods for deep learning. In *Proceedings of the 28th International Conference on Machine Learning (ICML)*, pp 265–272
28. Saadna Y, Behloul A (2017) An overview of traffic sign detection and classification methods. *Int J Multimed Inf Retriev* 6:193–210
29. Saha S, Islam S, Khaled A, Tairin S (2019) An efficient traffic sign recognition approach using a novel deep neural network selection architecture. *Department of Computer Science and Engineering, BRAC University, Bangladesh Springer Nature Singapore*
30. Said Y, Barr M (2021) Countries flags detection based on local context network and color features. *Multimed Tools Appl* 80:14753–14765. <https://doi.org/10.1007/s11042-021-10509-8>
31. Sajja TK, Kalluri HK (2021) Image classification using regularized convolutional neural network design with dimensionality reduction modules: RCNN–DRM. *J Ambient Intell Human Comput* 12:9423–9434. <https://doi.org/10.1007/s12652-020-02663-y>
32. Sarigul M, Ozyildirim BM, Avci M (2020) Deep convolutional generalized classifier neural network. *Neural Process Lett* 51:2839–2854. <https://doi.org/10.1007/s11063-020-10233-8>
33. Satti SK, Devi K, Dhar P, Srinivasan P (2020) Enhancing and classifying traffic signs using computer vision and deep convolutional neural network 243–253. https://doi.org/10.1007/978-981-15-6315-7_20
34. Serna CG, Ruichek Y (2021) Classification of traffic signs: the european dataset. *IEEE Access* 6:78136–78148, 2018. <https://doi.org/10.1109/ACCESS.2018.2884826>
35. Shustanov A, Yakimov P (2017) CNN design for real-time traffic sign recognition. *Proc Eng* 201:718–725
36. Singh D, Chahar V, Yadav V, Kaur M (2020) Classification of COVID-19 patients from chest CT images using multi-objective differential evolution-based convolutional neural networks. *Eur J Clin Microbiol Infect Dis*, Springer-Verlag GmbH Germany, part of springer nature
37. Sun C, Ai Y, Wang S, Zhang W (2020) Dense-Refinedet for traffic sign detection and classification. *Sensors* 20(22):6570. <https://doi.org/10.3390/s20226570>
38. Sun Y, Ge P, Liu D (2019) Traffic sign detection and recognition based on convolutional neural network. *2019 Chinese automation congress (CAC)*. <https://doi.org/10.1109/cac48633.2019.8997240>
39. Taki Y, Zemmouri E (2021) An overview of real-time traffic sign detection and classification. In: Ben Ahmed M, Rakıp Kardeş İ, Santos D, Sergeyeveva O, Boudhir AA (eds) *Innovations in smart cities applications volume 4*. SCA 2020. *Lecture notes in networks and systems*, vol 183. Springer, Cham https://doi.org/10.1007/978-3-030-66840-2_26
40. The German Traffic Sign Recognition Benchmark (GTSRB) and the German Traffic Sign Detection Benchmark (GTSDB): <https://benchmark.ini.rub.de/index.html>- February 2013
41. Timofte R, Zimmermann K, Van Gool L (2014) Multi-view traffic sign detection, recognition, and 3D localisation. *Mach Vis Appl* 25:633–647. <https://doi.org/10.1007/s00138-011-0391-3>
42. Triki N, Ksantini M, Karray M (2021) Traffic sign recognition system based on belief functions theory. In *proceedings of the 13th international conference on agents and artificial intelligence - volume 2: ICAART*, ISBN 978-989-758-484-8, pp 775–780. <https://doi.org/10.5220/0010239807750780>
43. Villalón-Sepúlveda G, Torres-Torriti M, Flores-Calero M (2017) Traffic sign detection system for locating road intersections and roundabouts: the chilean case. *Sensors*

44. Wali SB, Abdullah MA, Hannan MA, Hussain A, Samad SA, Ker PJ, Mansor MB (2021) Vision-based traffic sign detection and recognition systems. *Curr Trends Challenges Sens.* <https://doi.org/10.3390/s19092093>
45. Wang G, Ren G, Jiang L, Quan T (2013) Hole-based traffic sign detection method for traffic signs with red rim. *Vis Comput* 30(5):539–551
46. Wu Y, Liu Y, Li J, Liu H, Hu X (2013) Traffic sign detection based on convolutional neural networks. *The 2013 International Joint Conference on Neural Networks (IJCNN)*, pp 1–7. <https://doi.org/10.1109/IJCNN.2013.6706811>
47. 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
48. Xing J, Yan WQ (2021) Traffic sign recognition using guided image filtering. *Geometry Vis* 1386:85–99. https://doi.org/10.1007/978-3-030-72073-5_7
49. Zeng Y, Xu X, Fang Y, Zhao K (2015) Traffic sign recognition using deep convolutional networks and extreme learning machine. In *proceedings of international conference on intelligence science and big data engineering (IScIDE)*, pp 272–280
50. Zeng Y, Xu X, Shen D, Fang Y (2017) Traffic sign recognition using kernel extreme learning machines with deep perceptual features. *IEEE Trans Intell Transp Syst* 18(6):1647–1653
51. Zhang J, Huang M, Jin X, Li X (2017) A real-time chinese traffic sign detection algorithm based on modified YOLOv2 algorithms. *10(4):127*
52. Zhu Z, Liang D, Zhang S, Huang X, Li B, Hu S (2016) Traffic-sign detection and classification in the wild. In *proceedings of the IEEE conference on computer vision and pattern recognition*, pp 2110–2118
53. Zhu Y, Zhang C, Zhou D, Wang X, Bai X, Liu W (2016) Traffic sign detection and recognition using fully convolutional network guided proposals. *Neurocomputing* 214:758–766

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.