IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 22, 2021, accepted February 6, 2021, date of publication February 12, 2021, date of current version March 23, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3059052

Automatic Recognition of Traffic Signs Based on Visual Inspection

SHOUHUI HE[1], LEI CHEN[1], SHAOYUN ZHANG[1], ZHUANGXIAN GUO[1], PENGJIE SUN[1], HONG LIU[1], AND HONGDA LIU[2]

[1]Linyi University, Linyi 276000, China
[2]Linyi Audit Bureau, Linyi 276000, China

Corresponding author: Lei Chen (chenleiwl@lyu.edu.cn)

This work was supported in part by the Humanity and Social Science Research Foundation of Ministry of Education in China under Grant 20YJAZH131.

ABSTRACT The automatic recognition of traffic signs is essential to autonomous driving, assisted driving, and driving safety. Currently, convolutional neural network (CNN) is the most popular deep learning algorithm in traffic sign recognition. However, the CNN cannot capture the poses, perspectives, and directions of the image, nor accurately recognize traffic signs from different perspectives. To solve the problem, the authors presented an automatic recognition algorithm for traffic signs based on visual inspection. For the accuracy of visual inspection, a region of interest (ROI) extraction method was designed through content analysis and key information recognition. Besides, a Histogram of Oriented Gradients (HOG) method was developed for image detection to prevent projection distortion. Furthermore, a traffic sign recognition learning architecture was created based on CapsNet, which relies on neurons to represent target parameters like dynamic routing, path pose and direction, and effectively capture the traffic sign information from different angles or directions. Finally, our model was compared with several baseline methods through experiments on LISA (Laboratory for Intelligent and Safe Automobiles) traffic sign dataset. The model performance was measured by mean average precision (MAP), time, memory, floating point operations per second (FLOPS), and parameter number. The results show that our model consumed shorter time yet better recognition performance than baseline methods, including CNN, support vector machine (SVM), and region-based fully convolutional network (R-FCN) ResNet 101.

INDEX TERMS Traffic signs, automatic recognition system, CapsNet, traffic safety.

I. INTRODUCTION

Traffic sign recognition is a research hotspot in the application of visual navigation and computer vision in intelligent driving [1], [2]. Under multiple constraints, the recognition of traffic signs needs to realize various goals with a high accuracy through complex implementation methods. A minor classification error of traffic signs will bring disastrous consequences.

In automatic driving, most targets, including traffic lights, routes, special vehicles, and the gestures of traffic police, are recognized by cameras or vehicle-to-everything (V2X) communication. Meanwhile, radar is intrinsically unable to identify signals like speed limit and stop sign. Cameras are installed on the dashboard of many autonomous vehicles and driver assistance systems, and used to capture the real-time images or videos embedded in the machine learning model of the car system. The deep learning algorithm of the model must be robust and reliable, so that the model could capture the traffic signs in different directions and poses. After all, the speed and geographic location change continuously as the vehicle drives through different environments and lights.

However, traditional traffic sign recognition algorithms are basically driven by tasks, namely, color detection, shape recognition, and machine learning. Most of them are only applied in fully or semi-enclosed environments like expressways. Even the most popular traffic sign recognition algorithm, convolutional neural network (CNN), cannot effectively capture traffic sign features like pose, angle, and direction, due to the defect in max-pooling layer [3]. For software reasons, the image quality will be reduced if the images are collected or transmitted on the computer.

The associate editor coordinating the review of this manuscript and approving it for publication was Muhammad Khurram Khan.

In addition, the quality of the images collected by the imaging sensor varies greatly. For example, the image quality is generally undesirable on rainy or foggy days, in dark nights, and under very dark light.

The CNN-based machine learning models are unable to cope with the above challenges [4]. Therefore, it is urgent to cover traffic signs in the hierarchical contours of computer vision, and improve the accuracy and stability of traffic sign recognition. During the observation of scene images, the attention should be focused on the targets or regions of interest (ROIs). These targets or regions must carry striking visual features, such as edge contour, detailed texture, color gradient direction, color intensity, and spatial location.

On this basis, this paper designs a method to extract candidate regions from traffic sign images through content analysis and key information acquisition. In complex scenes, our method could extract salient foreground targets with universal significance from the input image, and realize the recognition of multiple traffic sign images. However, the input images must meet two requirements: (1) Each image needs to suit the perception mechanism of human eyes, that is, the visual target area must be clearly different from the background; (2) The collaborative visual targets in multiple images must have obvious similarities.

Angle is a thorny issue in image recognition, if there are illumination changes, and occlusions. In this case, it is difficult to obtain useful features through classification. Suppose our goal is to design a detector for the buttons on shirts or jackets, which are usually round (or oval in images) with several holes. Through edge detection, it is easy to judge whether a target in the image is a button based on the edges. In this example, edge information is useful, while color information is not. Besides, the useful features should also be discriminable. For instance, the good features extracted from an image should be able to differentiate buttons from other round objects (e.g., coins and wheels). Thus, the Histogram of Oriented Gradients (HOG) was designed for the target image, and the distribution of gradient directions was treated as a feature, in order to solve the projection distortion in image recognition. The gradient of the image (derivatives in the x and y directions) [5] is very useful, because the edges and corners of the image (regions where the intensity changes sharply) have a large amplitude. Compared with other areas on the same plane, edges and corners contain more information about the shape of the object.

To sum up, the traditional CNN cannot effectively recognize traffic signs in images taken in different environments, illuminations, speeds, positions, poses, angles, or directions. This paper puts forward a CapsNet-based traffic sign learning system [6]–[8]. The neuronal system can dynamically capture the poses and directions of vehicles on road, effectively identify the traffic signs in different angles and directions, and give case descriptions of traffic signs.

The main contributions of this paper are as follows:

1. To improve visual inspection effect, the authors designed a method to extract candidate regions from the input image through content analysis and key information recognition [9].

2. To prevent projection distortion, the authors developed an HOG method for actual images.

3. A CapsNet-based traffic sign learning system was created to effectively capture the poses and directions of traffic signs.

4. Through repeated experiments, our method was proved better than traditional CNN [10], [11], support vector machine (SVM) [12], [13], and region-based fully convolutional network (R-FCN) ResNet 101 in traffic sign recognition [14].

The rest of this paper is organized as follows: Section 2 reviews the related work of traffic sign recognition; Section 3 explains our approach; Section 4 analyzes, compares, and discusses the results of traffic sign recognition; Section 5 sums up the findings and provides suggestions on future research.

II. LITERATURE REVIEW

The recent traffic sign recognition methods are mainly based on shape or deep learning.

A. SHAPE-BASED TRAFFIC SIGN RECOGNITION

To detect triangular and rectangular symbols, Anandhalli and Baligar [15] applied the Harris corner detector to the ROIs, searched for corners in the predefined control area. Li [16] relied on edge information to recognize traffic signs that are difficult to detect in the driving environment: Based on the shape features of scale-invariant edge turning angles, the nonparametric shape detector was used to detect circles, triangles, and rectangles in the image; more than 95% of all traffic signs were covered by this detector. Jin *et al.* [17] derived a two-module detector from the multi-feature fusion traffic sign recognition method: the first module extracts the ROIs, using the commonality of symbol boundaries; the latter verifies the effectiveness of the extracted ROIs, and combines HOG and SVM to detect traffic signs. Zheng *et al.* [18] presented a sliding window detection method, which searches for traffic signs on different scales with the integrated channel feature classifier. Targeting the prohibition and mandatory signs in German Traffic Sign Benchmarks (GTSDB), Gim *et al.* [19] developed a system containing two coarse filter modules: the first module is based on HOG and linear discrete analysis (LDA); the second is based on a small sliding window; both modules involve a large window and an SVM classifier. While these efforts are effective in recognizing traffic signs based on graphical methods, these solutions do not work well in complex scenarios (e.g., low light, signs partially obscured, etc.), and are particularly ineffective in recognizing signs with different orientations or viewpoints.

B. DEEP LEARNING-BASED TRAFFIC SIGN RECOGNITION

Jumani *et al.* [20] integrated the local binary pattern (LBP) feature detector with the AdaBoost classifier [21] to roughly

FIGURE 1. Different types of traffic signs.

select the ROIs, and reduced the negative ROIs in traffic sign recognition by cascading the CNNs. Shao *et al.* [22] came up with a traffic sign recognition method for complete traffic network: the region-based CNN (R-CNN) was extended by the object proposal method of EdgeBox [23], and achieved the optimal results on Swedish traffic-sign dataset (STSD) [24]. Through convolution extension, Li *et al.* [25] created a CNN with multi-scale sliding window; the extended (or atrophic) convolution, which supports the exponential expansion of the receptive field, without scarifying the resolution or coverage, was adopted to expand a convolution filter; this filter was used to piece up a large background through fast computation with a few parameters. In addition, several loss functions have been proposed for bounding box regression: intersection over union network (IoU-Net) [26], Precise Rol Pooling (PrRol-Pooling) [27], and generalized IoU (GIoU) [28]. These functions open a new way to recognize traffic signs with multi-scale CNN. In general, deep learning-based traffic sign recognition learns features through big data training. The relevant methods are good at feature expression, and immune to external factors like illumination changes and occlusions. Compared with conventional methods, deep learning-based traffic sign recognition is highly accurate and strong in generalization.

III. REGION EXTRACTION OUR METHOD

A. REGION EXTRACTION

There is a high resemblance between the public datasets on traffic signs, because most places around the world use similar traffic signs. To recognize traffic signs based on deep learning, it is necessary to focus on the structural information of traffic signs in visual recognition, and extract every pixel from the traffic sign image. Hence, the authors designed an extraction method for ROIs that could identify the key information in traffic sign images (Figure 1). The designed ROI extraction method works in the following steps:

Step 1. The contour lines and blocks are recognized in the traffic sign image, and the distance between them is set to a fixed value.

Step 2. The traffic sign image is divided into small blocks to reduce repeated reading of the same information and prevent block loss.

Step 3. Each image block is recorded, and marked with the details on the high-frequency regions (e.g., traffic sign) extracted by a simple multilayer perceptron (MLP).

During image recognition, thick edges and end points are common in edge feature extraction and block division. These disturbances might affect the accuracy of the contour image, and cause information loss. To suppress these disturbances, the contour image was recognized layer by layer with a 4-level recognition and matching method.

From the angle of smoothness, most contour curvatures, except for a few easily identifiable geometric features, are difficult to recognize during contour matching. In real-world applications, most contours in digital images are expressed by pixels. To a certain extent, the smoothness of these contours is hard to express, making it difficult to calculate curvature. Thus, this paper develops an HOG method to facilitate the curvature calculation.

To recognize traffic signs, the contours were identified layer by layer to ensure the recognition accuracy. First, the binary image was tested to acquire the contour pixels of the contour image. After that, the contour features were realized on OpenCV.

Traffic sign images are relatively clear. Many of them have a high frequency. After low-pass filtering, the high-frequency information will be removed. The filtered image will differ sharply from the original image. Based on the error in structural similarity, the recognizable features could be extracted effectively. Besides, the variation in the measured contour shape of the traffic sign could be treated as a feature, making it easy to distinguish between traffic signs in different shapes. That is, common traffic sign images can be expressed as circles or squares.

When the length of the contour line is the same, the standard deviation of the pointing distance amounts to 0.00, 0.34, 1.00. The counter line must be unified in different types of traffic signs, whether they are for prohibition, warning, or instruction. For example, a prohibition traffic sign should be a black triangle with a graduated color background, a warning sign should have a yellow background, and an instruction sign should be a white circle with a blue background. The proposed ROI extraction method was tested on actual traffic sign images. The results show that this

proposed visual inspection method has good stability and fault-tolerance.

B. HOG FEATURE

The gradient of the image, i.e., the differentials in the x and y directions, is very useful. The edges and corners have very large gradients. Compared with other areas on the plane, these areas with sharp intensity changes contain rich information on object shape. Thus, this paper designs and describes an HOG algorithm. For clarity, the steps of the algorithm were explained with an ordinary image with movement trends.

Step 1. Preprocessing

The aspect ratio must be fixed before analyzing the multi-scale image blocks. Here, the aspect ratio of image blocks is maintained at 1:2. An original image of 720 × 475 resolution (Figure 2) was selected as an example to illustrate HOG calculation. The small blocks cropped from the image were of the size 64 × 128.

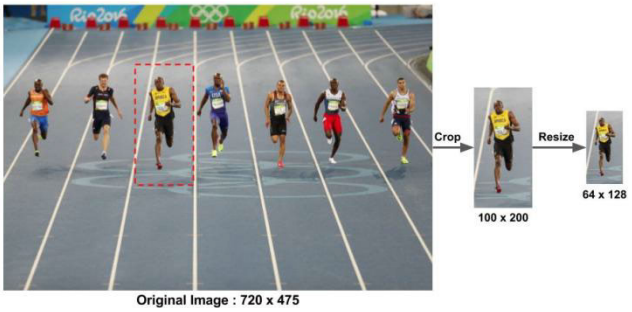

FIGURE 2. An example image.

Step 2. HOG calculation

The first step of HOG calculation is to compute the horizontal and vertical gradients of the image. Taking the kernel size of 1, the Sobel operator of OpenCV was chosen to derive the magnitude and direction of the gradient:

$$g = \sqrt{g_x^2 + g_y^2}$$
$$\theta = \arctan \frac{g_y}{g_x} \quad (1)$$

The gradient magnitude and direction can also be calculated by CartToPolar function. The calculated gradient magnitude is shown in Figure 3 [29].

As shown in Figure 3, straight lines and horizontal lines were highlighted in the x- and y-direction gradients, respectively. The gradient amplitude changed, where the intensity changed drastically. Virtually no gradient was distributed on flat areas. The gradient image contains no contour information, except for many unnecessary information (e.g., constant background). The amplitude and direction of gradient varied from pixel to pixel. For a color image, gradients of the 3 channels need to be calculated separately. The maximum amplitude among the 3 channels should be taken as the gradient amplitude; the corresponding angle should be treated as the gradient direction.

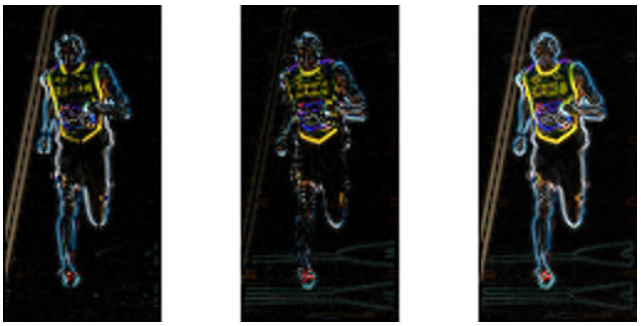

FIGURE 3. Gradient magnitude map (left: x direction; middle: y direction; right: gradient magnitude).

Step 3. Cell division

As shown in Figure 4, the image was further decomposed into 8 × 8 cells, and the HOG of each cell was calculated. The HOG feature was chosen to describe a small area of the image, thanks to its ability to describe the original image in a concise manner.

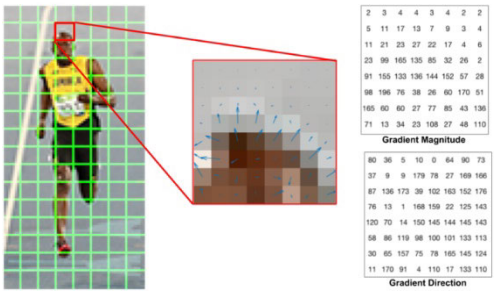

FIGURE 4. Each RGB cell and its gradient (left) [30]; gradient magnitude and direction (right).

In each image block, the gradient at a pixel contains two values: the gradient magnitude and direction. In total, there are 8 × 8 × 2 = 128 values, which could be described by a 9-bin HOG. In this way, the original image was expressed more compactly, and the HOG of each block was robust against noises.

Although the gradient information of a single pixel might contain noises, the HOG of 8 × 8 image blocks are very insensitive to noise. The 8 × 8 cells of each 64 × 128 image block can capture very interesting features, namely, face and top of head.

In the right subgraph of Figure 4, there were slight differences in the numbers representing the gradient in the 8 × 8 cell: the angle fell between 0° and 180°, rather than between 0° and 360°. This is called unsigned gradient, for the positive and negative gradient directions were represented by the same number. That is, a gradient arrow and its corresponding value (the value corresponding to the direction plus 180°) are regarded as the same gradient. Empirical evidence shows that unsigned gradients are better than signed gradients.

Step 4. Normalization of 16 × 16 blocks

Then, the HOG was plotted based on the gradient calculated through the above step. The image gradient is highly sensitive to the overall illumination. If the pixel values of the image are all divided by two, the entire image will become darker, and the gradient magnitude will be halved. Then, the HOG will be half the original size. In theory, the descriptor is expected to stay invariant against illumination changes, that is, the HOG should be normalized to exclude illumination interference.

The HOG feature of gradient direction was normalized by 16 × 16 blocks, each of which contain 4 HOGs. These HOGs could be connected to a 36 × 1 vector, which can be normalized to the size of 3 × 1. After each normalization, the whole window was moved by 8 pixels, and used to obtain another 36 × 1 normalized vector. This process was repeated continuously.

Step 4. Calculation of HOG feature of gradient direction

To calculate the final eigenvector of the entire block, all 36 × 1 vectors were connected into a large vector, and the HOG feature of gradient direction was calculated as follows:

The total number of 16 × 16 blocks at different positions was obtained as 7 × 15 = 105, where 7=(64-8)/8 is the number of blocks at horizontal positions, and 15=(128-8)/8 is the number of blocks at vertical positions. Since each 16 × 16 block was represented as a 36 × 1 vector, the large vector connected from them has a dimension of 3,780 = 36 × 105.

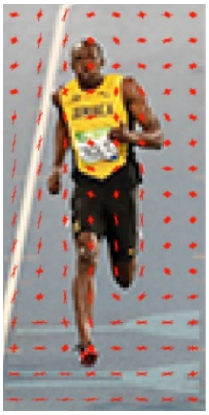

FIGURE 5. The visualized HOG feature.

Then, the HOG feature was visualized as shown in Figure 5. By drawing the 9 × 1 normalized vectors (histograms) in all 8 × 8 cells, it is clear that the histogram mainly captures the body shape, especially near the torso and legs.

C. STACKED CAPSNET

As shown in Figure 6, our model is a stacked model, in which the basic components are arranged in the shape of a fork. The leftmost component is a convolutional CapsNet. Similar to a filter, each capsule scans the input image, and outputs a part of that image. Suppose the task is to recognize handwritten numbers. The ordinary neural networks will output ten neurons, each of which corresponds to a possible number. Meanwhile, the CapsNet will output ten vectorized capsules, each of which corresponds to a possible number. The degree of normalization of the vectorized output reflects the confidence of the output. For example, the capsule corresponding to number 1 is outputted in the form of the vector corresponding to 1, and the degree of normalization of the capsule is the confidence of 1. The rest can be deduced by analogy. During the training, the goal is to maximize the confidence of output numbers; if numbers are imported to the CapsNet, the training goal will be maximizing the degree of normalization. Mathematically, CapsNet can be described as:

$$\mu^1 = \mathrm{W}^1 x^1, \mu^2 = \mathrm{W}^2 x^2$$
$$S = \mathrm{c}_1 \mu^1 + \mathrm{c}_2 \mu^2 \tag{2}$$

In short, the input scalar x was multiplied by the weight w, and converted into vector u; Next, the input vector u was multiplied by the weight c, and then summed up into vector S; After that, vector S was converted into vector v using the nonlinear function, i.e., the novel activation function Squashing. Hence, the output v can be calculated by:

$$v = \frac{\|S\|^2}{1 + \|S\|^2} \frac{S}{\|S\|} \tag{3}$$

The first part of the activation function is the zoom scale of the input vector S, and the latter part is the unit vector of S. This activation function not only preserves the direction of the input vector, but also compresses the modulus of the input vector to [0, 1]. In other words, the vector modulus is positively correlated with the probability of appearance for an entity [5], [6].

The intermediate structure in Figure 6 is the double stacking of residuals. In the classic ResNet, the input of the current stack is added to the output before the result is transferred to the next stack. This practice effectively prevents overfitting, and ensures the mining of image features, making the deep structure more trainable. In our network, two multi-level residual branches were designed (middle and right in Figure 6). One of them implements reverse detection on each layer, and the other operates on the detection branch of each layer. The operation can be described by:

$$x_l = x_{l-1} - \hat{x}_{l-1}, \hat{y} = \sum_l \hat{y}_l \tag{4}$$

The first block receives model-level input $x, x1 \equiv x$, and provides an output v1 that propagates to the next block. The output v1 is stacked with v2 of the next block. In this way, the outputs can be summated layer by layer, making the deep network interpretable. This stacking strategy has several advantages: the actual features are approximated, the detection in downstream blocks is simplified, and the backpropagation of gradients is facilitated.

The rightmost structure in Figure 6 contains two stacks: trend and branch. The trend takes the output of each stack as the input of the next stack; the branch superimposes the outputs of all stacks, and imports them to the subsequent fully-connected layer. The trend stack consists of multiple

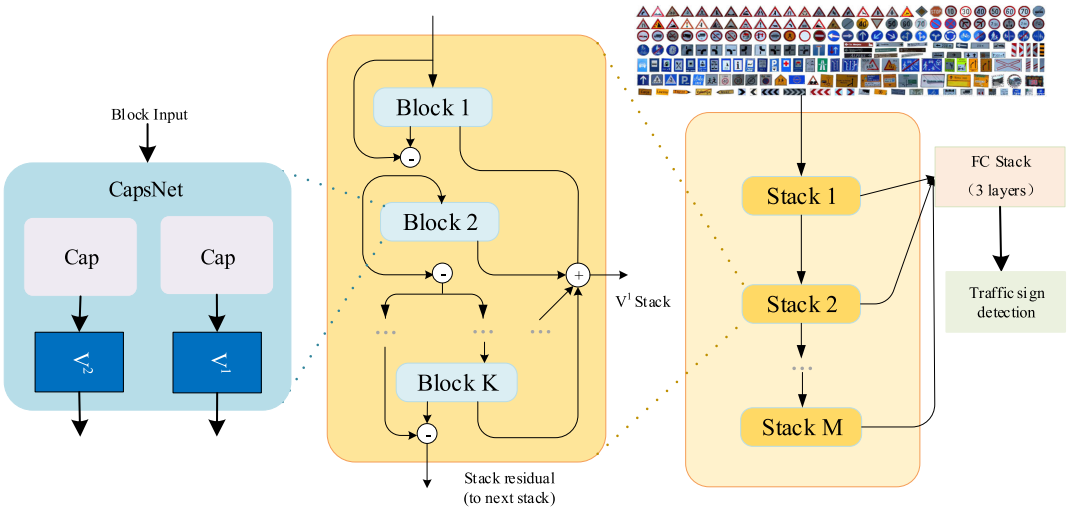

FIGURE 6. The stacked model we have been designed.

blocks connected by the residual connections (Figure 6). Each block has a unique CapsNet, which cannot be learned by others. The branch blocks can share weights to improve the verification performance. The operation of the l-th stack can be described by:

$$\begin{aligned} \mathrm{h}_{l,1} &= FC_{l,1}\left(\mathrm{x}_l\right), \\ \mathrm{h}_{l,2} &= FC_{l,2}\left(\mathrm{h}_{l,1}\right), \\ \mathrm{h}_{l,3} &= FC_{l,3}\left(\mathrm{h}_{l,2}\right), \\ \mathrm{h}_{l,4} &= FC_{l,4}\left(\mathrm{h}_{l,3}\right) \\ \mathrm{v}_l^b &= LINEAR_l^b\left(\mathrm{h}_{l,4}\right), \\ \mathrm{v}_l^f &= LINEAR_l^f\left(\mathrm{h}_{l,4}\right) \end{aligned} \tag{5}$$

where, LINEAR is the linear projection layer, that is, a $\theta_l^f = W_l^f h_{l,4}$ type network layer. This layer is a fully-connected layer with the nonlinear features of rectified linear unit (ReLU). Thus, the sub-network $FC_{l,1}$ can be obtained, e.g., $h_{l,1} = RELU\left(W_{l,1}x_l + b_{l,1}\right)$. For this sub-network, the specific task is to detect the forward expansion coefficient v_l^f, and the ultimate goal is to optimize part of the recognition accuracy of $\hat{y}_l$ by properly mixing the basis vectors provided by v_l^f. In addition, the detection expansion coefficient v_l^b of this sub-network stems from the estimated value of x_l, with the goal of removing unhelpful inputs to help downstream blocks.

The last module in Figure 6 is a three-layer fully connected layer. The number of neurons in each layer is an empirical value that varies with datasets. This layer involves techniques like DropOut and L2 regularization. The last layer was activated by softmax function. Thus, the objective function of our model can be defined as:

$$L = -[y \log \hat{y} + (1 - y) \log(1 - \hat{y})] \tag{6}$$

where, y and y are actual and recognized labels, respectively.

IV. EXPERIMENTS

A. DATASET

LISA (Laboratory for Intelligent and Safe Automobiles) traffic sign dataset, an open-source dataset, was selected for our experiments. There are 47 types of traffic signs in this dataset. Only one subset of LISA was used, for our method focuses on a specific type of traffic signs. The selected subset contains 654,285 images. The dataset size was further reduced by removing all the training samples without default matching boxes.

B. EXPERIMENTAL SETTINGS

Our model and baseline methods were programmed on Keras, under the TensorFlow framework. The stacked CapsNet model was realized, using CUDA Toolkit, and the GPU (graphics processing unit)-accelerated library of primitives called NVIDIA CUDA® Deep Neural Network library (cuDNN). The model training was conducted on an Intel Core i710500U 2.7GHz CPU (central processing unit) (memory: 8GB; RAM: 1TB) and an NVGT 940MX2 GDDR3 NvidiaGeForce GPU.

To reduce the dataset size, the original data were divided into a training set and a test set by 9:1. Each model was trained by the AdaDelta optimizer. The default hyperparameters were provided by TensorFlow. The training lasted 200 epochs, with the batch size of 128k. The baseline models include CNN, SVM, and R-FCN ResNet 101 [31].

The performance of each model was evaluated by mean average precision (mAP). First, the interpolation average accuracy (AP) of the tracking accuracy/recall curve was calculated by setting the recall r to the maximum accuracy ob of any recalls r ≥ r' (formula (7)), with p(r) being the recall r of measured value. The AP value is equivalent to the area under the curve of numerical integration, which is the product between the sum of precision variation and the recall variation r(k) at k points (formula (8)), with N being the total number

of points with recall variation. Finally, the average of all APs was taken as the mAP value.

$$p(r) = \max_{r'y' \geq r} p\left(r'\right) \tag{7}$$

$$AP = \sum_{k=1}^{N} p(k)\Delta r(k) \tag{8}$$

C. RESULTS

Traffic signs have great differences in illumination and contrast. Thus, the original images were enhanced and normalized by our method and several MATLAB functions. The original images are displayed in the top row of Figure 7(a), and the images preprocessed by imadjust, histeq, adapthisteq, and our method are shown in the second to last rows of Figure 7(a) in turn. After training, our model had an error rate of 0.54% on the test set.

FIGURE 7. (a) Original and preprocessed images; (b) 68 errors of our model.

Figure 7(b) records all the errors of the two stacks tested on the first and second blocks. Under each subgraph, several notations were provided: correct labeling (left), and optimal recognition rates of first and second blocks (right). It can be seen that over 80% of correct recognitions are attributable to the training on the second block, and the probability of incorrect recognition was generally low. Overall, our method achieved a close-to-1 probability of traffic sign recognition. Only 1% of images (confidence < 0.51) were recognized at an error rate smaller than 0.24%. To further lower the error rate to 0.01% (by adjusting the number of blocks), it was learned from the experimental results that our method achieved the best performance at 4 blocks (the last row in Figure 7 (b)) [32], [33].

Table 1 reports the MAP, memory (MB), floating point operations per second (FLOPS), and millions of bits of each model.

TABLE 1. The performance of each model.

Method	MAP	Memory (MB)	FLOPS	Parameters (10)
CNN	78.62	4.51	837.54	15.55
SVM	81.53	8.3	269.90	12.35
R-FCN ResNet 101	90.08	18.15	956.82	43.12
Our model	95.77	6.71	62.78	26.54

As shown in Table 1, our model realized the best mAP, which is 5% higher than that of the most advanced baseline R-FCN ResNet101 and 14% higher than that of the classic SVM. The hard-earned advantages attribute to the extraction of ROIs, which enables our model to precisely locate the targets.

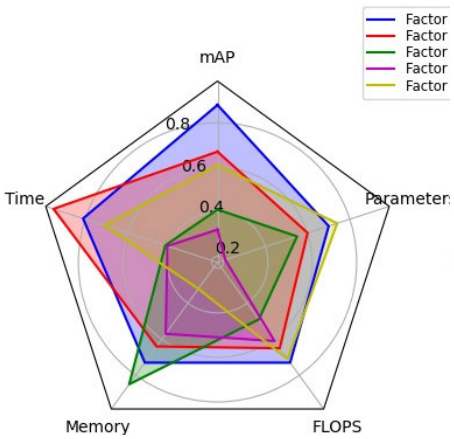

FIGURE 8. The radar chart of five performance indices.

The five metrics, namely, mAP, time, FLOPS, memory, and parameter, of all methods are plotted as Figure 8. Note that all values were converted into [0, 10], and only the maximum values of each metric were presented in the figure. Among the five metrics, mAP, time, and memory are more important than the other two metrics. It can be seen that our model consumed the shortest runtime among the five methods, thanks to the proposed preprocessing method. Therefore, the overall optimal model is our model [34], [35].

Table 2 further compares the recognition accuracy of each technique in our model. Clearly, every adaptational technique

TABLE 2. The recognition accuracy of each technique in our model.

Method	Accuracy
Original model	92.52
Original model + regional extraction	94.53
Original model + regional extraction + HOG feature	97.08

improved the recognition accuracy. This fully demonstrates the practicality of our model.

V. CONCLUSION

This study presents an automatic road sign recognition algorithm based on visual detection. The proposed algorithm includes ROI extraction, HOG feature design, and stacked CapsNet. The authors detailed the component of HOG feature, which could effectively obtain the pixel-level features of the image from any angle, and speed up the learning of the model. Experimental results on a real-world traffic sign dataset show that our model ran faster, occupied less memory, and required fewer parameters than baseline methods. The superiority over the classic CNN comes from the CapsNet, which fully utilizes images of different angles and directions with the aid of vectors.

REFERENCES

[1] M. S. Aminian, A. Allamehzadeh, M. Mostaed, and C. Olaverri-Monreal, "Cost-efficient traffic sign detection relying on smart mobile devices," in *Proc. Int. Conf. Comput. Aided Syst. Theory*, Feb. 2017, pp. 419–426.
[2] T. Wang, H. Shen, Y. Xue, and Z. Hu, "A traffic signal recognition algorithm based on self-paced learning and deep learning," *Ingénierie des systèmes d Inf.*, vol. 25, no. 2, pp. 239–244, May 2020.
[3] X. Yu, J. Yang, T. Wang, and T. Huang, "Key point detection by max pooling for tracking," *IEEE Trans. Cybern.*, vol. 45, no. 3, pp. 430–438, Mar. 2015.
[4] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising," *IEEE Trans. Image Process.*, vol. 26, no. 7, pp. 3142–3155, Jul. 2017.
[5] C.-A. Lameloise, H. Chauris, and M. Noble, "Improving the gradient of the image-domain objective function using quantitative migration for a more robust migration velocity analysis," *Geophys. Prospecting*, vol. 63, no. 2, pp. 391–404, Mar. 2015.
[6] K. Qian, L. Tian, Y. Liu, X. Wen, and J. Bao, "Image robust recognition based on feature-entropy-oriented differential fusion capsule network," *Appl. Intell.*, pp. 1–10, Sep. 2020.
[7] P. Bulla, L. Anantha, and S. Peram, "Deep neural networks with transfer learning model for brain tumors classification," *Traitement du Signal*, vol. 37, no. 4, pp. 593–601, Oct. 2020.
[8] U. Kamal, T. I. Tonmoy, S. Das, and M. K. Hasan, "Automatic traffic sign detection and recognition using SegU-net and a modified tversky loss function with L1-constraint," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 4, pp. 1467–1479, Apr. 2020.
[9] S. H. He, X. D. Li, Y. Wang, and H. H. Zhu, "An optimization model for automobile mixed assembly line under multiple constrains," *Int. J. Simul. Model.*, vol. 16, no. 4, pp. 720–730, Dec. 2016.
[10] D. Chen, "Multiple linear regression of multi-class images in devices of Internet of Things," *Traitement du Signal*, vol. 37, no. 6, pp. 965–973, Dec. 2020.
[11] S. S. S. Palakodati, V. R. Chirra, Y. Dasari, and S. Bulla, "Fresh and rotten fruits classification using CNN and transfer learning," *Revue d'Intelligence Artificielle*, vol. 34, no. 5, pp. 617–622, Nov. 2020.
[12] S. Wang, B. Yuan, and D. Wu, "A hybrid classifier for handwriting recognition on multi-domain financial bills based on DCNN and SVM," *Traitement du Signal*, vol. 37, no. 6, pp. 1103–1110, Dec. 2020.
[13] A. Ghorbanian, Y. Maghsoudi, and A. Mohammadzadeh, "Clustering-based band selection using structural similarity index and entropy for hyperspectral image classification," *Traitement du Signal*, vol. 37, no. 5, pp. 785–791, Nov. 2020.
[14] S. H. He, S. T. Zhang, X. M. Zhang, Z. X. Guo, C. X. Liu, and L. Chen, "Role of counselors in the cultivation of innovative talents from students' perspective," *Educ. Sci., Theory Pract.*, vol. 18, no. 6, pp. 3713–3723. 2018.
[15] M. Anandhalli and V. P. Baligar, "An approach to detect vehicles in multiple climatic conditions using the corner point approach," *J. Intell. Syst.*, vol. 27, no. 3, pp. 363–376, Jul. 2018.
[16] L. L. Li, "Image edge detection algorithm and its application in traffic video segmentation," Chongqing Jiaotong Univ., Chongqing, China, Tech. Rep., 2015.
[17] Y. Jin, Y. Fu, W. Wang, J. Guo, C. Ren, and X. Xiang, "Multi-feature fusion and enhancement single shot detector for traffic sign recognition," *IEEE Access*, vol. 8, pp. 38931–38940, 2020.
[18] T. Y. Zheng, N. Wang, and J. Tang, "Traffic signs detection based on ICF and multi-class classifier," Transducer Microsyst. Technol., Tech. Rep., 2017.
[19] J. Gim, M. Hwang, B. C. Ko, and J. Y. Nam, "Real-time speed-limit sign detection and recognition using spatial pyramid feature and boosted random forest," in *Proc. Int. Conf. Image Anal. Recognit.* Cham, Switzerland: Springer, 2015, pp. 437–445.
[20] S. Z. Jumani, F. Ali, I. A. Kandhro, Q. A. Lakhan, U. Ali, M. W. Haroon, and S. Ahmed, "Facial emotion identification based on local binary pattern feature detector," *Indian J. Sci. Technol.*, vol. 12, p. 28, Jul. 2019.
[21] Y. Zhao, L. Gong, B. Zhou, Y. Huang, and C. Liu, "Detecting tomatoes in greenhouse scenes by combining AdaBoost classifier and colour analysis," *Biosystems Eng.*, vol. 148, pp. 127–137, Aug. 2016.
[22] F. Shao, X. Wang, F. Meng, T. Rui, D. Wang, and J. Tang, "Real-time traffic sign detection and recognition method based on simplified Gabor wavelets and CNNs," *Sensors*, vol. 18, no. 10, p. 3192, Sep. 2018.
[23] Y. Zhu, C. Zhang, D. Zhou, X. Wang, X. Bai, and W. Liu, "Traffic sign detection and recognition using fully convolutional network guided proposals," *Neurocomputing*, vol. 214, pp. 758–766, Nov. 2016.
[24] J. Lu, Y. Qu, and X. Yang, "Traffic sign recognition with inception convolutional neural networks," in *Proc. Int. Conf. Internet Multimedia Comput. Service*. Singapore: Springer, 2017, pp. 487–494.
[25] W. Li, R. Dong, H. Fu, and A. L. Yu, "Large-scale oil palm tree detection from high-resolution satellite images using two-stage convolutional neural networks," *Remote Sens.*, vol. 11, no. 1, p. 11, Dec. 2018.
[26] J. Li, S. Luo, Z. Zhu, H. Dai, A. S. Krylov, Y. Ding, and L. Shao, "3D IoU-net: IoU guided 3D object detector for point clouds," 2020, *arXiv:2004.04962*. [Online]. Available: http://arxiv.org/abs/2004.04962
[27] L. Deng, Y. Gong, Y. Lin, J. Shuai, X. Tu, Y. Zhang, Z. Ma, and M. Xie, "Detecting multi-oriented text with corner-based region proposals," *Neurocomputing*, vol. 334, pp. 134–142, Mar. 2019.
[28] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 658–666.
[29] C. Huang, D. Chen, and X. Tang, "Implementation of workpiece recognition and location based on opencv," in *Proc. 8th Int. Symp. Comput. Intell. Design (ISCID)*, vol. 2, Dec. 2015, pp. 228–232.
[30] C. M. Fernandes, A. M. Mora, J. J. Merelo, and A. C. Rosa, "KANTS: A stigmergic ant algorithm for cluster analysis and swarm art," *IEEE Trans. Cybern.*, vol. 44, no. 6, pp. 843–856, Jun. 2014.
[31] E. Park, J. Ahn, and S. Yoo, "Weighted-entropy-based quantization for deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5456–5464.
[32] S. Fong, K. Cho, O. Mohammed, J. Fiaidhi, and S. Mohammed, "A time series pre-processing methodology with statistical and spectral analysis for classifying non-stationary stochastic biosignals," *J. Supercomput.*, vol. 72, no. 10, pp. 3887–3908, Oct. 2016.
[33] W. Cho and E. Choi, "Big data pre-processing methods with vehicle driving data using MapReduce techniques," *J. Supercomput.*, vol. 73, no. 7, pp. 3179–3195, Jul. 2017.
[34] M.-Y. Chen and B.-T. Chen, "A hybrid fuzzy time series model based on granular computing for stock price forecasting," *Inf. Sci.*, vol. 294, pp. 227–241, Feb. 2015.
[35] S. Zhang, X. Li, and C. Zhang, "A fuzzy control model for restraint of bullwhip effect in uncertain closed-loop supply chain with hybrid recycling channels," *IEEE Trans. Fuzzy Syst.*, vol. 25, no. 2, pp. 475–482, Apr. 2017.

SHOUHUI HE received the Ph.D. degree in traffic and transportation engineering from the Wuhan University of Technology, Wuhan, China, in 2014.

He worked with Linyi Transportation Bureau from 2014 to 2017. Since 2017, he has been working with the School of Logistics, Linyi University, Linyi, China. He is currently a Professor and a Master Supervisor. His research interests include transportation safety, transportation emergency management, analysis and application of intelligent logistics technology, and transportation system safety engineering.

Dr. He is the Registered Safety Engineer of the China Association of work safety and a member of China Logistics Society. He is also the Excellent Instructor of Linyi University.

LEI CHEN was born in Linyi, Shandong, China, in 1989. He received the B.S. degree in mathematics and applied mathematics from Zhengzhou University, Zhengzhou, China, in 2010, and the Ph.D. degree in transportation planning and management from Beijing Jiaotong University, Beijing, China, in 2016.

From 2016 to 2020, he was a Research Assistant with the School of Logistics, Linyi University, where he has been an Assistant Professor, since 2021. He is the author of more than 15 articles. His research interests include traffic safety management, transportation planning and management, analysis and application of intelligent logistics technology, transportation network optimization and design, and low-carbon logistics and transportation.

Dr. Chen's awards and honors include the Member of China Association of Work Safety, the Member of China Logistics Society, and the Award for Science and Technology Advancement of China Railway Society.

SHAOYUN ZHANG received the Ph.D. degree in business administration from Korea University, in 2012.

Since 2012, she has been a Teacher with the School of logistics, Linyi University, Shandong, China. Her research interests include regional logistics, business logistics, and cross-border logistics.

Dr. Zhang's awards and honors include the Published ten articles, one monograph, one social science plan in Shandong Province, one national college student innovation and entrepreneurship training program, one school level key project, and many times instructed students to win the First Class National Discipline Competitions Award, and the First Prize of Provincial competition. And instruct students to publish five articles in national excellent journals.

ZHUANGXIAN GUO was born in Ledu, Qinghai, China, in 1976. He received the B.S. degree in computer and application from the Xi'an University of Technology, Shanxi, China, in 1999.

From 2006 to 2011, he was a Research Assistant with the Financial Department, Linyi University, where he has been a Research Assistant with the School of Logistics, since 2011. He is the author of one book and more than three articles. His research interests include traffic safety management, application of intelligent logistics technology, and transportation planning and management.

Mr. Guo is a member of the China Association of Work Safety and the China Logistics Society.

PENGJIE SUN was born in Qingdao, Shandong, China, in 1973. He received the B.S. degree in business management from Qingdao University, Qingdao, China, in 1994, and the master's degree in business management from Huaqiao University, Quanzhou, China, in 2004, where he is currently pursuing the Ph.D. degree.

From 2004 to 2020, he was a Research Assistant with the School of Logistics, Linyi University, where he has been an Assistant Professor, since 2015. He is the author of more than 11 articles. His research interests include enterprise growth, innovation management, business model innovation, and platform enterprise growth.

Dr. Sun's awards and honors include a Member of Expert database of philosophy and Social Sciences, a member of the China Logistics Society, the Award for Science, and the Technology Award of China Federation of logistics and purchasing.

HONG LIU was born in Linyi, Shandong, China, in 1976. She received the bachelor's degree in economics from the Xi'an Institute of Statistics (Now the Xi'an University of Finance and Economics), Xi'an, China, in 1999, and the master's degree in economics from Nanjing Normal University, Nanjing, China, in 2006.

From 1999 to 2020, she was a Full-Time Teacher with Linyi University. Since 2016, she has been an Assistant Professor with the School of Logistics, Linyi University. Her research interests include economic theory, rural logistics, and cold chain logistics. She is the author of eight articles and two invention patents. She was awarded an excellent instructor in the Nineth National College Students market research and Analysis Competition.

HONGDA LIU received the bachelor's degree in business administration from the Qingdao University of Science and Technology, Qingdao, China, in 2004, the master's degree in international trade from Korean Changwon International University, Changwon, South Korea, in 2008, and the Ph.D. degree in economics from Pusan National University, Pusan, South Korea, in 2012. Since 2012, he has been working as an Officer of the Linyi Government, Linyi China. His research interests include finance, economics, and social government.