

# Hierarchical Traffic Sign Recognition

Yanyun Qu<sup>(✉)</sup>, Siying Yang, Weiwei Wu, and Li Lin

Computer Science Department, Xiamen University, Xiamen, China  
quyanyun@gmail.com

**Abstract.** Traffic Sign Recognition (TSR) is very important for driverless systems and driver assistance systems. Because of the large number of the traffic sign classes and the unbalanced training data, we propose a hierarchical recognition method for traffic sign recognition. A classification tree is constructed, where the non-leaf node is constructed based on shape classification with aggregated channel features and a leaf node is constructed based on random forest classifiers with histogram of gradient for multi-class traffic sign recognition in the non-leaf node. The proposed method can overcome the inefficiency of flat classification scheme and imbalance of training data. Extensive experiments are done on three famous traffic sign datasets: the German Traffic Sign Recognition Benchmark (GTSRB), Swedish Traffic Signs Dataset (STSD), and the 2015 Traffic Sign Recognition Competition Dataset. The experimental results demonstrate the efficiency and effectiveness of our methods.

## 1 Introduction

Traffic Sign Recognition (TSR) is an active topic in the intelligence traffic systems. It is important for automatic driverless vehicle and driver assistant systems [1]. For example, driver assistance systems could warn drivers to take strategies ahead of time to avoid accidents [2]. The task of traffic sign recognition usually contains two main stages: traffic sign detection (TSD) and traffic sign classification (TSC). Traffic sign detection aims at locating the position of the traffic signs accurately in an image or each frame of video. Traffic sign classification focuses on labeling a traffic sign. Although the two stages may overlap such as feature representation of traffic sign, they are usually studied independently. In this paper, we focus on the second task which is usually named traffic sign recognition.

Traffic sign recognition is challenging due to the complicated dynamic nature scene, which faces four difficulties: (1) Appearances of traffic signs change with variations of viewpoint and illumination, weather condition like rain or fog, motion-blur during driving, occlusions, physical damage, colors fading, graffiti, stickers and so on. (2) Traffic sign recognition should be of real time speed and high recognition accuracy for the purpose of the practical application. (3) The training data are unbalanced. The frequencies of occurrences of traffic signs are different greatly. For example, the speed limit signs appear more frequently than the no-entry signs in the German Traffic Sign Recognition Benchmark (GTSRB) [3] and the Swedish Traffic Signs Dataset (STSD) [4]. (4) The number of traffic sign classes is large. For example, there are 112 important warning sign templates in Chinese traffic signs.

There are many literatures to deal with the first two difficulties. Some popular machine learning methods are implemented on traffic sign recognition, such as Bayesian classifiers [5], boosting [6], support vector machines (SVM) [7], and random forest classifier [8]. These methods used hand-crafted features such as Histogram of Oriented Gradient (HOG) [9] and Scale-invariant feature transform (SIFT) [6–8]. In [10], Zaklouta used a tree classifier of K depth combined with HOG as well as the distance transforms. Maldonado [11] designed a recognition system based on SVMs, whose results showed high recognition accuracy and a very low false positive rate. Convolutional Neural Network (CNN) is used for traffic sign recognition [3, 12, 13] and achieved high recognition accuracy. Recently, CNN is hot in the field of computer vision. It has achieved several state-of-the-art performances in ILSVRC2012 [14–16] and the 2011 International Joint Conference on Neural Networks (IJCNN) competition [3, 4, 17].

However, few work discussed how to deal with the imbalance of training data and how to improve the efficiency of multi-class prediction for traffic sign recognition. Most of the current multi-class prediction schemes are flat, that is, a one-vs.-all or a one-vs.-one classification scheme is used for label prediction. The flat classification scheme is time-consuming. Moreover, the imbalance of training data has negative influence on the classification performance. As we know, the traffic signs are man-made signs of special shapes, which can be divided three shape classes: circle, triangle, and square. We construct a tree structure of two layers for traffic sign. The first layer contains the coarse shape classes and the second layer contains the fine classes which is the traffic sign identification. Thus, we propose a hierarchical traffic sign recognition method. The advantage of our method is to improve the efficiency of traffic sign recognition.

The paper is organized as follows. Section 2 introduces the hierarchical recognition method for traffic sign recognition. Section 3 introduces the experimental results. Conclusion are given in Sect. 4.

## 2 Hierarchical Class Prediction Algorithm

In this section, we detail the implementation of the hierarchical traffic recognition. Figure 1 shows the framework of our method. In the training stage, a classification tree  $G = (V, E)$  is constructed which has two layers. In the first layer, the traffic signs are divided into three groups based on the Adaboost classifier combined with Aggregate Channel Features (ACF) [18]. The non-leaf nodes are the shape nodes in the first layer, and in the second layer, each node contains traffic sign identification. A leaf node is identified by a random forest classifier which is learned on the data of classes contained in its parent node. In the testing stage, a query traffic sign image will traverse the classification tree. In each layer, the query image is given to the node with the maximum confidence value. Finally, the leaf node label with the maximum confidence value is regarded as the label of traffic sign.

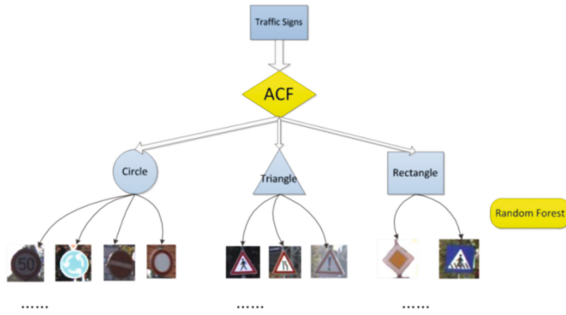


Fig. 1. The framework of the hierarchical recognition for traffic signs

### 2.1 Building the Non-leaf Node for Shape Classification

In this subsection, we introduce the construction of the non-leaf nodes based the shape classification. Aggregated channel features (ACF) [18] are proved to be useful for pedestrian detection with high speed and detection accuracy [19]. Motivated by the success in pedestrian detection, we use ACF for feature representation of traffic signs. The basic structure of the aggregated channel features is channel. We use ten channels: three color channels of the image with RGB color space, the gradient magnitudes, the six oriented gradient maps. Figure 2 shows the ACF used in our method. In our implementation, six oriented gradient filters are used: horizontal, vertical, 30°, 60°, 120°, and 150°. A traffic sign image is firstly normalized into a 10 × 10 image. And then its ACF are computed. We use all the obtained map for training an Adaboost classifier.

For the shape classification in the first layer, we adopt the Adaboost framework of Viola and Jones (VJ) framework [20]. As we know, the Adaboost classifier contains many weak classifiers called weak learners which can be combined into a strong classifier. It has been proven to converge to the optimal solution with a sufficient number of weak classifiers. AdaBoost assigns weights to weak classifiers based on their quality, and the resulting strong classifier is a linear combination of weak classifiers with the appropriate weights.

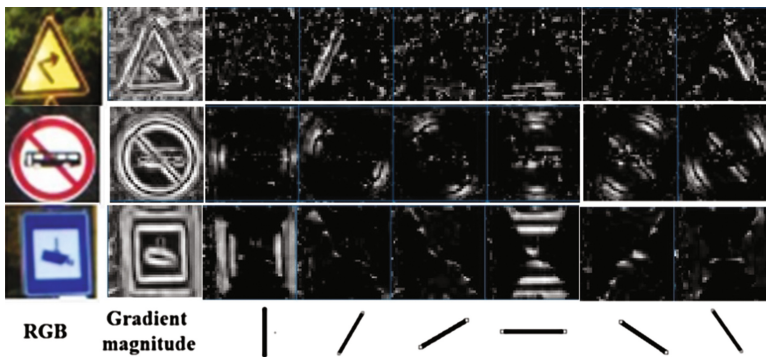


Fig. 2. Aggregated channel features for shape classification

We use depth-2 decision trees for boosting [21], where each node is a simple decision stump, defined by rectangular region, a channel, and a threshold [23]. We carry out the VJ framework and the final classifier is a weighted linear combination of boosted depth-2 decision trees of weak classifier. Because each weak classifier is a depth-2 decision tree, it implements only two comparing operations to apply a weak classifier, so the shape classification is quite fast.

## 2.2 Building the Leaf Node for Traffic Sign Identification

In this subsection, we detail how to build the leaf node based on random forest classifier. Because each shape node contains several traffic sign classes, we build a random forest classifier for the traffic sign classes contained in a shape node. In order to train a random forest classifier, each training sample is normalized to a  $40 \times 40$  image. If a shape node contains  $N$  classes of traffic signs, the samples from the  $N$  classes of traffic signs are used to train the random forest classifier, and each leaf node contains a traffic sign class. In order to train a random forest classifier, we use multiple features which include the following features: Histogram of Oriented Gradients (HOG), Local Binary Pattern (LBP), and HSV.

**HOG:** An image is converted from the RGB color space to gray scale space. And then it is divided into  $7 \times 7$  blocks and each block contains 4 cells. In each cell, a gradient oriented histogram with 9 bins is computed. Thus, HOG is 1764-dimensional.

**LBP:** Just like HOG, an image firstly is transformed to gray scale space. LBP has low computing complexity with rotation invariant and gray scale invariance performances. In this paper, a 256-dimensional LBP descriptor is used.

**HSV:** Because RGB color space is very sensitive to illumination, HSV color space is used in this paper. An image is firstly converted to HSV color space. For each pixel, values of hue and saturation are scaled to the range  $[0, 255]$ . For the H channel and S channel, the two components of two similar colors are numerically much closer, thus, HSV is less sensitive to illumination. A histogram is computed for each channel, and the two histograms are concatenated into a vector of 512 dimensions, which is treated as the color feature.

The tree types of features are concatenated and form a 2532-dimensional vector. In our experiments, 500 trees are used to form a random forest classifier. The prediction label is predicted by the ensemble learning of all the trees.

## 2.3 Class Prediction Scheme

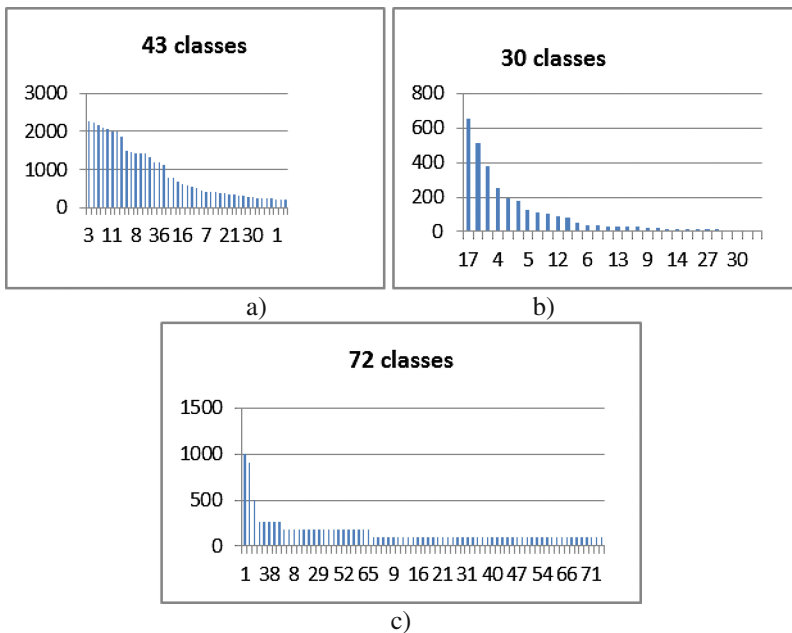
If a query image is input, it traverses the classification tree. In the first layer, it can be scored by the classifiers in shape nodes and the shape node is retained whose score is the maximum among the three shape nodes. And then, the query image is scored by the random forest classifier in the retained shape node. The label of traffic sign whose score is the maximum is given to the query image.



enhance the number of training samples. Samples are randomly perturbed in position ( $[-0.2, 0.2]$  pixels), in scale ( $[0.1, 5]$  ratio) and rotation ( $[-10, +10]$  degrees).

### 3.1 The Imbalance of Traffic Sign Classes

We analyze the distribution of training data for the three traffic sign datasets. The histograms of the class frequencies are given in Fig. 4. It demonstrates that the imbalance of traffic sign classes exist in all the three dataset. The plots of histogram of the class frequencies are of long tails. In GTSRB, the biggest class set contains more than 2000 samples while the smallest class set contains only dozens of samples. In Swedish30, the biggest set contains about 600 samples, while the smallest set contains only several samples. In Mutil-72TSD, the biggest set contains about 1000 samples while the smallest set contains only dozens of samples. The imbalance of training data has negative influence on the classification performance. It implies that our method is required for multi-class classification.



**Fig. 4.** The histogram of the class frequencies in traffic sign dataset. The x-axis is the label of each class; the y-axis is the number of samples in train dataset. (a) the histogram for GTRSB. (b) the histogram for Swedish30. (c) The histogram for Mutil-72TSD.

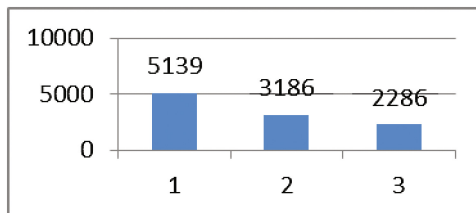
### 3.2 The Analysis of Computational Complexity

The main computational complexity of our method comes from the complexity of building the decision tree. Building the random forest model is an ensemble method, so it's going to be close to the sum of the complexities of building the individual decision trees in the model. If each model has the same complexity, then it would be the complexity of the individual model times the number of models you build. If having  $n$  instances and  $m$  attributes, the computational cost of building a tree is  $O(mn \log n)$ . If growing  $M$  trees, then the complexity is  $O(M(mn \log n))$ . This is not an exact complexity, because the trees in the model are grown using a subset of the features, and additional time may be added in to handle the randomization processes. However, this would get close to the complexity. The parameters here are  $n$ ,  $m$ , and  $M$  - the of instances in the training data, the number of attributes, and the number of trees you build. The number of trees is a parameter you set yourself when you run the model [24].

We compare the flat classification scheme with the proposed hierarchical scheme in terms of the computational complexities. We give the number of classes contained in shape nodes in Table 1. Take the GTSRB for example. This database contains 43 classes, in which 26 classes are circle, 16 classes are triangle and 1 class is rectangle. For the flat classification scheme, 43 classifiers are used and the label with the largest score is given to the query image. Instead, for our method, 3 shape classifiers and 16 classifiers identifying traffic signs are used, thus, the total number of classifier is 19, which is more time-saving. Moreover, in the training stage, all data are loaded for training, while our method does not load all data for all classifiers, that is, the classifiers in the leaf nodes do not load all data. We also show the distribution of training data of Mutil-72TSD in the non-leaf nodes in Fig. 5. The horizontal axis denotes the shape nodes, and the vertical axis denotes the number of samples in each shape node. It demonstrates that our method can overcome the imbalance of classes.

**Table 1.** The class number in the classification tree for the three datasets

Classifiers		GTSRB	Swedish30	Mutil72
Flat		43	30	72
Hierarchical	Shape nodes	3		
	Circle	26	22	29
	Triangle	16	5	19
	Rectangle	1	3	25



**Fig. 5.** The distribution of classes in the shape nodes for Mutil-72.

### 3.3 The Performance of the Proposed Method

In this subsection, we estimate the performance of the proposed method in term of the recognition accuracy. We first implement our method on GTSRB, which is famous because of the 2011 IJCNN traffic sign competition. In Table 2, we compare the methods of traffic sign recognition in terms of recognition accuracy. For the purpose of fair comparison, we only compare the method based on tree classifiers and SVM classifiers with HOG. The result demonstrates that our method is superior to other tree based methods. We also compare our method with the CNN [22] method trained on GTSRB. The result proves that our method performs CNN method with the same amount of train samples.

In Swedish30, we compare our method with the random forest classifier used in [25] which discusses the performances of HOG and LBP in different color channels. Table 3 shows the comparison results in which our result is the average of five testing results. HH means three HOG is computed in the three channels of HSV color space and the feature vector is the concatenation of the three channels of HOG. HL means that LBP is computed in the three channels of HSV color space, and H+L means that the histogram of color in HSV color space and LBP histogram are concatenated to form a feature vector. Table 3 demonstrates that our method can achieve the comparable results while the dimension of features used in our method is lower than the comparison method [25].

We also implement the proposed method on Multi-72STD. The results are shown in Table 4. Furthermore, we compare the flat classification scheme with the hierarchical classification method. In Table 4, we give the comparison results on the three datasets. We also compare the flat classification scheme with the hierarchical classification scheme.

**Table 2.** Comparison of traffic sign recognition in GTSRB

Method	Accuracy (%)
Ours	95.97
CNN	92.82
HOG_02-L2RL2SVM <sup>a</sup>	95.89
logistic23 <sup>a</sup>	95.82
Linear SVM <sup>a</sup>	95.81
RandomForest_Hog_Hue_100tree <sup>a</sup>	94.74
KdTree+HOG 9720 KNN 5 <sup>a</sup>	93.40
KdTree+HOG 2520 <sup>a</sup>	92.90
Random trees HOG3 <sup>a</sup>	92.13

<sup>a</sup>All the compared methods come from the 2011 IJCNN. <http://benchmark.ini.rub.de>

**Table 3.** Comparison of traffic sign recognition in Swedish30.

Method	HH	HH+HL	H+L	Ours
Accuracy (%)	97.69	97.49	97.91	97.94



**Table 4.** Comparison between the flat classification and the hierarchical classification

Method	GTRSB Final_test (%)	GTRSB Online_test (%)	Swedish30 (%)	Mutil-72TSD (%)
Random forest	95.20	–	97.49	86.68
Ours	95.97	99.87	97.94	88.87

## 4 Conclusion

In this paper, we focus on traffic sign recognition in a hierarchical classification scheme. A classification tree is firstly constructed, in which the non-leaf nodes are constructed based on shape classification and the leaf nodes are constructed based on traffic sign identification. For the shape classification, aggregated channel features are used for feature representation of a traffic sign image and the Adaboost classifier based on weak decision tree are used for shape classification. In each shape node, a random forest classifier is trained based on HOG. The proposed method can overcome the inefficiency of flat classification scheme and the imbalance of the training data. The proposed method is implemented on three famous traffic sign recognition datasets and the experimental results demonstrate the efficiency and effectiveness of our method.

## References

1. Mogelmoose, A., Trivedi, M.M., Moeslund, T.B.: Vision-based traffic sign detection and analysis for intelligent driver assistance systems: perspectives and survey. *IEEE Trans. Intell. Transp. Syst.* **13**(4), 1484–1497 (2012)
2. Cai, Z., Gu, M.: Traffic sign recognition algorithm based on shape signature and dual-tree complex wavelet transform. *J. Central South Univ.* **20**, 433–439 (2013)
3. Cireşan, D., Meier, U., Masci, J., et al.: A committee of neural networks for traffic sign classification. In: *The 2011 International Joint Conference on Neural Networks (IJCNN)*, pp. 1918–1921. IEEE (2011)
4. Stallkamp, J., Schlipsing, M., Salmen, J., et al.: The German traffic sign recognition benchmark: a multi-class classification competition. In: *The 2011 International Joint Conference on Neural Networks (IJCNN)*, pp. 1453–1460. IEEE (2011)
5. Meuter, M., Nunn, C., Görmer, S.M., et al.: A decision fusion and reasoning module for a traffic sign recognition system. *IEEE Trans. Intell. Transp. Syst.* **12**(4), 1126–1134 (2011)
6. Ruta, A., Li, Y., Liu, X.: Robust class similarity measure for traffic sign recognition. *IEEE Trans. Intell. Transp. Syst.* **11**(4), 846–855 (2010)
7. Greenhalgh, J., Mirmehdi, M.: Real-time detection and recognition of road traffic signs. *IEEE Trans. Intell. Transp. Syst.* **13**(4), 1498–1506 (2012)
8. Zaklouta, F., Stanculescu, B.: Real-time traffic-sign recognition using tree classifiers. *IEEE Trans. Intell. Transp. Syst.* **13**(4), 1507–1514 (2012)
9. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005*, vol. 1, pp. 886–893. IEEE (2005)
10. Zaklouta, F., Stanculescu, B., Hamdoun, O.: Traffic sign classification using KD trees and random forests. In: *The 2011 International Joint Conference on Neural Networks (IJCNN)*, pp. 2151–2155. IEEE (2011)

11. Maldonado-Bascon, S., Lafuente-Arroyo, S., Gil-Jimenez, P., et al.: Road-sign detection and recognition based on support vector machines. *IEEE Trans. Intell. Transp. Syst.* **8**(2), 264–278 (2007)
12. LeCun, Y., Bottou, L., Bengio, Y., et al.: Gradient-based learning applied to document recognition. *Proc. IEEE* **86**(11), 2278–2324 (1998)
13. Sermanet, P., LeCun, Y.: Traffic sign recognition with multi-scale convolutional networks. In: *The 2011 International Joint Conference on Neural Networks (IJCNN)*, pp. 2809–2813. IEEE (2011)
14. He, K., Zhang, X., Ren, S., et al.: Deep residual learning for image recognition. *arXiv preprint [arXiv:1512.03385](https://arxiv.org/abs/1512.03385)* (2015)
15. Russakovsky, O., Deng, J., Su, H., et al.: Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.* **115**(3), 211–252 (2015)
16. Szegedy, C., Vanhoucke, V., Ioffe, S., et al.: Rethinking the inception architecture for computer vision. *arXiv preprint [arXiv:1512.00567](https://arxiv.org/abs/1512.00567)* (2015)
17. Jin, J., Fu, K., Zhang, C.: Traffic sign recognition with hinge loss trained convolutional neural networks. *IEEE Trans. Intell. Transp. Syst.* **15**(5), 1991–2000 (2014)
18. Dollár, P., Appel, R., Belongie, S., et al.: Fast feature pyramids for object detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **36**(8), 1532–1545 (2014)
19. Benenson, R., Omran, M., Hosang, J., Schiele, B.: Ten years of pedestrian detection, what have we learned? In: *Agapito, L., Bronstein, Michael M., Rother, C. (eds.) ECCV 2014. LNCS, vol. 8926*, pp. 613–627. Springer, Heidelberg (2015). doi:[10.1007/978-3-319-16181-5\\_47](https://doi.org/10.1007/978-3-319-16181-5_47)
20. Viola, P., Jones, M.: Robust real-time object detection. *Int. J. Comput. Vis.* **57**(2), 137–154 (2007)
21. Dollár, P., Tu, Z., Perona, P., et al.: Integral channel features. In: *Proceedings of British Machine Vision Conference, BMVC 2009, London, UK, 7–10 September 2009* (2009)
22. Wang, T., Wu, D.J., Coates, A., et al. End-to-end text recognition with convolutional neural networks. In: *2012 21st International Conference on Pattern Recognition (ICPR)*, pp. 3304–3308. IEEE (2012)
23. Mathias, M., Timofte, R., Benenson, R., et al.: Traffic sign recognition—how far are we from the solution? In: *The 2013 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8. IEEE (2013)
24. Biau, G.Á.Š.: Analysis of a random forests model. *J. Mach. Learn. Res.* **13**, 1063–1095 (2012)
25. Yang, X., Qu, Y., Fang, S.: Color fused multiple features for traffic sign recognition. In: *Proceedings of the 4th International Conference on Internet Multimedia Computing and Service*, pp. 84–87. ACM (2012)