TRAFFIC SIGN CLASSIFICATION USING TWO-LAYER IMAGE REPRESENTATION

Yingying Zhu, Xinggang Wang, Cong Yao, Xiang Bai

Huazhong University of Science and Technology, Wuhan, P.R. China

ABSTRACT

This paper makes use of locality-constrained linear coding (LLC) in a two-layer image representation framework for traffic sign recognition. As a multi-category classification problem with unbalanced frequencies and variations, many machine learning approaches have been adopted with some low level features for traffic sign recognition. To the best of our knowledge, this is the first method using coding features for traffic sign recognition. First, we extract features(dense SIFT features, HOG features and LBP features) and encode them with a k-means generated codebook and LLC. Second, each traffic sign image is represented by the features generated by spatial pyramid matching (SPM). Then, all the image representations from each kind of features are concatenated together as the final image representation. Finally, we show that a linear SVM classifier trained with this image representation can achieve the state-of-the-art recognition rate of 99.67% on the well-known German Traffic Sign Recognition Benchmark.

Index Terms— Locality-constrained Linear Coding, Spatial Pyramid Matching, Traffic Sign Recognition

1. INTRODUCTION

Traffic sign recognition is a challenging real-world problem of high industrial relevance. There are some main difficulties inevitably in traffic sign recognition: (1) Traffic signs may be similar within or across classes. (2) Class frequencies may be unbalanced. (3) Variable illumination may make the color change due to weather conditions. (4) Traffic signs may be faded or be dirty. (5) Traffic signs may be bent, and therefore, are not perpendicular to the road or trajectory of the vehicle. (6) Partial occlusions may make the traffic signs incomplete. (7) The scales of the traffic signs may change significantly. (8) The images collected by a moving vehicle may be prone to motion blur.

Many studies have been presented for solving this problem using several types of features [1], such as HOG features [2], Haar-like features, Hue histograms, etc. A wide range of state-of-the-art machine learning methods were employed [3], including Support Vector Machine, linear discriminant analysis, subspace analysis, ensemble classifiers, slow feature analysis, kd-trees, random forests, and k-means [4]. In this paper, dense SIFT [5] features, HOG [6] features and LBP [7] features are extracted from each image. Then, we encode the features using a k-means generated codebook and locality-constraint linear coding (LLC) [8] is introduced to preserve both locality and sparsity. Next, spatial pyramid matching (SPM) [9] are utilized as feature representations to convey statistical as well as spatial information of traffic signs images. SPM is the most efficient and effective extension of the bag-of-features (BoF) method. The SPM method partitions image into increasingly finer spatial sub-regions, and computes histograms of local features for each sub-region. Typically, $2^l \times 2^l$ sub-regions, $l = 0, 1, 2$ are used. Other pooling methods, such as feature context [10], can also be adopted. The coded features are concentrated and form a final descriptor vector of the image, which is fed into a SVM [11] classifier to predict the class of the traffic signs.

The two-layer framework is firstly proposed by Yang etal in [12] for object recognition. It is also suitable for traffic sign recognition, since (1) local features (dense SIFT features, HOG features and LBP features) can capture local texture and shape of traffic signs; (2) LLC can map local features into a high-dimensional space, in which features are more easier to be classified; (3) pooling strategy can obtain salient features for coding and reduce noise; (4) SPM can capture spatial information among local features of traffic signs. Within this framework, we fuse three local features (SIFT, HOG and LBP), which obviously boost the the traffic sign image classification performance.

The remainder of the paper is organized as follows: Section 2 introduces our traffic signs classification method based on LLC coding; Section 3 presents the experiments and evaluations; Section 4 concludes our method.

2. METHODS

Various of features and learning methods have been proposed in previous work to address the problem of traffic sign recognition. Most of works directly fed global features like HOG features, Haar-like features etc into classifiers like SVM and Random Forest. Other works use convolutional neural network (CNN) [13] to learn features on raw images have obtained very impressive results on this task. Different from the previous methods, we proposed a two-layer framework to represent traffic sign image; in the first layer, local image descrip-

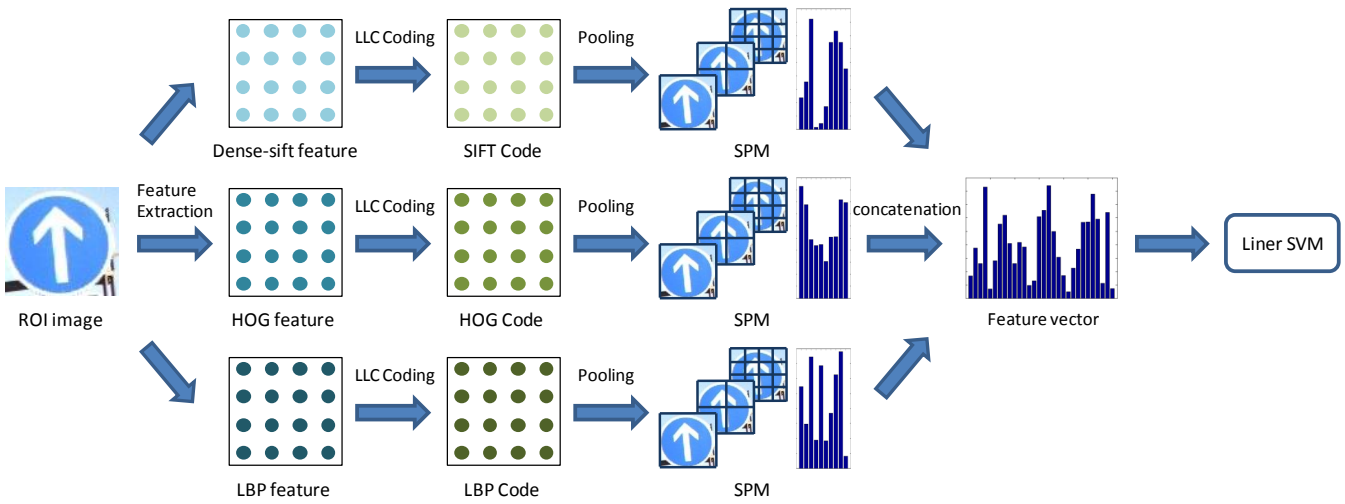

Fig. 1. Flowchart of our approach. Three kinds of local image descriptors (SIFT, HOG and LBP) are extracted; then they are encoded via LLC and their codes are pooled via SPM individually; final image representation combines the three individual image representations.

tors are encoded into texture codes; in the second layer, image codes are pooled with spatial pyramid to form a compact image presentation; and then, all the image representations are concatenated together as the final image representation; based on this image representation, traffic sign recognition task can be easily done using a simple linear SVM classifier. The whole traffic sign recognition system is demonstrated in Figure 1. In the follow subsections, we will give details of all components of this system.

2.1. Local descriptor

We extract dense SIFT features in image I. For example, for every 2 pixel in row and column, we extract a 12×12 image patch. To describe every image patch, we build a histogram based on gradient of pixels follow the way in [5] and denote the histograms as $I = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$. These dense SIFT features capture all local cues in image. To deal with variation of object scale, we can extract image patches in multiple scales. Since some regions in image may be useless for recognition, we use feature pooling in the second layer in our approach to select the most salient local features for image representation.

We also extract HOG features and LBP features in image I as we extract dense SIFT features. For every 3 pixel in row and column, we extract a 6×6 image patch. The three kinds of features are individually encoded and pooled given in Section 2.2 and 2.3.

2.2. The first layer: local-constrained linear coding

Taking SIFT feature for example, in the first layer, we map SIFT features into a high-dimensional space; in this space, features are much easier to be classified. Bases of the high-dimensional space is spanned by a codebook learned using k-means over randomly selected features extracted from training images. The mapping process is called coding here. A naive way for coding is vector quantization (VQ); it assign every feature to its nearest code in the codebook. Later researchers found that sparse coding (SC) is very effective for feature coding [12]; but original SC is very time-consuming. Locality-constrained linear coding (LLC) is an adaptation of SC with locality constraints. It is more efficient for feature coding. We show the formulation of LLC as following.

Given a codebook with M codewords, $\mathbf{B} = [\mathbf{b}_1, \ldots, \mathbf{b}_M] \in \mathbf{R}^{d \times M}$ and a dense SIFT feature $\mathbf{x}_i$, LLC follows the following criteria to obtain codes of $\mathbf{x}_i$ denoted as $\mathbf{z}_i \in \mathbf{R}^{M \times 1}$.

$$\min_{\mathbf{z}_i} \|\mathbf{x}_i - \mathbf{B}_i \mathbf{z}_i\|^2 + \lambda \|\mathbf{d}_i \odot \mathbf{z}_i\|^2 \qquad \text{s.t.} \quad \mathbf{1}^T \mathbf{z}_i = 1. \quad (1)$$

where λ is a weight parameter for adjusting the weight of locality, $\odot$ denotes element-wise multiplication, and $\mathbf{d}_i \in \mathbf{R}^{M \times 1}$ is locality adaptor that gives different freedom for each basis vector proportional to its similarity to the input descriptor $\mathbf{x}_i$. Specifically,

$$\mathbf{d}_i = \exp\left(\frac{\text{dist}(\mathbf{x}_i, \mathbf{B})}{\sigma}\right)$$

where $\text{dist}(\mathbf{x}_i, \mathbf{B}) = [\text{dist}(\mathbf{x}_i, \mathbf{b}_1), \ldots, \text{dist}(\mathbf{x}_i, \mathbf{b}_M)]^T$, and $\text{dist}(\mathbf{x}_i, \mathbf{b}_m)$ is the Euclidean distance between $\mathbf{x}_i$ and $\mathbf{b}_m$. σ is used for adjusting the weight decay speed for locality adaptor.

A fast approximated solution for Eq. 1 has been proposed in [8]. It first finds nearest neighbours of $\mathbf{x}_{ij}$ in $\mathbf{B}$, then computes codes by solving a linear system with the nearest neighbours.

2.3. The second layer: spatial pyramid pooling

In this layer we build a compact image representation on codes $\mathbf{z}_i$. First, we divide image into different regions. Specifically, shape image is divided into 1×1, 2×2 and

4×4 regions; in total, there are 21 regions. Then for every region $R_r, r \in [1, \ldots, 21]$, we do max-pooling. Let $\mathbf{z}^p$ denotes a encoded shape descriptor in the position of p in image. Max-pooling works as following:

$$\mathbf{f}_r = \max(\mathbf{z}^p | p \in R_r). \quad (2)$$

where the $\max$ function works in row-wise, returns a feature vector of R_r, $\mathbf{f}_r$, with the same size with $\mathbf{z}_i$. For each codeword, we take the max value of all codes in a region for image representation, so we called this method as max-pooling. Max-pooling is robust to noise and works well with linear classifier. For example, image representation $\mathbf{f}_{SIFT}$ for image I based on SIFT feature is a concatenation of the feature vectors for all regions.

$$\mathbf{f}_{SIFT} = [\mathbf{f}_1^T, \ldots, \mathbf{f}_{21}^T]^T. \quad (3)$$

2.4. Feature fusion

This kind of two-layer image representation is built from SIFT, HOG and LBP features individually, denoted as $\mathbf{f}_{SIFT}$, $\mathbf{f}_{HOG}$ and $\mathbf{f}_{LBP}$. Final image representation $\mathbf{f}_I$ is a concatenation of $\mathbf{f}_{SIFT}$, $\mathbf{f}_{HOG}$ and $\mathbf{f}_{LBP}$.

$$\mathbf{f}_I = [\mathbf{f}_{SIFT}^T, \mathbf{f}_{HOG}^T, \mathbf{f}_{LBP}^T]^T. \quad (4)$$

Finally, $\mathbf{f}_I$ is fed into linear SVM for training and testing.

3. EXPERIMENT

3.1. Databases

We have evaluated our classification performance on two standard databases: German Traffic Sign Recgonition Benchmark(GTSRB) [14] and KUL Belgium Traffic Sign Classification Benchmark (BelgiumTSC) [15]. The details about these databases are shown in Table 1.

Table 1. Information of the two public traffic sign databases

	GTSRB	BelgiumTSC
number of classes	43	62
number of images	39209	7095
sign sizes	15×15 to 250×250 px	100×100 to 1628×1236 px

The GTSRB is the one of the most widely used database, which has been presented in [1] and created for the competition, "The German Traffic Sign Recognition Benchmark." The competition was held at the International Joint Conference on Neural Networks (IJCNN) 2011. It is a large database containing 26640 training images and 12569 testing images. Image sizes vary between 15x15 to 250x250 pixels. The original images contain a border of 10% around the actual traffic signs (at least 5 pixels) to allow for edge-based approaches. We crop all images the bounding box of the traffic signs and resize them to 48×48 pixels for processing.

The BelgiumTSC is built for traffic sign classification purposes, which is a subset of BelgiumTS dataset and contains cropped images around annotations for 62 different classes of traffic signs. The BelgiumTSC is split into a training part with 4575 images and a testing part with 2520 images. As we deal with the GTSRB, we also crop all images of the BelgiumTSC the bounding box of the traffic sign and resize them to 48×48 pixels.

3.2. Experimental Details and Results

MATLAB is used to simulate our approach. We use a computer with Intel(R) Xeon(R) CPU E5-2650 (2.00GHz) without using graphics cards to accelerate.

The patch size of dense SIFT features is 12×12, and the grid size is 2. When generating codebook, we randomly select 200 dense SIFT features from each image to form the feature set. The number of clusters in k-means clustering for dense SIFT features is set to 1000. The patch size of HOG features and LBP features is 6×6, and the grid size is 3. The number of clusters in k-means clustering for HOG features and LBP features is set to 500. In SPM, we pool features in 3 layers, each layer contain 1, 4, 16 subregions respectively. When conducting approximated LLC, we choose 5 nearest neighbors. Linear SVM are used for classification our approach.

In order to better evaluate the performance of our method, we run the experiments for 10 times and report the average accuracy and standard deviation.

Table 2. Comparison of classification performances of different methods on the GTSRB

Method	Accuracy
LLC+SPM+SVM(best)	**99.67**%
LLC+SPM+SVM(Mean+STD)	**99.64 ± 0.018**%
Committee of CNNs [13]	99.46%
Human Performance(average)	98.84%
Multi-Scale CNNs [16]	98.31%
Random Forests [17]	96.14%

As shown in Table 2 , we report the classification performance of our approach and the three best-performing machine learning approaches in the GTSRB competition complemented with human performance. Team IDSIA won the GTSRB competition by using a committee of CNNS [13] trained on raw pixels. They only use the central ROIs containing the traffic signs and the regions are scaled to a size of 48×48 pixels. Sermanet and LeCun (2011) employed a multi-scale convolutional neural network (CNN or ConvNet) [16]. The gray images are scaled to a size of 32×32 pixels and used as input. In contrast to traditional CNNs, not only the output of the last stage but also all feature extraction stages are fed into the classifier. Team CAOR employed a Random Forest [17]

of 500 trees with the official HOG 2 dataset. A random subset of the training data is used to built the decision trees and the remaining data is used to estimate the classification error. Our approach achieves a correct classification rate of 99.67%, outperforming all the existing methods. In addition, the time cost of feature extraction is 0.0195 second per image, and the time cost of testing each image is only 0.0130 second.

Table 3. Comparison of classification performance of different methods using HOG feature and classification performance of different features using our method on the GTSRB

Feature	Accuracy
LDA on HOG 1 [3]	93.18%
LDA on HOG 2 [3]	95.68%
LDA on HOG 3 [3]	92.34%
HOG	98.31%
SIFT	99.55%
LBP	97.33%
SIFT+HOG+LBP	**99.67**%

In Table 3 , we report the classification performance of different methods using HOG feature. In the GTSRB competition, the results trained by linear discriminant analysis(LDA) is used as a baseline. We can see our approach using HOG features achieves higher accuracy than LDA. Moreover, we also report the classification performance of different features using our method. The performance using concatenated features is better than using one feature alone.

In Figure 2, we show the incorrectly classified traffic signs of our approach together with the committee's CNN. As can be seen many errors emerging in the committee's CNN do not appear in our approach. However, both the committee's CNN and our approach incorrectly classify many speed limit signs, which seem to be easy to recognize. This is because speed limit signs may be similar within classes and bad illumination or serious blur may occur. Both the committee's CNN and our approach confuse some red light warning signs with some danger warming signs since these two classes have high similarity. Most of the remaining errors are either due to bad illumination, serious blur or dirty traffic signs.

[18] propose a traffic sign recognition algorithm called sparse-representation-based graph embedding (SRGE). The accuracy of SRGE on the BelgiumTSC is 96.29% better than the other dimensionality reduction methods mentioned in [18]. We also test our approach on the BelgiumTSC using the same parameter as on the GTSRB. As a result, we obtain a classification rate of 98.77%, outperforming all the methods mentioned above.

4. CONCLUSION

We presented a two-layer image representation based method to classify traffic signs. To the best of our knowledge, this is the first paper introduces feature coding and spatial pooling

(a) The incorrectly classified signs of the committee's CNN

(b) The incorrectly classified signs of our approach

Fig. 2. The wrongly classified traffic signs of the committee's CNN together with our approach.

for the problem of traffic sign recognition problem. We evaluates our method on two public standard databases, and experimental result shows that classification rate is high. Comparing to other methods in the GTSRB competition, our method achieves the state-of-the-art performance on the GTSRB. The results indicated that the proposed two-layer image representation based method is an efficient way to recognize traffic signs and can be employed to real-time traffic sign recognition in videos. The state-of-the-art performance obtained by our method confirms that classifying traffic signs using this two-layer image representation is a very promising direction. Our future works focus on improving the proposed method by adding edge and color features.

Acknowledgement

This work was supported by the National Natural Science Foundation of China (NSFC) Grants 60873127, 60903096 and 61173120.

5. REFERENCES

[1] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, "The german traffic sign recognition benchmark: A multi-class classification competition," in *IEEE International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2011, pp. 1453–1460.

[2] I.M. Creusen, R.G.J. Wijnhoven, E. Herbschleb, and P.H.N. de With, "Color exploitation in hog-based traffic sign detection," in *IEEE International Conference on Image Processing (ICIP)*. IEEE, 2010, pp. 2669–2672.

[3] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, "Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition," *Neural Networks*, vol. 32, no. 0, pp. 323–332, 2012.

[4] J.F. Khan, S.M.A. Bhuiyan, and R.R. Adhami, "Distortion invariant road sign detection," in *IEEE International Conference on Image Processing (ICIP)*. IEEE, 2009, pp. 841–844.

[5] D.G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision (IJCV)*, vol. 60, no. 2, pp. 91–110, 2004.

[6] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, 2005, vol. 1, pp. 886–893.

[7] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern recognition*, vol. 29, no. 1, pp. 51–59, 1996.

[8] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-constrained linear coding for image classification," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2010, pp. 3360–3367.

[9] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2006, vol. 2, pp. 2169–2178.

[10] X. Wang, X. Bai, W. Liu, and L.J. Latecki, "Feature context for image classification and object detection," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 961–968.

[11] C.C. Chang and C.J. Lin, "Libsvm: a library for support vector machines," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2, no. 3, pp. 27, 2011.

[12] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2009, pp. 1794–1801.

[13] D. Ciresan, U. Meier, J. Masci, and J. Schmidhuber, "A committee of neural networks for traffic sign classification," in *International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2011, pp. 1918–1921.

[14] A. Møgelmose, M.M. Trivedi, and T.B. Moeslund, "Vision-based traffic sign detection and analysis for intelligent driver assistance systems: Perspectives and survey," *IEEE Transactions on Intelligent Transportation Systems (ITS)*, vol. 13, pp. 1484–1497, 2012.

[15] R. Timofte, K. Zimmermann, and L. Van Gool, "Multi-view traffic sign detection, recognition, and 3d localisation," in *Applications of Computer Vision (WACV), 2009 Workshop on*. IEEE, 2009, pp. 1–8.

[16] P. Sermanet and Y. LeCun, "Traffic sign recognition with multi-scale convolutional networks," in *International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2011, pp. 2809–2813.

[17] F. Zaklouta, B. Stanciulescu, and O. Hamdoun, "Traffic sign classification using kd trees and random forests," in *International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2011, pp. 2151–2155.

[18] Ke Lu, Zhengming Ding, and Sam Ge, "Sparse-representation-based graph embedding for traffic sign recognition," *IEEE Transactions on Intelligent Transportation Systems (ITS)*, vol. 13, pp. 1515–1524, 2012.