2016 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)

Traffic Sign Recognition with Convolutional Neural Network Based on Max Pooling Positions

Rongqiang Qian*, Yong Yue*, Frans Coenen† and Bailing Zhang*
*Department of Computer Science and Software Engineering
Xi'an Jiaotong-Liverpool University
Suzhou, P.R.China, 215123
†Department of Computer Science
University of Liverpool
Liverpool, L69 3BX, United Kingdom

***Abstract*—Recognition of traffic signs is vary important in many applications such as in self-driving car/driverless car, traffic mapping and traffic surveillance. Recently, deep learning models demonstrated prominent representation capacity, and achieved outstanding performance in traffic sign recognition. In this paper, we propose a traffic sign recognition system by applying convolutional neural network (CNN). In comparison with previous methods which usually use CNN as feature extractor and multilayer perception (MLP) as classifier, we proposed max pooling positions (MPPs) as an effective discriminative feature to predict category labels. Through extensive experiments, MPPs demonstrates the ideal characteristics of small inter-class variance and large intra-class variance. Moreover, with the German Traffic Sign Recognition Benchmark (GTSRB), outstanding performance has been achieved by using MPPs.**

***Keywords:*—Advanced Driver Assistance, traffic sign recognition, deep learning, convolutional neural networks, max pooling**

I. Introduction

The acquisition of information from real-world traffic system is a key component in many applications, such as self-driving car/driverless car, traffic mapping and traffic surveillance. With the advent of some publicly available benchmark datasets such as the German Traffic Sign Recognition Benchmark (GTSRB) [1], a number of outstanding results for recognition of European traffic signs have been reported in the literature [2], [3], [4].

Recently, the development of deep learning has attracted much attention in computer vision research as more and more promising results are published on a range of different vision tasks. Among the deep learning models, the convolutional neural networks (CNN) have acquired unique noteworthiness for their repeatedly confirmed superiorities. Convolutional neural networks have powerful representational learning capabilities, with a number of desirable properties such as the translation invariance and spatially local connections. A pre-trained CNN model can be efficiently exploited as a generic feature extractor for different vision tasks.

Despite the excellent performance achieved by CNN, exploring, understanding and interpreting the internal working principle of CNN remains the most elusive problems to researchers. Some recent works visualize CNN models and perform recognition tasks by activation of CNN [5], [6], [7], [8]. Inspired by these works, we propose a novel way for the recognition of the traffic sign recognition with CNN based on MPPs.

The main motivation of this paper is to present a novel scheme for traffic sign recognition. The main characteristics of our proposed system include:

- A CNN model to learn a compact yet discriminative feature representation.
- A novel method to perform classification based on MPPs.
- A novel method to improve classification performance and speed using MPPs.

The rest of this paper is organized as follows: Section 2 outlines some related research on traffic sign recognition and CNN applications; Section 3 provides a detailed description of the proposed system; Implementation details and experimental results will be provided in Section 4, followed by conclusion in Section 5.

II. Related Works

A. Traffic Sign Recognition

TSR has been an active area of research in computer vision community for many years. With many mature off-the-shelf techniques from machine learning, TSR can be generally treated as a pattern classification issue. Among the plenteous models, Support Vector Machine (SVM) demonstrates its excellent performance, which has been applied in [9], [10]. Boosting is another powerful method for traffic sign classification. A robust sign similarity measurement with SimBoost and fuzzy regression tree method was proposed in [11]. An ensemble of classifiers based on the Error-Correcting Output Code (ECOC) framework was introduced in [12], where the ECOC was designed through a forest of optimal tree structures that are embedded in the ECOC matrix.

As for any visual object classification, feature expression is the critical factor that affects system performance. How to design discriminative and representative features has been in the central stage of computer vision research. Due to the powerful representational learning capabilities of CNN, in recent works on traffic sign recognition, the dominant

978-1-5090-4093-3/16/$31.00 ©2016 IEEE

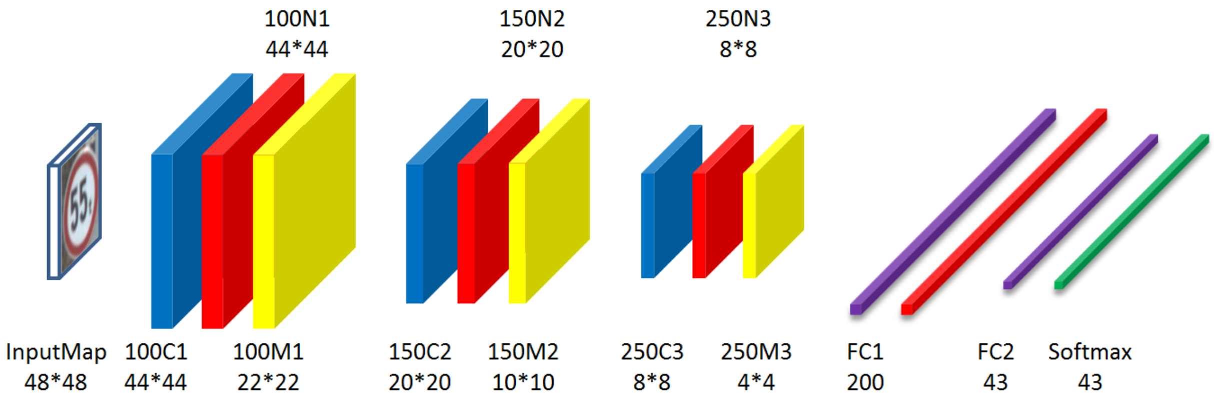

Fig. 1. Network architecture in training stage

approaches are generally based on CNN models. For example, committee CNN [3], multi scale CNN [4], multi column CNN [13] and hinge-loss CNN [14]. And preeminent performance has been achieved in GTSRB [1].

B. Understanding of Convolutional Neural Network

A CNN is a special type of multi-layer neural network that extracts features by combining convolution, pooling and activation layers. The most successful CNN architecture [15] is trained with back-propagation, and leading performance has been achieved on many benchmark datasets. Although CNN models have been proven to have powerful description ability, it remains unclear how features are learned inside the network. The lack of understanding of CNN has attracted some researches to seek deeper insight into its working principle. For example, part detector discovery (PDD) was proposed based on the analysis of the gradient maps of the network outputs and finding activation centers in [5]. An unsupervised fine-grained recognition scheme [6] was introduced and part models were generated by finding constellations of neural activation patterns. In [7], the visualisation of image classification models are displayed based on computing the gradient of the class score with respect to the input image. Motivated by visualizing and understanding CNNs, a recent work also proposed a novel scheme to visualize activations based on a multi-layered Deconvolutional Network (deconvnet) in [8].

Inspired by previous researches that aims to visualizing and activating CNNs for performing particular tasks, we focus on exploring the relationships between max pooling positions and particular recognition tasks. To be more specific, instead of directly using features for training classifies, we use max pooling to sample and encode the features in the first, and then the encoded pooling sequence will be further used to predict detail results.

III. Approach

An overview of the proposed recognition system is demonstrated in Fig. 2. In the first stage of our system, an input image will be normalized by using contrast-limited adaptive histogram equalization (CLAHE) [16]. And then, the normalized image will be passed to a CNN model to extract discriminative features. Finally, MPPS is adopted to predict the detail labels of the input image.

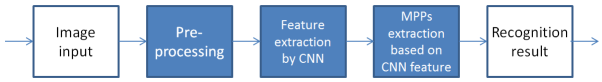
Image input
Pre-processing
Feature extraction by CNN
MPPs extraction based on CNN feature
Recognition result

Fig. 2. System overview

A. Network Architecture in Training Stage

The network applied in training stage is similar to [3], which is illustrated in Fig. 1. The network consists of three convolution stages followed by fully connection layers and softmax layer. Each convolution stage includes convolutional layer, non-linear activation layer and max pooling layer. ReLU [15] is employed as the activation function for convolutional layers and full connection layers. Local response normalization (LRN) is used for normalizing feature maps. Dropout [15] is also adopted for preventing over-fitting. The final softmax layer has 43 outputs, corresponding to each category in GTSRB [1].

The structure of the networks and the hyper-parameters were empirically initialized based on previous works using ConvNets [3]. Then we setup cross-validation experiment to optimize the parameters of network architecture, with details shown in Table I.

B. Network Architecture in Testing Stage

The network applied in testing stage is illustrated in Fig. 3. It is easily noticed that the original full connection layers and softmax layer are replaced with 903 new full connection layers. Each of the full connection layers can be regarded as a one-versus-all classifier. Instead of training all the weights by the standard back-propagation algorithm, all the parameters can be simply selected by using our MPPs method. The details will be further elaborated in the next section.

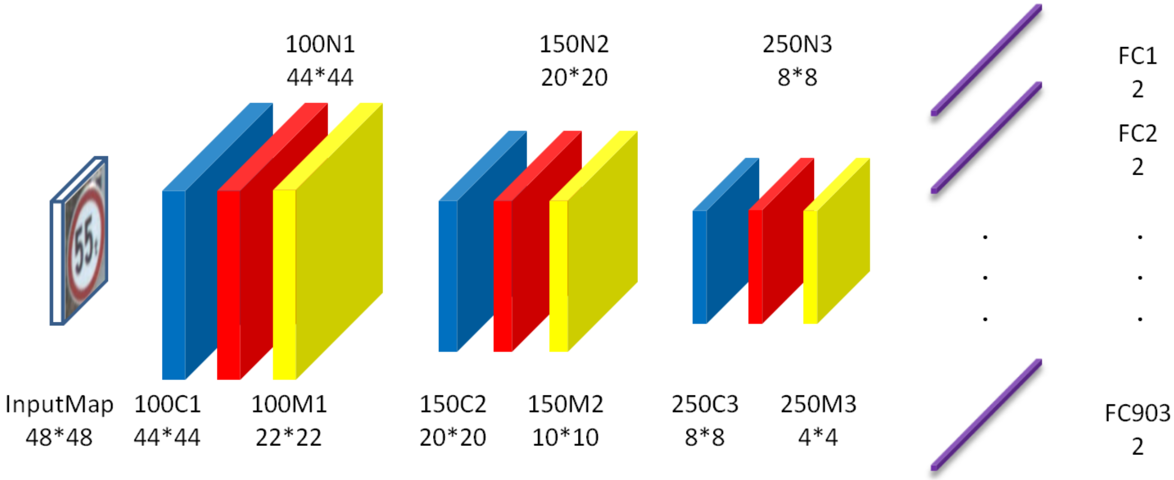

Fig. 3. Network architecture in testing stage

TABLE I
SELECTION OF CNN PARAMETERS

Layer	Type	Feature maps & Size	Kernel
1	Input	1 × 48 × 48	
2	Convolution C_1	100 × 44 × 44	5 × 5
3	ReLU	100 × 44 × 44	
4	LRN	100 × 44 × 44	
5	Max pooling M_1	100 × 22 × 22	2 × 2
6	Convolution C_2	150 × 20 × 20	3 × 3
7	ReLU	150 × 20 × 20	
8	LRN	150 × 20 × 20	
9	Max pooling M_2	150 × 10 × 10	2 × 2
10	Convolution C_3	250 × 8 × 8	3 × 3
11	ReLU	250 × 8 × 8	
12	Max pooling M_3	250 × 4 × 4	2 × 2
13	Fully connection FC_1	200	
14	ReLU	200	
15	Dropout	200	
16	Fully connection FC_2	43	
17	Softmax	43	

C. Classification by Max Pooling Positions

Since the final layer of the proposed CNN has 250 feature maps with 4 × 4 neurons, the extracted features have a dimension of 4000. As Fig. 4 demonstrates, for each 4 × 4 feature map, max pooling with kernel size 2 × 2 and stride 2 is performed, with the position of each max value being recorded. And then, each recorded position is encoded into a four bits binary value. Finally, the whole MPPs sequence can be obtained by concatenating all the binary values. The dimension of MPPs sequence is also 4000.

For each classifier, the training stage can be described in the following steps.

Step 1. Data collection. Two corresponding classes are acquired based on their labels.

Step 2. Data processing. In order to measure the similarities of MPPs belonging to same class, all of the MPPs sequences come from same class are accumulated together and normalized by dividing the number of samples. So that we get a series of sequences that indicate the probability of appearance for each of the max value positions. Therefore, two probability sequences are achieved, namely, p_1 and p_2.

Step 3. Activation selection. Based on the values of the two probability sequences p_1 and p_2, the highest n $(n = 5, 10..., 4000)$ channels will be activated. To be more specific, features of each class will be selected and reduced to n dimension according to the two probability sequences.

Step 4. Classifier initialization. The purpose of this step is to select a decision matrix for each classifier with a dimension of 2 × 4000. The corresponding decision matrix $d(x, y)$ is initialized by the probability value of each selected channel in p_1 and p_2, the value of rest channels are set to be 0.

Step 5. Classifier fine-tuning. The obtained decision matrix $d(x, y)$ needs further tuning. In each iteration, if the current classifier predicts uncorrect label, the corresponding decision matrix is fine-tuned by adding the product of current activated MPPs sequence and a learning rate.

IV. EXPERIMENT

In this section, we will first introduce the implementation details of our CNN model, including the architecture selection and training. And then, the proposed MPPs method will be evaluated. Finally, performance comparison will be provided for the traffic sign recognition. The details will be introduced with the corresponding experiments in the following.

A. Implementation Details

To implement our TSD system, a computer with Xeon 2.93GHz CPU, 24GB memory and 6GB memory TITAN GPU is employed. The program runs on a 64-bit Windows system with CUDA 7.0, Matlab 2015a and MatConvNet [17] installed.

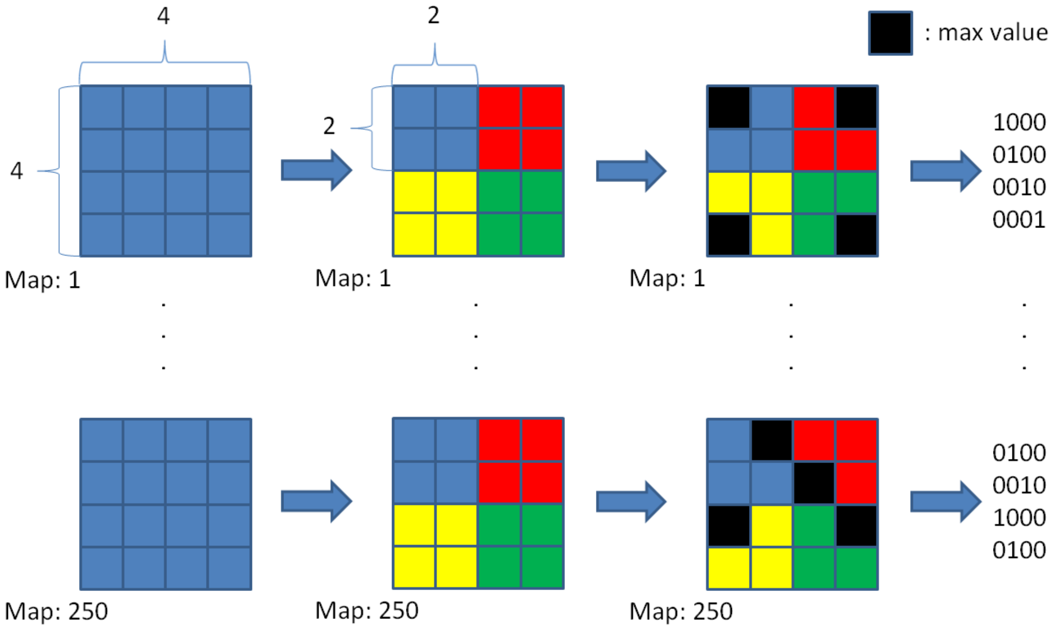

Fig. 4. Classification by max pooling positions

1) Structure Selection: The representational features extracted by CNN mainly depend on its structure. The CNN structure should have enough layers and kernels for extracting discriminate features. However, to prevent over-fitting, the capacity of CNN should be limited to a rational range. Inspired by work in [3], we built a similar CNN model. Instead of using hyperbolic tangent as the activation function, ReLU [15] is employed in our CNN. One of the main advantages of ReLU is that it does not saturate at the upper end thus avoiding the gradient vanishing problem as with sigmoid and tanh functions in the classical neural networks.

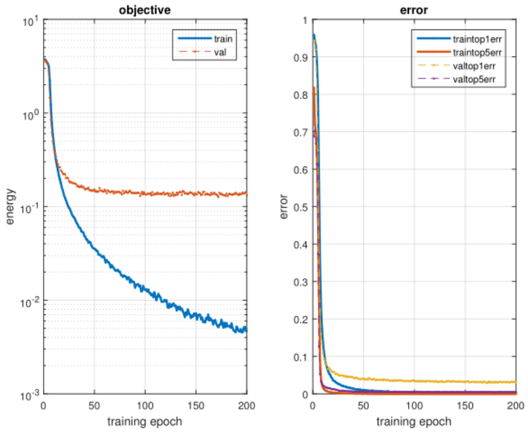

Fig. 5. Illustration of training loss and error vs. epoch.

2) Training by GTSRB dataset: The train was conducted based on GTSRB [1], as illustrated in Fig. 5. Data augmentation is a technique used to enlarge the training data set and improve the generalization of CNN. In order to demonstrate the advantages of MPPs, no data augmentation is used here.

The detail training scheme is processed as follows: (i) initial weights of convolution layers and full connection layers are achieved from a uniform random distribution in the range [-0.01, 0.01] ; (ii) the learning rate is set to be 0.001; (iii) training is conducted by using cross-entropy loss and mini batch gradient descent for 200 epoches.

B. Max Pooling Positions as a Category-level Attribute

In order to measure the similarities of MPPs belonging to same category, all of the MPPs sequences come from same category are accumulated together and normalized by dividing the number of samples. So that we get a series of sequences that indicate the probability of appearance for each max value positions. The data for the first five categories in GTSRB [1] are shown in Fig. 6. It clearly demonstrates that about 100 positions always pooling at the same place. The results are indeed encouraging as the samples from a category trend to be pooled at same place.

C. Max Pooling Positions as an Effective Discriminative Feature

In this section, the discrimination of MPPs from different categories are discussed. We also select the the first five categories in GTSRB [1] for the investigation. Firstly, the MPPs of all the samples are compared. And then, for every pair of the categories, an average distance value is calculated, as shown in Table II. It is obvious that MPPs are significantly variant for samples come from different categories.

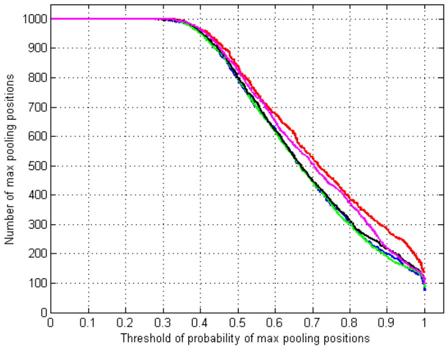

Fig. 6. Similarities of max pooling positions for same category

TABLE II
DISCRIMINATION OF MAX POOLING POSITIONS FOR DIFFERENT CATEGORIES

	1	2	3	4	5
1	738	853	896	888	858
2	853	840	883	898	887
3	896	883	847	894	895
4	888	898	894	823	914
5	858	887	895	914	783

D. Classification by Max Pooling Positions

Based on the above discussion that MPPs are similar for same category samples and discriminative for different categories samples, we further investigate the recognition potential of MPPs. Using the method introduced in Section 3, outstanding performance has been achieved, as explained by Table III.

Since no data augmentation is used in fine-tuning stage, the recognition rate is about 96.95% based on our CNN - MLP scheme. However, the recognition rate can be boosted to 98.86% by the proposed MPPs method, which means the representation capabilities of MPPs is very promising.

TABLE III
RECOGNITION RATE OF DIFFERENT METHODS

Team	Method	Accuracy
Jin [14]	HLSGD-CNNs	99.65%
IDSIA [13]	Committee of CNNs	99.46%
Ours	MPPs-CNNs	98.86%
INI-RTCV [1]	Human Performance	98.84%
Sermanet [4]	Multi-Scale CNNs	98.31%
Ours	MatConvNet-CNNs	96.95%
CAOR [18]	Random Forests	96.14%

V. CONCLUSION

In this paper, a novel traffic sign recognition system is proposed, with main contributions including: (i) a CNN model to learn a compact yet discriminative feature representation; (ii) a novel method to perform recognition based on MPPs; (iii) a novel method to improve classification performance and speed using MPPs. By introducing MPPs for recognition, accuracy rate is significantly improved. Extensive experiments have been performed, yielding promising results.

REFERENCES

[1] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, "Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition," *Neural Networks*, vol. 32, pp. 323 – 332, 2012, selected Papers from {IJCNN} 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0893608012000457

[2] M. Mathias, R. Timofte, R. Benenson, and L. Van Gool, "Traffic sign recognition - how far are we from the solution?" in *Neural Networks (IJCNN), The 2013 International Joint Conference on*, Aug 2013, pp. 1–8.

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

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

[5] M. Simon, , E. Rodner, , and J. Denzler, *Computer Vision – ACCV 2014: 12th Asian Conference on Computer Vision, Singapore, Singapore, November 1-5, 2014, Revised Selected Papers, Part II*. Cham: Springer International Publishing, 2015, ch. Part Detector Discovery in Deep Convolutional Neural Networks, pp. 162–177.

[6] M. Simon and E. Rodner, "Neural activation constellations: Unsupervised part model discovery with convolutional networks," in *International Conference on Computer Vision (ICCV)*, 2015.

[7] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," *CoRR*, vol. abs/1312.6034, 2013. [Online]. Available: http://arxiv.org/abs/1312.6034

[8] M. D. Zeiler and R. Fergus, *Computer Vision – ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I*. Cham: Springer International Publishing, 2014, ch. Visualizing and Understanding Convolutional Networks, pp. 818–833.

[9] S. Maldonado-Bascon, S. Lafuente-Arroyo, P. Gil-Jimenez, H. Gomez-Moreno, and F. Lopez-Ferreras, "Road-sign detection and recognition based on support vector machines," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 8, no. 2, pp. 264–278, June 2007.

[10] M. Shi, H. Wu, and H. Fleyeh, "Support vector machines for traffic signs recognition," in *Neural Networks, 2008. IJCNN 2008. (IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on*, June 2008, pp. 3820–3827.

[11] A. Ruta, Y. Li, and X. Liu, "Robust class similarity measure for traffic sign recognition," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 11, no. 4, pp. 846–855, Dec 2010.

[12] X. Baro, S. Escalera, J. Vitria, O. Pujol, and P. Radeva, "Traffic sign recognition using evolutionary adaboost detection and forest-ecoc classification," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 10, no. 1, pp. 113–126, March 2009.

[13] D. Ciresan, U. Meier, J. Masci, and J. Schmidhuber, "Multi-column deep neural network for traffic sign classification," *Neural Networks*, vol. 32, pp. 333 – 338, 2012, selected Papers from {IJCNN} 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0893608012000524

[14] J. Jin, K. Fu, and C. Zhang, "Traffic sign recognition with hinge loss trained convolutional neural networks," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 15, no. 5, pp. 1991–2000, Oct 2014.

[15] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. Burges, L. Bottou, and K. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105.

[16] K. Zuiderveld, "Contrast limited adaptive histogram equalization," in *Graphics gems IV. Academic Press Professional*, 1994, pp. pp. 474–485.

[17] A. Vedaldi and K. Lenc, "Matconvnet: Convolutional neural networks for matlab," in *Proceedings of the 23rd ACM International Conference on Multimedia*, ser. MM '15. New York, NY, USA: ACM, 2015, pp. 689–692. [Online]. Available: http://doi.acm.org/10.1145/2733373.2807412

[18] F. Zaklouta, B. Stanciulescu, and O. Hamdoun, "Traffic sign classification using k-d trees and random forests," in *Neural Networks (IJCNN), The 2011 International Joint Conference on*, July 2011, pp. 2151–2155.