Lightweight Spatial Pyramid Convolutional Neural Network Classifier for Traffic Sign Classification

Reza Fuad Rachmadi*†, Gou Koutaki* and Kohichi Ogata*
*Graduate School of Science and Technology (GSST), Kumamoto University, Kumamoto, Japan
†Department of Computer Engineering, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia
fuad@navi.cs.kumamoto-u.ac.jp, {koutaki,ogata}@cs.kumamoto-u.ac.jp

***Abstract*—In this paper, we proposed a lightweight spatial pyramid convolutional neural network (SP-CNN) classifier for image-based traffic sign classification. The lightweight SP-CNN classifier is formed based on ResNet (residual network) CNN architecture which originally used for CIFAR10 image classification problems. Our proposed classifier consists of five parallel convolutional networks and each network processes a cropped region using spatial pyramid configuration. For smoother transitions between the regions cropped in the level 1 of spatial pyramid configuration, we overlap the level 1 of spatial pyramid regions configuration for around 12.5% on each axis. The proposed classifier trained by fine-tuning the CIFAR10 weights with NAG (Nesterov Accelerated Gradient) training algorithm. Experiments on GTSRB (German Traffic Sign Recognition Benchmark) dataset show that our lightweight SP-CNN version produces an accuracy of 99.70% and an execution time of 60 ms. The proposed classifier produces a very competitive accuracy compared with other methods but with less number of parameters.**

***Index Terms*—spatial pyramid features, convolutional neural network, traffic sign classification,**

I. Introduction

ITS (Intelligent Transportation System) is one of the active research topic for several different fields, including computer science, transportation engineering, and mechanical engineering. One of the active research in ITS is image-based traffic sign classification problem which intersects with computer vision research in the computer science field. Image-based traffic sign classification is a challenging problem due to the optical sensor used in the data acquisition process. There are several different traffic sign classification datasets that are available for research purposes and can be used to evaluate the traffic sign classification method, including GTSRB (German Traffic Sign Recognition Benchmark) dataset [1], BTS (Belgium Traffic Sign) dataset [2], Malaysian traffic sign dataset [3], Japan road sign dataset [4], CURE-TSR dataset [5], and traffic-sign in the wild [6].

Utilizing state-of-the-art convolutional neural network (CNN) classifiers can be a possible solution for traffic sign classification, indeed such approaches are already described in [4], [7]–[13]. The CNN classifier becomes a very good solution for a lot of general image-based classification problems after Krizhevsky et al. [14] won the ILSVRC (ImageNet Large Scale Visual Recognition Challenges) 2012 with large margin compared with traditional bag-of-features approaches. One of the disadvantages of CNN classifier is that the classifier consists of very huge parameters which can be very difficult to implement in the real world application. Some of the state-of-the-art CNN classifier for traffic sign classification has a varied amount of parameters from 0.5 million to 38.5 million. Unfortunately, the CNN approaches with the lowest number of parameters achieved an accuracy around the same accuracy as the human performance.

In this paper, we proposed a lightweight SP-CNN (Spatial Pyramid Convolutional Neural Network) classifier. On GTSRB dataset, our proposed classifier achieved around the same accuracy as the state-of-the-art methods but with a lower number of parameters. Our contributions can be described as follows

- We investigated a lightweight spatial pyramid convolutional neural network (SP-CNN) classifier by designing the classifier using lightweight state-of-the-art CNN architectures. Experiments on GTSRB dataset show that our proposed classifier produces a very good accuracy and comparable with other state-of-the-art methods.
- We investigated the accuracy of the proposed classifier when fine-tuning from the CIFAR10 dataset (a small scale dataset) instead of the ImageNet dataset. The experiments show that the fine-tuning from CIFAR10 weights can also provide a very good feature extraction network.

The rest of the paper organized as follows. Section 2 describes the detail of our proposed lightweight SP-CNN classifier. Setup of the experiments on GTSRB dataset is described in section 3. The results and discussion of the experiments are discussed in section 4. Finally, we concluded the experiments in section 5.

II. Lightweight SP-CNN Classifier

This section describes our proposed lightweight SP-CNN classifier based on ResNet CNN architecture. We already use the SP-CNN classifier for several problems, including social event detection in static images [15] and Japan road sign classification problems [16].

A. Spatial Pyramid CNN

Spatial pyramid CNN classifier is formed using five parallel convolutional networks taken from state-of-the-art CNN architecture. The parallel convolutional network aims to extract spatial pyramid features from the input images. Each

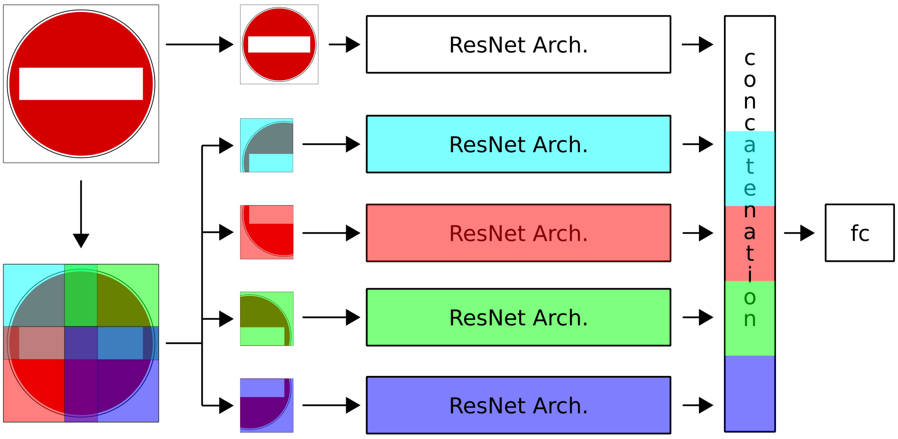

Fig. 1. Diagram of SP-ResNet classifier for traffic sign classification experiments. The level 1 of spatial pyramid region is overlapping each other by approximately around 12.5% on each axis.

convolutional network is given an input of the region of the input image cropped based on spatial pyramid configuration. At the end of the parallel convolutional network, the features extracted from each convolutional network are concatenated and processed further using fully-connected layer. The original SP-CNN classifier is described in our previous paper [15], [16] which is designed based on AlexNet CNN architecture [14]. One of the disadvantages of our previous SP-CNN classifier is that the number of parameters is very high and impractical for real-time world application.

B. Lightweight SP-CNN

To create a lightweight version of SP-CNN classifier, we redesign the previous SP-CNN classifier based on ResNet CNN architecture that originally used for CIFAR10 image classification problems. We use ResNet20 and ResNet32 CNN architecture as the main architecture for the parallel convolutional network because the number of parameters in the classifier is quite low compared with other approaches on CIFAR10 dataset. The regions in the level 1 spatial pyramid configuration are overlapped 12.5% on each axis to create a smooth transition between the regions. The input resolution of the image is also changed to the resolution of the CIFAR10 dataset used, 32×32. Unlike AlexNet based SP-CNN classifier which has three fully-connected layers, the ResNet based SP-CNN classifier has only one fully-connected layer follows the original ResNet architecture. Figure 1 shows the diagram of the ResNet based SP-CNN classifier. The total number of parameters for ResNet20 based SP-CNN classifier is 1.41 million parameters, while the number of parameters for ResNet32 based SP-CNN classifier is 2.39 million parameters. For comparison, we also perform experiments on GTSRB dataset using AlexNet based SP-CNN classifier with the same training parameters and spatial pyramid configuration as the ResNet based SP-CNN classifier.

III. Experiments Setup

In this section, we describes the setup of the SP-CNN experiments using GTSRB traffic sign classification dataset, including a brief explanation of GTSRB dataset, data augmentation, training process, and testing process.

A. GTSRB Dataset

GTSRB (German Traffic Sign Recognition Benchmark) dataset [1] is a dataset that usually used to evaluate the traffic sign recognition system. The GTSRB dataset is divided into two different evaluation tasks, traffic sign classification and traffic sign recognition. The traffic sign classification task aims to recognize traffic sign image with an assumption that each input image has only one traffic sign category, while the traffic sign recognition task aims to recognize multiple traffic sign in the input image. Examples of individual traffic sign images in GTSRB dataset can be viewed in Figure 2. The GTSRB dataset is very challenging because the traffic sign images contain a lot of noise, large resolution variety, and unbalance; due to the optical sensor used in the data acquisition. The number of examples for each traffic sign category varies from 30-600 image. The GTSRB dataset consists of 51,839 images and 43 traffic sign category with roughly 75%/25% training/testing split configuration. The traffic sign images were taken using a digital camera in the road around Germany.

B. Data Augmentation

We use heavy data augmentation methods for the training process. The heavy data augmentation performed by enriching the GTSRB training dataset with three different methods, including random 2D rotation transformation, CLAHE [17], and histogram equalization. The final GTSRB dataset used for the training process consists of 645,000 images with 15,000 examples for each traffic sign category. The images in the training dataset were resized into a resolution of 256×256 for the SP-AlexNet based SP-CNN classifier and 36×36 for

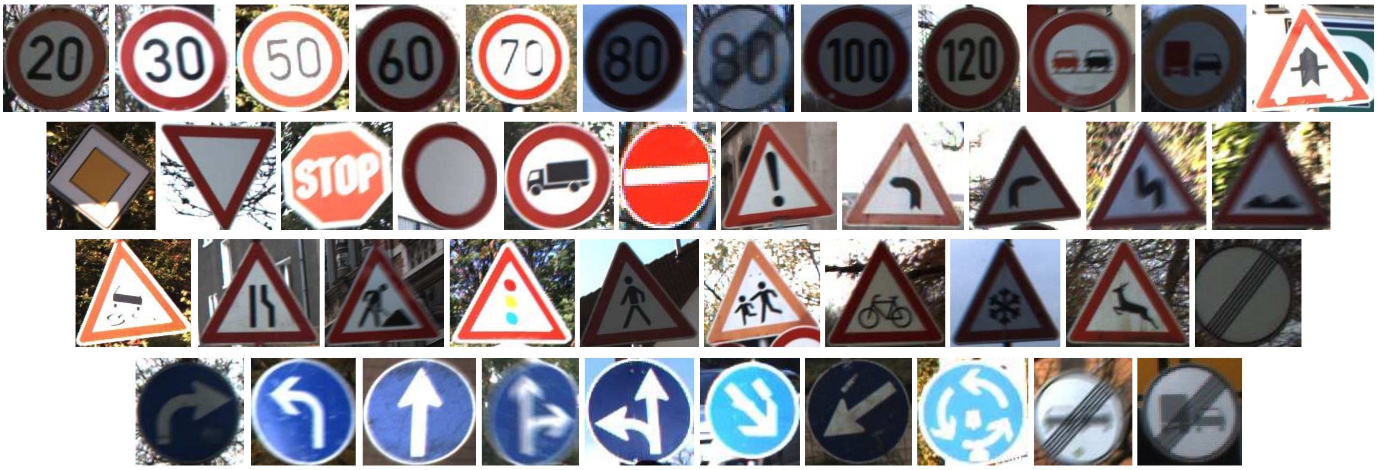

Fig. 2. Examples of traffic sign images in the GTSRB traffic sign classification dataset. Each category represented by one image with a total of 43 traffic sign category.

TABLE I
COMPARISON OF SEVERAL DIFFERENT SP-CNN CLASSIFIERS AND CLASSIFIER EXECUTION TIME ON GTSRB DATASET. THE EXECUTION TIME MEASURED USING NVIDIA GTX 960 HARDWARE.

No.	**Method**	**#Param**	**Accuracy**		**Time Execution**	
			Center Crop	**Five Crop**	**Center Crop**	**Five Crop**
1.	SP-AlexNet	83.1 M	99.39%	99.38%	14.31 ms	47.85 ms
2.	SP-ResNet20	1.41 M	99.37%	99.39%	22.47 ms	37.76 ms
3.	SP-ResNet32	2.39 M	99.65%	99.70%	35.69 ms	60.04 ms
4.	Ensemble of (2) & (3)	3.80 M	99.73%	**99.75%**	58.16 ms	97.80 ms

the SP-ResNet based SP-CNN classifier. We also performed on-fly data augmentation with three different processes, image blurring, image contrast variation (random from -10% to 10%), and random 2D rotation transformation. The training data were subtracted with 128 to create zero mean version of the data.

C. Training Process

The training process is done using Caffe deep learning framework [18] with an initialized learning rate of 0.01 and reduced using polynomial policy along the training process iterations. The proposed classifier trained for 10 epochs or around 24,000 iterations using a mini batch of 256. The Stochastic Gradient Descent (SGD) with additional NAG (Nesterov Accelerate Gradient) method [19] is used as a training algorithm with momentum parameter of 0.9 and weight decay of 0.0005. All weights of the five parallel convolutional networks are initialized using CIFAR10 weights for SP-ResNet20 and SP-ResNet32. The final fully-connected layer weights are initialized using initialization weights method described in [20].

D. Testing Process

Same as in the training process, the input image is resized to 256×256 for AlexNet based SP-CNN and 36×36 for ResNet based SP-CNN. The testing data also subtracted with 128 to create the same data distribution as in the training data. Two different approaches are used in the testing process, classify the input image using only center crop and classify the input image using five crops (left bottom, right bottom, left top, right top, and center crop). The mirror version of the image is not used because there are pairs of categories that opposite mirror of each other.

IV. RESULTS AND DISCUSSION

This section discusses the results of the experiments on GTSRB dataset. The results of the experiments reported using global accuracy and per traffic sign category which can be compared with other methods for comparison.

A. Testing Results

Table I shows the summary of the experiments using SP-CNN classifier on GTSRB dataset. We also reported the time execution of the classifier with NVIDIA GTX 960 hardware. As shown in Table I, our proposed lightweight SP-CNN classifier has a fewer number of parameters compared with the AlexNet based SP-CNN classifier. The best accuracy for the single classifier is achieved using SP-ResNet32 classifier with an accuracy of 99.70%, an execution time of 60.04 ms, and 2.39 million parameters. Figure 3 shows the misclassified

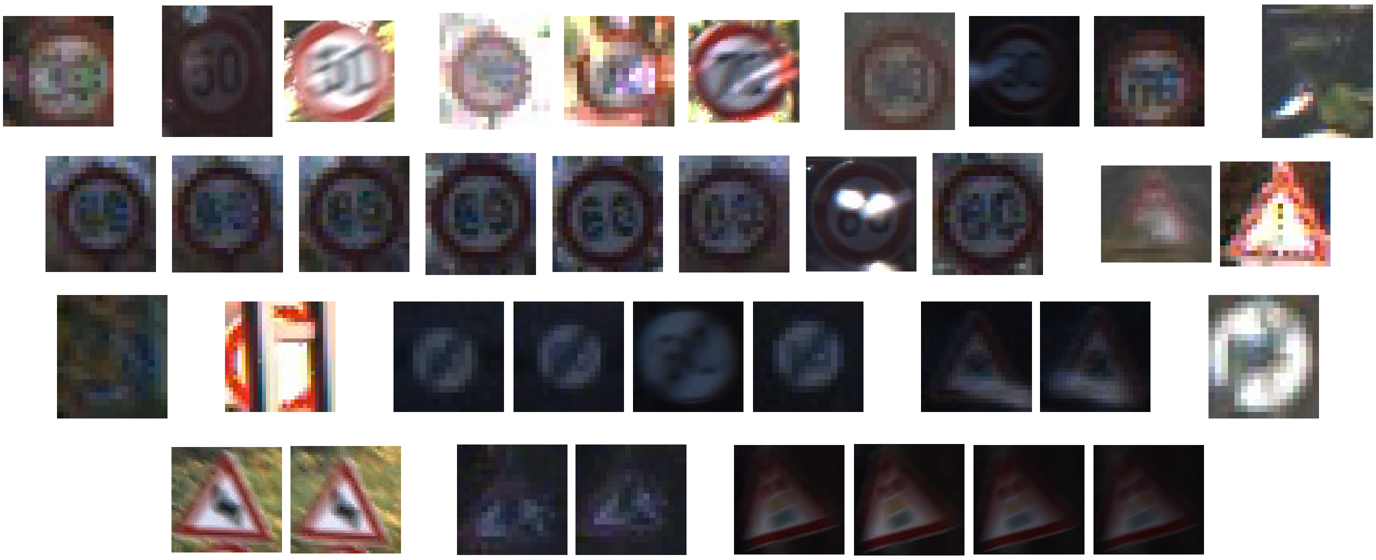

Fig. 3. The testing data that wrongly classified by the SP-ResNet32 classifier. The total number of wrongly classified data is 37 out of 12,630 data.

TABLE II
COMPARISON OF OUR PROPOSED SP-CNN CLASSIFIER WITH SEVERAL OTHER APPROACHES ON GTSRB DATASET.

Method	#Param	Accuracy
Human Performance [21]	-	98.84%
EPCNN [12]	5.22 M	99.70%
STDCNN [11]	14.6 M	99.71%
EHLDCNN [10]	23.2 M	99.65%
MCDCNN [9]	38.5 M	99.46%
μNet [8]	0.51 M	98.90%
Our approach (SP-ResNet32)	2.39 M	99.70%
Our approach (Ensemble)	**3.80 M**	**99.75%**

traffic sign images on the GTSRB testing dataset for the SP-ResNet32 classifier. As shown in Figure 3, the misclassified images either have a degradation image quality, low contrast, and/or partial visibility problem.

To improve the performance of the classifier, we also conducted the testing process using an ensemble of SP-ResNet20 and SP-ResNet32 classifier. The ensemble is done by averaging the predictions results between two classifiers. As a result, the ensemble classifier produces a slightly better accuracy compared with single SP-ResNet32 classifier with 99.75% accuracy. Although the accuracy is better, the execution time of the ensemble classifier is increased and close to around 10 images per second.

B. Comparison

Table II shows the comparison between the SP-CNN classifier with several other approaches on GTSRB dataset. Several other approaches are described as follows

- **Human Performance** describes the average accuracy of human performance on testing data of GTSRB dataset reported in [21].
- **EPCNN (Ensemble of Practical CNN)** method evaluated on GTSRB dataset and reported in [12]. The best accuracy achieved using four single PCNN (Practical CNN) classifier.
- **STDCNN (Spatial Transformers Deep CNN)** method evaluated on GTSRB dataset and reported in [11]. The best accuracy achieved using single CNN with 3 STNs (Spatial Transformers Network).
- **EHLDCNN (Ensemble of Hinge Loss Deep CNN)** method evaluated on GTSRB dataset and reported in [10]. The best accuracy is achieved using an ensemble of 20 single HLDCNN (Hinge Loss Deep CNN).
- **MCDCNN (Multi-Column Deep CNN)** method evaluated on GTSRB dataset and reported in [9]. The best accuracy achieved using an ensemble of 25 single MCD-CNN classifier.
- **μNet** aims to investigate small CNN classifier for embedded system implementation of traffic sign classification module and reported in [8].

As shown in Table II, our single SP-ResNet32 classifier achieved a comparable performance among other approaches. Our proposed SP-ResNet32 classifier has around 2.39 million number of parameters which is a middle-size classifier compared with other approaches. Although SP-ResNet32 has lower parameters compared with other approaches, the execution time of the classifier is not the lowest due to non-parallelable layer and element-wise operations in the residual network CNN architecture. The ensemble of SP-ResNet20 and SP-ResNet32 outperforms the best method described in [11] by around 0.05%. Although the accuracy is slightly similar, our approaches have a lower number of parameters compared with the classifier described in [11]. Table III shows the accuracy comparison for each subset in the GTSRB dataset. As shown in Table III, our proposed classifier achieved the highest accuracy in three subset, danger subset, red-other subset, and spezial

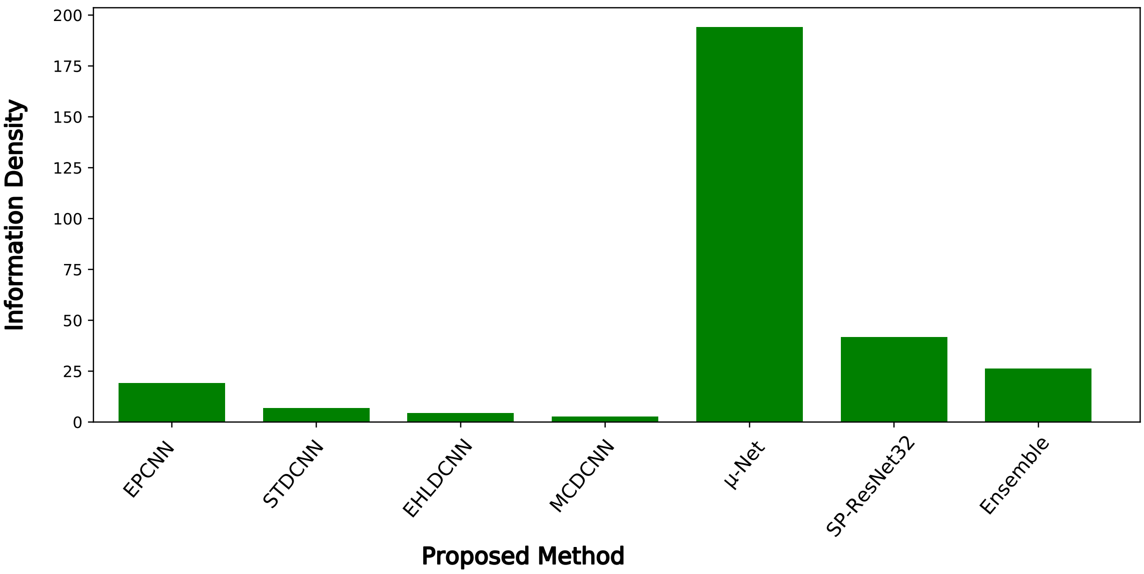

Fig. 4. Comparison of information density for several state-of-the-art CNN methods on GTSRB dataset, including our approaches (SP-ResNet32 and Ensemble). The information density shows as % per million params (higher is better).

TABLE III
COMPARISON OF OUR PROPOSED CLASSIFIER WITH SEVERAL OTHER APPROACHES IN SEVERAL DIFFERENT SUBSETS OF GTSRB DATASET.

Method	**Subset**							
	Blue	**Danger**	**End of**	**Red Round**	**Red Other**	**Speed**	**Spezial**	**All Signs**
Human Performance [21]	99.72%	98.67%	98.89%	98.00%	**99.93%**	97.63%	100.0%	98.84%
MCDCNN [9]	**99.89%**	99.07%	**99.72%**	99.74%	**99.93%**	99.47%	99.22%	99.46%
STDCNN [11]	99.77%	99.64%	98.89%	**99.86%**	99.87%	**99.69%**	99.80%	99.71%
Our approach (Ensemble)	99.83%	**99.75%**	99.44%	99.84%	**99.93%**	99.64%	**99.85%**	**99.75%**

subset.

C. Information Density

For further analysis, we compute the information density of the classifier which is also used in [8], [22]. The metric is described as the ratio between the performance of the classifier (in %) with the number of parameters in the classifier. In a formal way, the metric can be described using the following equation

$$D = \frac{p_c}{n_p} \tag{1}$$

with D is the information density of a deep neural network, p_c is the performance of the deep neural network (in %), and n_p is the number of parameters in the deep neural network classifier.

Figure 4 shows the information density for several state-of-the-art CNN methods on GTSRB dataset. The highest information density is achieved by μ-Net and followed by our proposed classifier. The μ-Net is designed for embedded system and has only around 0.5 million parameters. Although our proposed classifier has lower information density compared with μ-Net, our proposed classifier produces better accuracy compared with μ-Net classifier.

V. CONCLUSION

We have presented a lightweight SP-CNN (Spatial Pyramid Convolutional Neural Network) classifier for traffic sign classification. Our proposed classifier is formed using five parallel convolutional networks and a fully-connected layer. Each of the five parallel convolutional network is created using ResNet CNN architecture which originally used for CIFAR10 image classification problem. The ResNet20 and ResNet32 are chosen as the main architecture for the five parallel convolutional networks because the CNN architecture has less amount of parameters compared with other residual network architecture variant. For the smoother transition between regions in the level 1 of spatial pyramid configuration, the resolution of regions in the level 1 of spatial pyramid configuration is overlapped around 12.5% on each axis. Experiments on GTSRB dataset show that our proposed SP-ResNet32 classifier produces a comparable accuracy with other state-of-the-art methods but with a less number of parameters. Further experiments by ensembling the SP-ResNet20 and SP-ResNet32 classifier show that the ensemble classifier outperforms other state-of-the-art methods with less number of parameters.

An investigation of a trade-off between a number of parameters in the ResNet based SP-CNN classifier and the performance of the classifier needs to be conducted to provide a deeper analysis about SP-CNN classifier on GTSRB dataset. Experiments on other datasets, such as BTS (Belgium Traffic Sign) and CURE-TSR dataset, are required in order to generalize the performance of the SP-CNN classifier.

REFERENCES

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

[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*. IEEE, 2013, pp. 1–8.

[3] A. Madani and R. Yusof, "Malaysian traffic sign dataset for traffic sign detection and recognition systems," *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)*, vol. 8, no. 11, pp. 137–143, 2016.

[4] R. F. Rachmadi, Y. Komokata, K. Uchimura, and G. Koutaki, "Road sign classification system using cascade convolutional neural network," *International Journal of Innovative Computing, Information, and Control*, vol. 13, no. 1, pp. 95–109, 2017.

[5] D. Temel, G. Kwon, M. Prabhushankar, and G. AlRegib, "Cure-tsr: Challenging unreal and real environments for traffic sign recognition," *arXiv preprint arXiv:1712.02463*, 2017.

[6] Z. Zhu, D. Liang, S. Zhang, X. Huang, B. Li, and S. Hu, "Traffic-sign detection and classification in the wild," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2110–2118.

[7] Y. Zeng, X. Xu, Y. Fang, and K. Zhao, "Traffic sign recognition using extreme learning classifier with deep convolutional features," in *The 2015 international conference on intelligence science and big data engineering (IScIDE 2015), Suzhou, China*, vol. 9242, 2015, pp. 272–280.

[8] A. Wong, M. J. Shafiee, and M. S. Jules, "μnet: A highly compact deep convolutional neural network architecture for real-time embedded traffic sign classification," *arXiv preprint arXiv:1804.00497*, 2018.

[9] D. CireşAn, U. Meier, J. Masci, and J. Schmidhuber, "Multi-column deep neural network for traffic sign classification," *Neural networks*, vol. 32, pp. 333–338, 2012.

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

[11] lvaro Arcos-Garca, J. A. lvarez Garca, and L. M. Soria-Morillo, "Deep neural network for traffic sign recognition systems: An analysis of spatial transformers and stochastic optimisation methods," *Neural Networks*, vol. 99, pp. 158 – 165, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0893608018300054

[12] H. H. Aghdam, E. J. Heravi, and D. Puig, "A practical approach for detection and classification of traffic signs using convolutional neural networks," *Robotics and Autonomous Systems*, vol. 84, pp. 97 – 112, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S092188901530316X

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

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[15] R. F. Rachmadi, K. Uchimura, and G. Koutaki, "Spatial pyramid convolutional neural network for social event detection in static image," in *International Student conference on Advanced Science and Technology (ICAST)*, 2016.

[16] ——, "Road sign classification using spatial pyramid convolutional neural network," in *IIEEJ International Workshop on Image Electronics and Visual Computing*, 2017.

[17] K. Zuiderveld, "Contrast limited adaptive histogram equalization," *Graphics gems*, pp. 474–485, 1994.

[18] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proceedings of the 22Nd ACM International Conference on Multimedia*, ser. MM '14, 2014, pp. 675–678.

[19] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, "On the importance of initialization and momentum in deep learning," in *International conference on machine learning*, 2013, pp. 1139–1147.

[20] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[21] 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.

[22] A. Canziani, A. Paszke, and E. Culurciello, "An analysis of deep neural network models for practical applications," *arXiv preprint arXiv:1605.07678*, 2016.