Robust Traffic Signs Classification using Deep Convolutional Neural Network

1st Amine Kherraki
IMAGE Laboratory, School of Technology
Moulay Ismail University of Meknes
Meknes, Morocco
amine.kherraki.9@gmail.com

2nd Muaz Maqbool
Head of AI Department
OMNO AI
Lahore, Pakistan
muaz.maqbool@omno.ai

3rd Rajae El Ouazzani
IMAGE Laboratory, School of Technology
Moulay Ismail University of Meknes
Meknes, Morocco
elouazzanirajae@gmail.com

2022 International Conference on Intelligent Systems and Computer Vision (ISCV) | 978-1-6654-9558-5/22/$31.00 ©2022 IEEE | DOI: 10.1109/ISCV54655.2022.9806122

***Abstract*—Smart traffic management systems have recently piqued the interest of scientists and researchers due to the enormous growth in the number of vehicles. In fact, Intelligent Transportation Systems (ITS) can handle numerous problems by using computer vision such as; traffic sign detection, recognition, and classification. Lately, Deep Convolutional Neural Network (DCNN) has been exceedingly used in traffic signs classification thanks to the powerful feature extraction and robust prediction. However, the majority of related work focuses on one aspect, the accuracy, or the parameters requirement, which makes the task unsuitable for real-time or practical uses. To address this issue, we propose a novel efficient, and lightweight neural network for traffic signs classification in road scenes. Our proposed network is able to save parameter resources while maintaining high accuracy. We mention that we have used the Belgium Traffic Sign dataset (BelgiumTS) to prove the efficiency of our proposed model in terms of accuracy and parameters requirements.**

***Index Terms*—Autonomous Driving, Image Classification, Convolution Neural Network, Deep Learning, Computer Vision, Traffic Signs Classification, Belgium Traffic Sign Dataset.**

I. Introduction

Traffic signs are an important component of road flux and are extremely effective for alerting and guiding drivers and maintaining road safety. In ITS, traffic signs classification is a critical task for self-driving. However, due to the complexity of the external environment, traffic signs classification in real-time is far more difficult than many other computer vision tasks [1]. More recently, CNN has evolved into a powerful tool for computer vision [2], leading to the application of CNN for a variety of tasks, such as object recognition, segmentation, and classification [3], [4]. Actually, classifying traffic signs is one of the frequently accomplished tasks in ITS by using different machine learning and deep learning techniques [5]. Fig. 1 shows some images from the Belgium Traffic Signs (BelgiumTS) dataset [6]. This dataset provides a large number of traffic signs samples, therefore it is ideal to run several simulations in order to assess the traffic signs classification models. In spite of the attempts made in the traffic signs classification topic, this task is still challenging. Several approaches have already been used to investigate this issue. In the literature, several approaches have been carried out to classify traffic signs, including Boosting [8], and Support Vector Machines (SVM) [9] classifiers. To properly classify images using these methods, several feature extractors techniques such as Histograms of Oriented Gradients (HOG) [10], and Scale Invariant Feature Transform (SIFT) [11] are needed. However, the latter requires much time processing and speed compared to the new techniques.

Fig. 1. Examples of traffic signs images from the BelgiumTS dataset [7].

More recently, Deep Learning approaches, particularly CNN, have shown promising results in traffic signs classification [12]–[14]. In [15], a new approach that combines 3-D point clouds and 2-D images to detect and classify traffic signs based on hierarchical models has been proposed. The authors achieve an encouraging result, however, no information is provided about the parameters numbers. We can note from relevant previous research that many works do not take the parameter count seriously and focus only on accuracy, which makes the task inappropriate for real-time uses. To address the challenge, we propose a novel CNN that can make a balance between accuracy and parameter requirements, which makes it suitable for real-time applications. The workflow of traffic sign classification is depicted in Fig. 2.

There are several processes that must be completed in order to evaluate the proposed network. We begin by loading the data with class labels, then we build our model and train it on the trainset. Then, the model is assessed and used to assign the proper class labels to the images in the test set. In this paper, the main contributions are as follows:

- The proposition of a novel, efficient, and lightweight

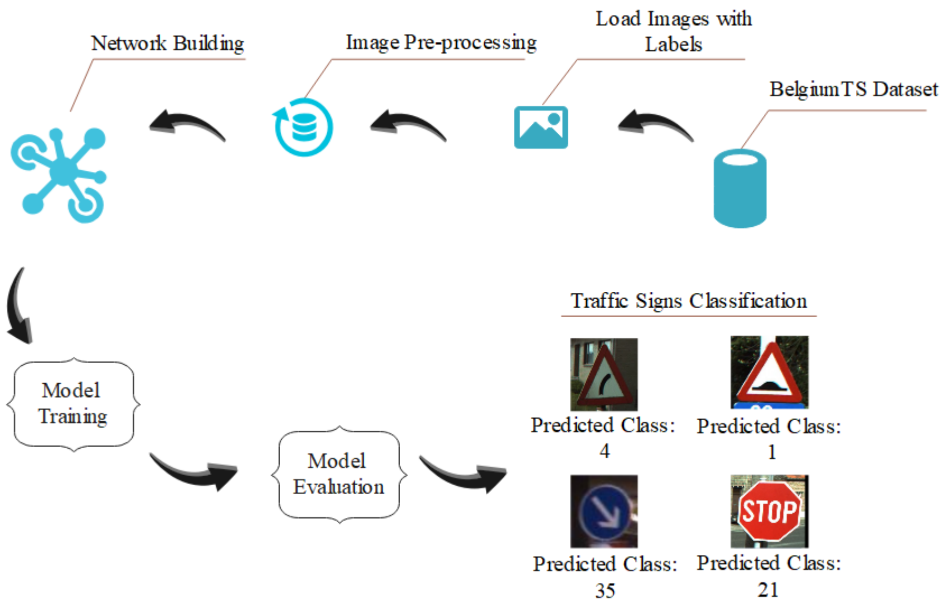

Fig. 2. The workflow of the traffic signs classification.

CNN model which does not require a large number of parameters.

- The proposed network achieves good results in terms of accuracy.
- The proposed model can be used in real-time and practical cases due to the reduced parameter requirements.

The rest of the paper is structured as follows. Section II discusses several approaches for traffic signs classification. In section III, a lightweight and efficient neural network is proposed. Next, we test our proposed network on the difficult BelgiumTS dataset. The experimental results are presented in section IV. Finally, in section V, the conclusion and perspectives are discussed.

II. Related Work

In this section, we will review some recent and important works that have been carried out on the classification of traffic signs. According to [1], a new deep learning approach to classify traffic signs based on transfer learning has been proposed. The authors used a pre-trained CNN model based on VGG16 layers. However, the authors did not provide more information about the number of parameters. But, to our knowledge, the VGG family networks require many parameter resources, therefore, it is not suitable for real-time uses [16]. Later, in [17], a new and intelligent system has been proposed by implementing a traffic signs image classification using CNN. The authors achieved encouraging results in terms of accuracy, however, they did not achieve a good result in terms of parameters requirement. Which makes the model not appropriate for real-time applications. After, the authors in [18], have proposed a novel neural network based on MobileNet architecture to classify traffic signs. This later tries to find a combination of accuracy and resources requirement. But, the achieved results were not interesting and the authors did not provide information about the number of parameters. Next, a new neural network based on TS-module has been proposed in [19]. The authors have achieved encouraging results in terms of precision and parameters requirement, but, they have used around 60000 epochs to train their proposed CNN. In [20], the authors have proposed a new Deep Neural Network (DNN) that comprises convolutional layers with transformer networks. This latter has achieved good results in terms of precision. Furthermore, a large study has been done in order to find out a combination between the Complete Local Binary Pattern (CLBP), Gabor, and HOG feature extraction techniques with Extreme Learning Machine (ELM) classifier [21]. The authors have achieved encouraging results, however, the old methods need much time for training.

III. The Proposed Model

Fig. 3 shows the proposed model that aims to minimize the resources in terms of parameters number while maintaining high accuracy. We use fifteen convolutional layers with different filters and kernel sizes from one layer to the other, in order to get a lightweight model.

The proposed model consists of nine blocks, the first one contains three convolutional layers with kernel k= 3, and different filters of 256, 128, and 64, followed by two convolutional layers with kernel k= 5, and different filters of 32, and 16. The second one contains two convolutional layers with kernel k= 3, and different filters of 128, and 64, followed by two convolutional layers with kernel k= 5, and different filters of 32, and 16. The third block contains one convolutional layer with kernel k= 3, and filters of 64, followed by two convolutional layers with kernel k= 5, and different filters of 32, and 16. The fourth block contains two convolutional layers

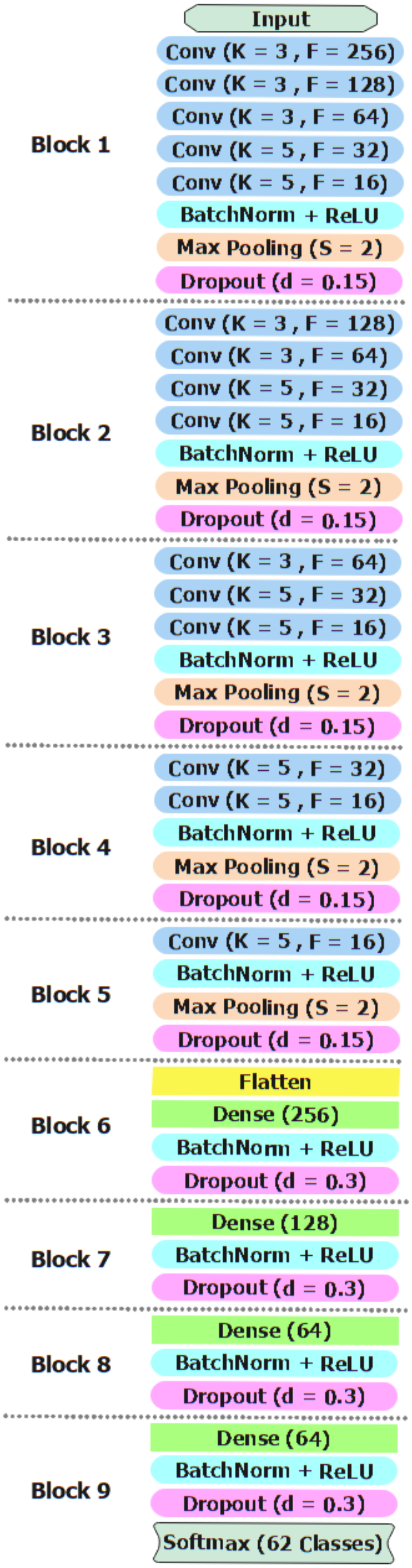

Fig. 3. The whole architecture of the proposed network for traffic signs classification.

with kernel k= 5, and different filters of 32, and 16. The fifth one contains one convolutional layer with kernel k= 5, and filters of 16. For each block, after the final convolutional layer, we have added a BatchNorm + ReLU, and a Max pooling layer with a stride of 2, in order to reduce the feature maps and computing time. In addition, we have added a dropout with a value of 0.15 for each block that contains a convolutional layer. After applying the flatten layer in the sixth block, we apply a Dense layer with a kernel value of 256, followed by BatchNorm + ReLU, and a Dropout layer with a value of 0.3 to get out overfitting. The seventh, eighth, and ninth blocks are quite similar to the sixth block, except that the kernel values in the seventh, eighth, ninth block are 128, 64, and 64 respectively. Finally, we have used a Softmax function in the output layer with a value of 62 which reflects the number of class labels. We focus on preventing overfitting in our model by adding multiple dropout layers, thus we can argue that our model will be more generalized.

IV. Experimental Results and Discussion

A. Experiments

All experiments were implemented by Python 3.7 using Keras framework and CUDA backend. Furthermore, we conduct substantial experimentation by training our model on the BelgiumTS dataset using two optimization techniques, specifically, Adam [22] and Stochastic Gradient Descent (SGD) [23] which has recently become an important optimization algorithm in deep learning. We state that we employ a learning rate of 0.001. Our network has been trained from scratch without pretrained weight using Google Colaboratory. We train our model across 200 epochs, and we compare the results with related work using the accuracy metric and the parameters number requirement.

B. Dataset

The BelgiumTS dataset [7] has been widely used in severals research work to classify and recognize traffic signs. This dataset contains 62 classes, 4575 images for train and 2520 images for test.

C. Results and Discussion

In this subsection, we will discuss the experimental evaluation of the proposed CNN model with different optimization algorithms including Adam and SGD. In addition, we have used and tested three shapes of input images, in particular, 32x32, 48x48, and 64x64. The obtained results are shown in TABLE I. First, in terms of accuracy, we can note that all the results related to the Adam optimizer are slightly superior to the SGD optimizer. Moreover, we can see that the 64x64 input format gets the best results in terms of accuracy. In terms of parameter requirements, it is normal for the small input model to get the best results because the number of convolutional layers will be small. The best result we found in our experiment is 96.64% with only 0.61M parameters, by using the Adam optimization algorithm with a 64x64 input image shape. TABLE II displays the results of some traffic signs classification work on the BelgiumTS dataset. We note that the proposed CNN has surpassed some existing works in terms of accuracy. However, our model exceeds two related works in terms of the number of parameters, while the rest of the relevant works did not provide information about the number of parameters.

TABLE I
DETAILED RESULTS ON THE PROPOSED CNN MODEL WITH DIFFERENT IMAGE SIZES AND OPTIMIZATION ALGORITHMS ON BELGIUMTS DATASET

Performance Evaluation			
Accuracy (%)	***Parameters (M)***	***Image size***	***Optimizer***
93.41	0.55	32x32	Adam
93.49	0.58	48x48	Adam
96.64	0.61	64x64	Adam
88.86	0.55	32x32	SGD
91.05	0.58	48x48	SGD
94.09	0.61	64x64	SGD

TABLE II
PERFORMANCE COMPARISON WITH RELATED WORK ON BELGIUMTS DATASET

Paper	Accuracy (%)	Parameters (M)
Mehta et al. [17]	97.04	10.34
Swaminathan et al. [18]	83.7	-
Arcos-Garcia et al. [20]	98.87	14.62
Mathias et al. [24]	97.83	-
Lu et al. [25]	96.29	-
Li et al. [26]	96.4	-
Gudigar et al. [27]	97.79	-
Our proposed CNN	96.64	0.61

In Fig.4 and 5, we plot the results of the curves obtained from our proposed model with different optimization algorithms and input image formats over 200 epochs. The left part of Fig. 4 shows the accuracy, and the right part shows the loss of the Adam optimizer on different input image

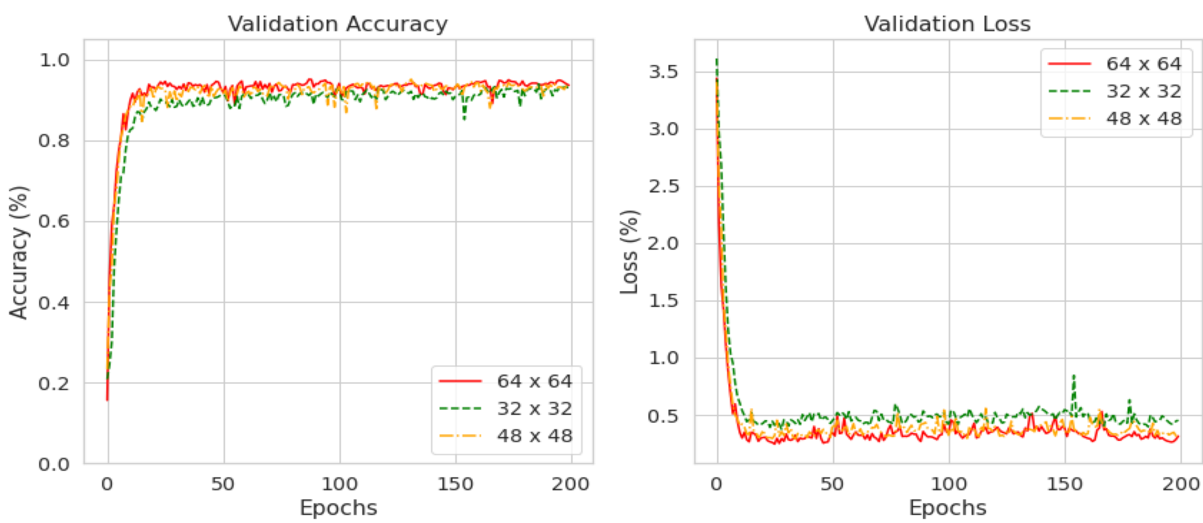

Fig. 4. Accuracy and loss curves of the proposed CNN using Adam optimizer.

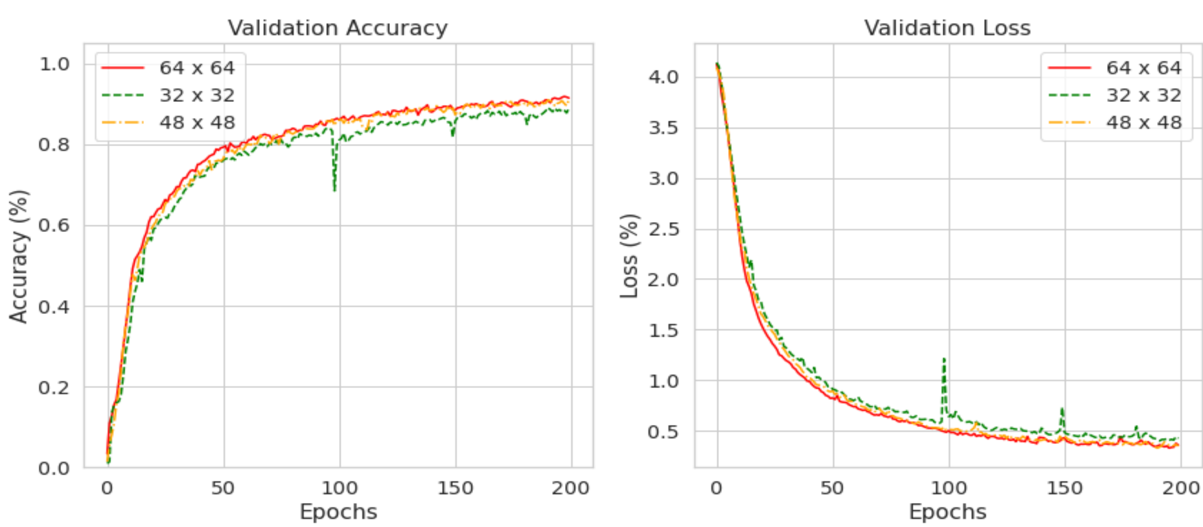

Fig. 5. Accuracy and loss curves of the proposed CNN using SGD optimizer.

shapes. From the accuracy and loss curves, we can see that during training our model has a good initialization and a good trace without oscillations, which means that we have avoided overfitting. Fig. 5 shows the accuracy and loss curves of the SGD optimizer on different input image shapes. In this case, we can notice that the Adam optimizer gave a good initialization in comparison with SGD. However, the SGD has a good trace, and with high probability, it will overtake the Adam optimizer with 300 or 400 epochs.

D. Prediction and test

In this subsection, we will show the prediction of our proposed model on the test set. After training our model on the BelgiumTS dataset, we have started generating the evaluation scores, and then we have run the test set using the predefined methods of Keras and TensorFlow. Afterward, we have used the Matplotlib library to display the images with their predicted classes. Fig. 6 shows some examples of the predicted classes using our proposed model. As shown in the figure, many test images have been well predicted however, some of them have been misclassified, which is very usual owing to a variety of issues such as image quality, motion blur, or poor lighting. The model works well on clear photos, and it works well on pictures with minor blur and illumination issues.

Fig. 6. Results of predicted classes using our best classification model.

V. Conclusion and Perspectives

In this paper, we propose a new CNN model for classifying traffic signs. We evaluated our proposed model on the BelgiumTS dataset. In our experiments, we used two different optimization algorithms, including Adam and SGD. Furthermore, we used three different input image sizes. Our proposed model gets an accuracy of 96.64% and uses 0.61M parameters only, which makes it useful and suitable for real-time applications. As a perspective, we are looking forward to implementing CNN models with blocks in order to achieve good accuracy and maintain a few parameter requirements.

References

[1] A. Jain, A. Mishra, A. Shukla, and R. Tiwari, "A novel genetically optimized convolutional neural network for traffic sign recognition: A new benchmark on belgium and chinese traffic sign datasets," *Neural Processing Letters*, pp. 1–25, 2019.

[2] M. Berrahal and M. Azizi, "Augmented binary multi-labeled cnn for practical facial attribute classification," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 23, no. 2, pp. 973–979, 2021.

[3] I. Idrissi, M. Mostafa Azizi, and O. Moussaoui, "A lightweight optimized deep learning-based host-intrusion detection system deployed on the edge for iot," *International Journal of Computing and Digital System*, 2022.

[4] I. Idrissi, M. Boukabous, M. Azizi, O. Moussaoui, and H. El Fadili, "Toward a deep learning-based intrusion detection system for iot against botnet attacks," *IAES International Journal of Artificial Intelligence*, vol. 10, no. 1, p. 110, 2021.

[5] A. Kherraki, M. Maqbool, and R. El Ouazzani, "Traffic scene semantic segmentation by using several deep convolutional neural networks," in *2021 3rd IEEE Middle East and North Africa COMMunications Conference (MENACOMM)*, pp. 1–6, 2021.

[6] R. Timofte, K. Zimmermann, and L. V. Gool, "Multi-view traffic sign detection, recognition, and 3d localisation," in *WACV*, 2009.

[7] R. Timofte, K. Zimmermann, and L. V. Gool, "Multi-view traffic sign detection, recognition, and 3d localisation," in *WACV*, 2009.

[8] A. Ruta, Y. Li, and X. Liu, "Robust class similarity measure for traffic sign recognition," *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, pp. 846–855, 2010.

[9] J. Greenhalgh and M. Mirmehdi, "Real-time detection and recognition of road traffic signs," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, pp. 1498–1506, 2012.

[10] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1, pp. 886–893 vol. 1, 2005.

[11] G. LoweDavid, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, 2004.

[12] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, "Introduction to the special issue on machine learning for traffic sign recognition," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, pp. 1481–1483, 2012.

[13] M. Boukabous and M. Azizi, "Review of learning-based techniques of sentiment analysis for security purposes," in *The Proceedings of the Third International Conference on Smart City Applications*, pp. 96–109, Springer, 2020.

[14] M. Boukabous and M. Azizi, "A comparative study of dl-based language representation learning models," *Indones. J. Electr. Eng. Comput. Sci*, vol. 22, no. 2, 2021.

[15] Y. Yu, J. Li, J. Li, C. Wen, H. Guan, H. Luo, and C. Wang, "Bag-of-visual-phrases and hierarchical deep models for traffic sign detection and recognition in mobile laser scanning data," *Isprs Journal of Photogrammetry and Remote Sensing*, vol. 113, pp. 106–123, 2016.

[16] A. Kherraki and R. E. Ouazzani, "Deep convolutional neural networks architecture for an efficient emergency vehicle classification in real-time traffic monitoring," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 11, pp. 110–120, mar 2022.

[17] S. Mehta, C. Paunwala, and B. Vaidya, "Cnn based traffic sign classification using adam optimizer," in *2019 International Conference on Intelligent Computing and Control Systems (ICCS)*, pp. 1293–1298, IEEE, 2019.

[18] V. Swaminathan, S. Arora, R. Bansal, and R. Rajalakshmi, "Autonomous driving system with road sign recognition using convolutional neural networks," in *2019 International Conference on Computational Intelligence in Data Science (ICCIDS)*, pp. 1–4, IEEE, 2019.

[19] W. Li, D. Li, and S. Zeng, "Traffic sign recognition with a small convolutional neural network," in *IOP conference series: Materials science and engineering*, vol. 688, p. 044034, IOP Publishing, 2019.

[20] Á. Arcos-García, J. A. Alvarez-Garcia, 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.

[21] S. Aziz, F. Youssef, *et al.*, "Traffic sign recognition based on multi-feature fusion and elm classifier," *Procedia Computer Science*, vol. 127, pp. 146–153, 2018.

[22] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[23] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2016.

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

[25] K. Lu, Z. Ding, and S. Ge, "Sparse-representation-based graph embedding for traffic sign recognition," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, pp. 1515–1524, 2012.

[26] W. Li, D. Li, and S. Zeng, "Traffic sign recognition with a small convolutional neural network," *IOP Conference Series: Materials Science and Engineering*, vol. 688, p. 044034, nov 2019.

[27] A. Gudigar, S. Chokkadi, U. Raghavendra, and U. R. Acharya, "An efficient traffic sign recognition based on graph embedding features," *Neural Computing and Applications*, vol. 31, no. 2, pp. 395–407, 2019.