Performance Analysis for Recognition and Detection of Traffic Signs on Indian Roads using Convolutional Neural Networks

Sohham Seal
Department of Mathematical and Computational Sciences
National Institute of Technology Karnataka, Surathkal
Mangalore, India
sohham.seal.29.08@gmail.com

Sonali Chakraborty
Department of Mathematical and Computational Sciences
National Institute of Technology Karnataka, Surathkal
Mangalore, India
al.sonali@nitk.edu.in

***Abstract*— Convolutional Neural Networks (CNNs) have seen tremendous growth in their application for image categorization and pattern recognition because of their high recognition rate, quicker execution, and higher accuracy. In the present study, the authors perform an analysis of various optimizers using the CNN model for the recognition and classification of traffic signals on Indian roads. The model is tested using 3, 4 and 6 layered CNN models which are contrasted with one another and compared based on their architectures, the impact of optimizers and the number of training epochs. The models are created and trained in real-time on the NVIDIA Tesla K80 GPU offered by Google Colaboratory's online services.**

***Keywords*— Traffic signs, Convolutional Neural Network, Traffic Sign Recognition, Keras, Google Colab., Indian highway traffic signs.**

I. Introduction

Road traffic signs are used worldwide in order to control the flow of traffic and to alert drivers. The traffic signs are designed using various symbols with different colours helpful for recognition by human beings. The capacity for visual perception in humans varies from person to person and is dependent on their physical and mental health. Moreover, it is also observed that the traffic signs get damaged, concealed or blurred and therefore practically impossible to recognize. These problems lead to the occurrences of fatal accidents on roads and highways, and therefore a need for automated road sign identification arises. Such automated tools can assist the drivers as a supplemental tool for precautionary measures while driving on the road. As a result, the Driver Support Systems (DDS) thus developed must be trustworthy and quick to identify traffic signs in order to assist the driver while travelling.

Additionally, the need for such assistance is required during poor weather conditions, and therefore, higher robustness and accuracies are desired. They should not be impacted by lighting, shadows or signs that are crooked, bent, blurry, corroded, or worn out. Despite its importance, Biswas et al. [2] claim that traffic sign recognition and identification is one of the least researched issues in the field of DDS. The majority of this field's research is focused on auto-pilot functions, which call for the recognition of road borders or obstructions. However, as we have already covered, the leading causes of accidents include the driver's failure to read prohibitory traffic signs, poor vision owing to bad weather, oncoming vehicle headlights, stress, tension and vision acuity. Therefore, these systems may not only be used in autonomous automobiles but also to alert drivers of prohibitory traffic signs in advance to avoid potential hazards.

Møgelmose et al. [4] also go over the need to take the driver's visual system into account when conducting Traffic Sign Recognition (TSR) studies. It claims that while designing human-centred traffic sign identification for driver assistance, we should take into account both the cognitive burden and the driver's focus of attention. The systems should be trained to detect and highlight signs that the driver has not seen. The system should be able to detect all traffic signs; however, they should avoid presenting all of them to the driver to reduce confusion. It needs to highlight only those signs that are easily overlooked. Additionally, they can keep track of the driver's field of view in order to warn them of any signals that are outside of their line of sight or peripheral vision.

The Ministry of Road Transport and Highways' Transport Research Wing study [10] states that in 2019, there were more than 4.4 lakh traffic accidents, with highways accounting for about 30.5 percent of those incidents. Due to the nationwide lockdown, however, the number of traffic accidents fell to 3.6 lakh in 2020, although the percentage of highway accidents maintained at about 31.8 percent of the total. According to the survey, drunk driving, exceeding the posted speed limit, or using a cell phone while driving are the leading causes of highway accidents; out of those, significant highway accidents involving over speeding account for almost 70% of the total.

Due to the pressing necessity for traffic sign detecting systems in future driverless vehicles and the convenience of a typical passenger, traffic sign recognition has expanded quickly. These systems perform well in terms of detection, high recognition rates, real-time implementation, several kinds of traffic signs as recognition objects, and resilience in various situations.

Utilizing the modified Generalized Hough Transform (GHT) technique for localization, Shustanov and Pavel [1] suggested a strategy that included two key steps: classification using a straightforward template matching procedure after the classification stage. The model's accuracy was 97.3 percent when applied to the German Traffic Sign Recognition Benchmark (GTSRB) and the German Traffic Sign Detection Benchmark (GTSDB) datasets [8] [9]. However, only the blue and red pixels are retrieved from the photos during image pre-processing using the HSV (Hue, Saturation and Value) colour space, resulting in a model that only categorizes traffic signs of such colour types and would fail on numerous significant yellow, green, orange or brown-coloured Indian signals. Islam [5] offered an alternative strategy that involved developing two distinct models, one for categorizing shapes and the other for signs. Additionally, it involved feeding the neural network both positive and negative pictures. In the case of specified and well-formed datasets of the British and Ukrainian traffic signs, the model had a high accuracy of approximately 90–93, while it had an inferior precision of around 69 percent on the Bangladeshi signs. Models with high accuracy but low precision trained on positive and negative pictures imply that the classifier will not only identify and classify signs but also mistakenly label innocuous items as traffic signs.

Ying Sun et al. [7] tried to reduce the mistakes and achieved an accuracy of 98.2 percent on the GTSRB dataset. However, the method, which used the Hough Transform and was only concerned with circular traffic signs, is unsuitable for classifying other shapes of traffic signs.

Similar to the approaches stated above, Biswas et al. [2] used a two-step model for recognizing speed limit signs (alone), which first extracts the sign from the image using the Circular Hough transform (CHT), and then digit segments and classifies the data using an SVM classifier. They created an extremely reliable system with 98 percent accuracy.

Traffic sign detection is generally regarded as a challenging problem due to various complexities, for example, the diversified backgrounds of traffic sign images. The present study uses multiple CNN models to train the data obtained from Indian roads with a primary focus on the Indian highway traffic signs. The presented work is organized as follows: Section 2 gives a description of the dataset used in the study, while the comprehensive system description is given in Section 3. Section 4 discusses the experimental observations of the study, and the concluding remarks are discussed in Section 5.

II. Analysis Domain

The dataset used in the present study consists of images available from the world wide web and some of the images captured from the streets of Mangalore city, India. The dataset comprises 5691 images out of which 3984 i.e. 70% of the images are used for training while 1707 images are used for testing the models. The images are categorized into nine classes based on prohibitory, mandatory, and road construction safety signs. The dataset comprises good quality images which are clearly visible and also damaged and distorted images that are difficult to recognize such as those that are rusted, corroded, partly covered, blurry, bent and faded. Figure 1 illustrates some of the images used in the study that have the aforementioned defects.

The networks were trained using Python Keras [12] library running on top of the Tensorflow library [13]. Keras allows for easy and fast prototyping of neural networks. During the calculations, we utilized the GPU library provided by Google Colaboratory (Colab), which allows parallel computing on Google cloud servers. The GPU runtime was used to train all four networks. The models' training took an average of forty-eight minutes on 50 epochs, and the length of time rose with the number of epochs. The training was 40 times quicker with GPU use than with CPU. To inspect and resize the photos, we made use of the OpenCV library [14]. All processes were performed on a computing unit equipped with: Nvidia Tesla K80 GPU with 12.68GB RAM and 107.72GB of storage space.

Fig. 1. (A) No left turn sign (B) No cycling sign (C) Falling rocks sign (D) Compulsory right turn sign (E) Faded and broken no-cycling sign (F) Rust-covered wild animals sign (G) Distorted men at work sign (H) Snow-covered slippery road sign.

III. System Overview

Recent years have seen the ubiquitous use of deep learning as a technique for object recognition across several industries. When the size of the dataset is huge, they produce very high accuracy in image classification compared to traditional computer vision algorithms [6]. The neural network architectures that were used to create these models make use of parameterized, sparsely connected kernels that preserve the images' spatial characteristics [3].

A. *Convolutional Neural Networks (CNN)*

CNNs are a form of Artificial Neural Network (ANN) that is particularly well suited for image processing and analysis, designed to automatically and adaptively learn spatial hierarchies of features, from low to high-level patterns. It is a supervised deep learning approach that requires large labelled data for training on the network [11]. They are a mathematical construct typically composed of three types of layers: convolution, pooling, and fully connected layers. Feature extraction is carried out by the first two layers—convolution and pooling—while the third layer—a fully connected layer—maps the retrieved features onto the output, such as classification. Extracted characteristics may gradually and hierarchically become more sophisticated as one layer feeds its output into the following layer. The process of optimizing parameters such as kernels is called training, which minimizes the difference between outputs and labels through an optimization algorithm.

B. *Analysis of CNN-based models*

Four different CNN architectures are used in the present study with the sequential models using ReLU as an activation function for the non-linear part along with pooling and dropout layers. The optimizers used in the model are AdaGrad, Adam, RMSProp and SGD to train the four CNN models. The model descriptions used in the study are described as follows:

1) Model 1- One convolutional layer followed by two fully-connected layers having 40,147,081 trainable parameters.

2) Model 2- Two convolutional layers and two fully-connected layers having 38,552,233 trainable parameters.

3) Model 3- Four convolutional layers and two fully connected layers with 34,754,60 trainable parameters.

4) Model 4- Four convolutional layers and three fully connected layers with 139,099,465 trainable parameters.

The training set for each of these models has the following specifications:

- Learning rate: 0.001,
- Metric: Accuracy,
- Loss Function: Categorical Cross-Entropy,
- No. of epochs: Ranging from 15 to 500.

IV. Experimental Observations

Among the models used for training and testing, those trained using the Adam optimizer achieved the greatest percentage accuracy. The maximum level of precision was attained by Model 4. The model displayed 93.8% training accuracy and 82.0% testing accuracy. Figures 2 to 5 compare the results of the various optimizers, whereas Table I describes all the maximum accuracy levels reached by each model. The plots also show that the highest accuracy for Adam is at 50 epochs, whereas it is between 100 and 200 for SGD, 50 for RMSProp, and 150 for AdaGrad. Moreover, the plots also prove that the Adam optimizer, which offers the highest training and testing accuracies, continues to be the best optimizing function.

Fig. 2. Testing accuracy using Adam optimizer for models 1, 2, 3 and 4.

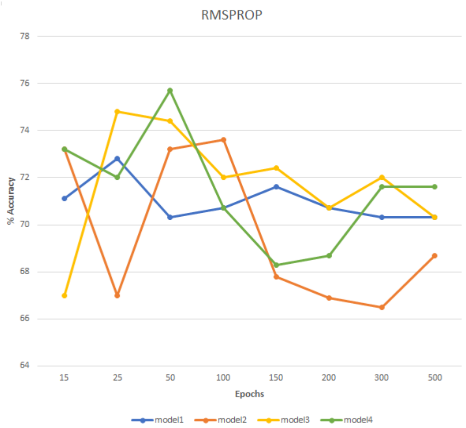

Fig. 3. Testing accuracy using RMSProp optimizer for models 1, 2, 3 and 4.

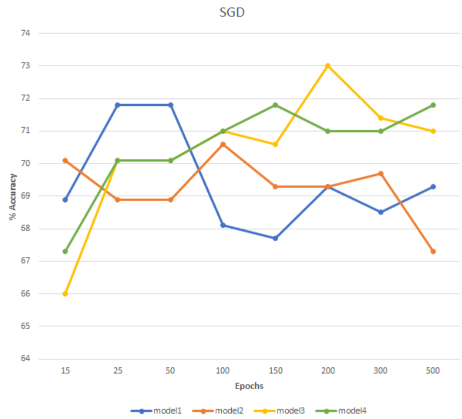

Fig. 4. Testing accuracy using SGD optimizer for models 1, 2, 3 and 4.

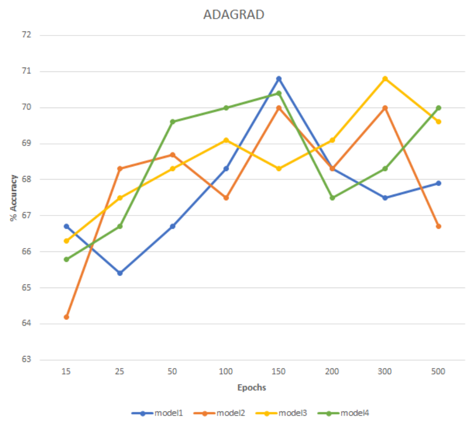

Fig. 5. Testing accuracy using AdaGrad optimizer for models 1, 2, 3 and 4.

TABLE I. Accuracy Percentage on Training and Testing Data using Models 1, 2, 3 and 4.

Model Name	Maximum percentage accuracy attained	
	Training	***Testing***
Model 1	94.2	80.0
Model 2	94.0	79.5
Model 3	93.8	82.0
Model 4	93.7	81.6

V. Conclusion and Future Work

This paper suggested four CNN models with autonomous feature generation and output prediction for Indian highway traffic signs in this paper. The models coalesce the advantage of CNNs in recognizing traffic signs. The experimental findings showed that the classification accuracy of our suggested strategy for the custom traffic sign dataset was 82.0%. We have also contrasted the four models that have been suggested to compare the effect of the different optimizers and epochs. In the future, the suggested model can also be enhanced to account for both positive and negative pictures.

Deep learning requires a vast amount of training data, but creating such a large-scale training data set of Indian traffic signs is a difficult task. As traffic sign images are not evenly distributed across a region, assembling a balanced dataset of all classes is one of the key challenges. For this reason, popular datasets such as the GTSDB, GTSRB and the Belgium Traffic Sign (BTS) are more effective in TSR research, however, they are mostly limited to European traffic signs. As a result, more in-depth research and fieldwork are required to gather a larger Indian traffic sign dataset which will increase the models' precision and accuracy, making them more dependable.

REFERENCES

[1] Alexander Shustanov and Pavel Yakimov, "CNN Design for Real-Time Traffic Sign Recognition," Procedia Engineering, vol. 201, pp. 718-725, 2017, ISSN 1877-7058, doi: 10.1016/j.proeng.2017.09.594.

[2] Biswas, Rubel et al. "Detection and classification of speed limit traffic signs," 2014 World Congress on Computer Applications and Information Systems (WCCAIS) (2014): 1-6.

[3] S. Mahapatra, "Why Deep Learning over Traditional Machine Learning?", Accessed on: July 29, 2022. [Online]. Available: https://towardsdatascience.com/why-deep-learning-is-needed-over-traditional-machine-learning-1b6a99177063.

[4] A. Mogelmose, M. M. Trivedi and T. B. Moeslund, "Vision-Based Traffic Sign Detection and Analysis for Intelligent Driver Assistance Systems: Perspectives and Survey," in IEEE Transactions on Intelligent Transportation Systems, vol. 13, no. 4, pp. 1484-1497, Dec. 2012, doi: 10.1109/TITS.2012.2209421.

[5] M. T. Islam, "Traffic sign detection and recognition based on convolutional neural networks," 2019 International Conference on Advances in Computing, Communication and Control (ICAC3), 2019, pp. 1-6, doi: 10.1109/ICAC347590.2019.9036784.

[6] R. Poojary and A. Pai, "Comparative Study of Model Optimization Techniques in Fine-Tuned CNN Models," 2019 International Conference on Electrical and Computing Technologies and Applications (ICECTA), 2019, pp. 1-4, doi: 10.1109/ICECTA48151.2019.8959681.

[7] Y. Sun, P. Ge and D. Liu, "Traffic Sign Detection and Recognition Based on Convolutional Neural Network," 2019 Chinese Automation Congress (CAC), 2019, pp. 2851-2854, doi: 10.1109/CAC48633.2019.8997240.

[8] J. Stallkamp, M. Schlipsing, J. Salmen and C. Igel, "Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition," Neural Networks, 2012.

[9] S. Houben, J. Stallkamp, J. Salmen, M. Schlipsing and C. Igel, "Detection of Traffic Signs in Real-World Images: The German Traffic Sign Detection Benchmark," International Joint Conference on Neural Networks, 2013.

[10] Ministry of Road Transport & Highways, "Road Accidents in India 2020." 2020, Accessed on: July 29, 2022. [Online]. Available: https://morth.nic.in/sites/default/files/RA_2020.pdf.

[11] A. Kwasigroch, A. Mikolajczyk and M. Grochowski, "Deep neural networks approach to skin lesions classification - A comparative analysis," 22nd International Conference on Methods and Models in Automation and Robotics, MMAR 2017, Międzyzdroje, Poland, August 28-31, 2017, 1069–1074.

[12] F. Chollet, "Keras," March 27, 2015, Accessed on: July 10, 2022. [Online]. Available: https://github.com/fchollet/keras.

[13] Martín Abadi and others, "TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems," November 5, 2015. Accessed on: July 10, 2022. [Online]. Available: https://github.com/tensorflow/tensorflow.

[14] G. Bradski, "The OpenCV Library," Dr Dobb's Journal of Software Tools, 2000. Accessed on: July 10, 2022. [Online]. Available: https://github.com/opencv/opencv.