Road Sign Classification using Convolutional Neural Network

Mohsinali Vijapura*(AU2040080) , Manan Vadaliya*(AU2040264) ,

Pruthviraj Dodiya*(AU2040175) , Harshrajsinh Vaghrola* (AU2040195)

School of Engineering and Applied Science, Ahmedabad University

***All Authors have contributed equally**

***Abstract*— We present a computer vision-based method for accurate traffic sign tracking and recognition. In an autonomous car, such a technology offers crucial support for driver assistance. Convolutional Neural Networks (CNNs) for object identification tasks, including traffic sign recognition and classification. While preprocessing the data set images we have converted them into grayscale, standardize the lighting in images and normalize the image from 0 to 1 instead of 0 to 255 . The article presents the development of a traffic sign recognition system using a CNN and compares various CNN architectures. The model was trained using Tensorflow and Keras and achieved an accuracy of 98.58%.**

***Keywords*— traffic signs recognition, computer vision, CNN, soft-max layer, image preprocessing.**

I. Introduction

Since a driver assistance system controls both the flow of traffic and the driver, its ability to recognise traffic signs is a major problem. Severe accidents occur when distracted or mentally unbalanced drivers misread road signs. Consequently, automated traffic sign identification is a key area of research for autonomous navigation systems. To accurately identify traffic signs in a real-time environment, such a system must be quick and effective. Also, they must deal with various issues that may reduce the effectiveness of detection and recognition. Variations in illumination (fog, shadow and sunlight), motion blur, weather and sign occlusion are some of these issues. Being effective is important since one incorrectly identified or unnoticed indicator could interfere with the navigation system.

Indeed, there is no 100% accuracy guarantee offered by the current technologies. This has inspired numerous academics to find ways to enhance the effectiveness of traffic sign detection, tracking, and recognition in challenging environments, which is the goal of the method we've just described. As a result, we developed a new technique for quickly detecting, monitoring, and classifying traffic signs from moving vehicles under challenging circumstances. Designing a system that can recognise traffic signs is the main objective. This technology can help local or national authorities in the upkeep and modernization of their road and traffic signs by automatically identifying and classifying one or more traffic signs from a complex image captured by a vehicle's camera.

The TensorFlow library, a massively parallel architecture for multithreaded programming, is used to train the neural network. To read traffic signs, follow these three steps. Traffic signals are first divided into subregions based on the color of the supplemental signs and the main sign using a region proposal based on segmentation. Second, increasing the amount of data accessible for deep neural network learning by including images in the training dataset.

The purpose of this effort is to find a solution to the small data problem. It is used to enhance deep learning's capabilities. Eventually, a deep neural network is used to combine the original and suggested images in a dataset of photographs.

II. Literature Survey

Using the visual key of traffic sign properties like color and form, the picture is split in the detection stage. As traffic signs are made of primary colors that contrast sharply with their surroundings, they really serve to express fundamental information. As a result, many techniques start with a segmentation stage within a particular color space. An RGB picture is often the result of a mounted camera. Nevertheless, due to its sensitivity to changes in lighting, the RGB color space is not suited for detecting the colors of signs.

In order to identify the sign colors in the image, some writers employed a color ratio between the RGB intensity components, while others used just one RGB component as a reference. The Hue Saturation Intensity (HSI) method and HSV have been widely utilized to lessen the reliance on light variance.

Contrarily, there are techniques built on the TS shape that completely disregard color information in favor of shape data from grayscale photos. To find the spots of interest in the TS picture, for instance, the local radial symmetry method was used. This method uses a center point vote for circular signs and a line vote for regular polygons on the gradient of a grayscale picture.

III. IMPLEMENTATION

In order to identify traffic signs, use the deep learning toolkit TensorFlow. Training is done using the entire dataset. The 43 different types of traffic indicators may be categorised by the algorithm that was created. Building network architecture involves pre-processing the pictures, giving them a depth of 1, and utilising the CNN model with max pooling.

Despite this, the design of network architecture mostly on heuristic approaches. The relationship between the network depth and the data volume ought to be negative. When there is a large network and limited data, an overfit model is more likely. On the other side, a shallow network with a lot of data would not offer any benefits.

The required libraries, including NumPy, Matplotlib, OpenCV, pandas, and Keras, are first imported by the programme. Additionally, it imports particular modules from the Keras library, including Dense, Dropout, Flatten, Conv2D, MaxPooling2D, and ImageDataGenerator. The path to the dataset, the name of the file containing the class labels, the batch size, the number of epochs, the picture dimensions, the test ratio, and the validation ratio are some more hyperparameters that we declare.

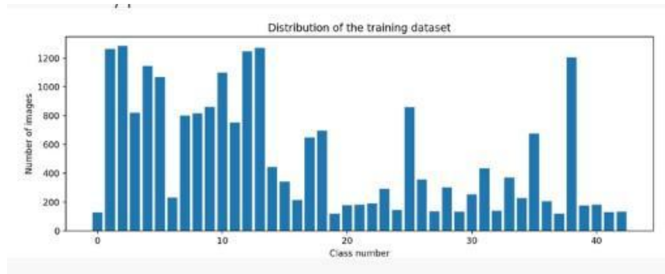
The train_test_split() method from the scikit-learn library is used to split the imported photos from the dataset into the training, validation, and testing sets. The CSV file is read, and several pre-processing operations including grayscale, equalise, and pre-processing are performed. These operations standardise illumination in a picture, convert it to grayscale, and normalise values between 0 and 1 rather than 0 to 255.

The CNN model architecture is then defined by the programme using Keras' Sequential API. Multiple layers, including Conv2D, MaxPooling2D, Flatten, Dense, and Dropout layers, are present in the model. Additionally, the Adam optimizer, categorical cross-entropy loss, and accuracy are used to construct the model. The model is then trained using the fit() method, and the model's effectiveness is assessed using the evaluate() function.

The design has one soft-max layer and several convolutional layers. In order to forecast a multinomial probability distribution using neural network models, the soft-max function must be used as the activation function in the output layer. In other words, the activation function for multi-class classification issues is soft-max. TensorFlow provides tools for visualizing models at various abstraction levels, from complex mathematical operations to high-level mathematical processes. The training and testing datasets are divided by 80:20.

50 photos from the train dataset are processed in batches throughout each training session. The intermediate accuracy was calculated using a batch of 50 photos from the test dataset every 100 iterations. Accuracy was decided after obtaining proper training.

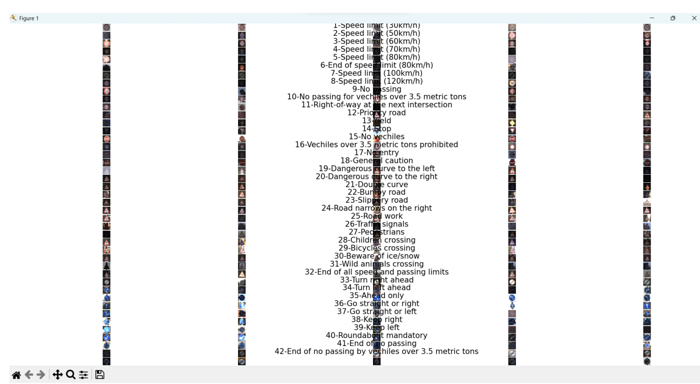
IV. RESULTS

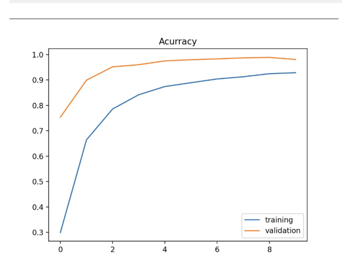


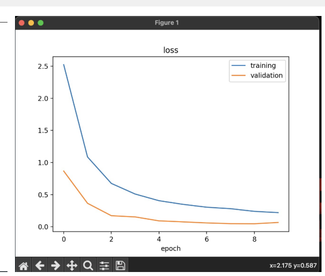
The above picture describes the frequency distribution of each class(i.e type of road sign) from the dataset.

The above image denotes the pre-processing of the dataset images by converting them to :

- Grayscale
- Standardizing the lighting in image
- Normalizing the image from 0 to 1 instead of 0 to 255







V. ACKNOWLEDGEMENT

The solution to the challenge of classifying traffic indicators is examined in the suggested study. When used in conjunction with preceding work's pre-processing and localization steps, the proposed approach for traffic sign classification yields 90.58 percent of accurately classified images. The suggested categorization approach is put into practice using the Tensor Flow framework.

VI. FUTURE WORK

One potential area for future work is to investigate the performance of the proposed approach on datasets from different regions or countries. This could help identify any variations or discrepancies in traffic sign designs and ensure that the model is generalized to diverse settings.

Another area for improvement is the incorporation of temporal information into the model. Traffic signs often appear in a specific sequence, and the context provided by preceding signs can help disambiguate the meaning of the current sign. Therefore, incorporating temporal information, such as the order and duration of sign appearances, could help improve the accuracy of the model.

Additionally, we can also explore the use of more advanced deep learning techniques, such as attention mechanisms, to further improve the model's ability to focus on relevant features in the input data.

VII. REFERENCES

A. Kapoor, N. Nehra, and D. Deshwal, “Traffic signs recognition using CNN,” in 2021 International Conference on Industrial Electronics Research and Applications (ICIERA), 2021, pp. 1–4.

N. Veličković, Z. Stojković, G. Dimić, J. Vasiljević, and D. Nagamalai, “Traffic sign classification using Convolutional Neural Network.” .

S. Reddy, D. Dash, and N. Rakesh “Image Classification Using Machine Learning Techniques for Traffic Signal”. In Intelligent Data Communication Technologies and Internet of Things (pp. 233-244), Springer, Singapore 2021.

Luo, H., Yang, Y., Tong, B., Wu, F., & Fan, B. (2017). Traffic sign recognition using a multi-task convolutional neural network. IEEE Transactions on Intelligent Transportation Systems, 19(4), 1100-1111.

Shustanov, A., & Yakimov, P. (2017). CNN design for real-time traffic sign recognition. Procedia engineering, 201, 718-725.

Zhu, Y., Zhang, C., Zhou, D., Wang, X., Bai, X., & Liu, W. (2016). Traffic sign detection and recognition using fully convolutional network guided proposals. Neurocomputing, 214, 758-766.

A. Vennelakanti, S. Shreya, R. Rajendran, D. Sarkar, D. Muddegowda and P. Hanagal, "Traffic Sign Detection and Recognition using a CNN Ensemble," 2019 IEEE International Conference on Consumer Electronics (ICCE), Las Vegas, NV, USA, 2019, pp. 1-4, doi: 10.1109/ICCE.2019.8662019.

H. S. Lee and K. Kim, "Simultaneous Traffic Sign Detection and Boundary Estimation Using Convolutional Neural Network," in IEEE Transactions on Intelligent Transportation Systems, vol. 19, no. 5, pp. 1652-1663, May 2018, doi: 10.1109/TITS.2018.2801560.