Detection of Traffic Signs Using Deep Learning Techniques

J.Sirisha[1]

Assistant Professor, Dept of IT, PVP Siddhartha Institute of Technology, Vijayawada, India,

Email: sirisha.j@pvpsiddhartha.ac.in

G.D.K.Kishore[2]

Assistant Professor, Dept of IT, SRK Institute of Technology, Vijayawada, India

Email: kishore.galla1@gmail.com

ABSTRACT

As the number of vehicles on the road increases in today's world, so do the number of accidents, and according to reports, India ranks first in terms of the number of accidents per country. This is due to a variety of factors, including lax enforcement of legislation, carelessness, and so on. People do not recognise or follow traffic signs, which is one of the reasons. All traffic rules must be understood and followed by automobiles. Several experts and large businesses, like Tesla, Uber, Google, Mercedes-Benz, Toyota, Ford, Audi, and others, are experimenting with self-governing automobiles and self-driving vehicles in the domain of Artificial Intelligence and advances. As a result, in order to achieve precision in this technology, cars should have the ability to understand traffic signs and make decisions. As a result, utilizing Keras and deep learning approaches, we developed a traffic sign recognizer that can warn a vehicle's driver to an approaching traffic sign and tell them to follow it. This may aid in the reduction of traffic collisions.

KEYWORDS: CNN, computer vision, Vehicle Data, Traffic sign, Keras

1. Introduction

Every day, over 400 traffic accidents occur in India, according to official statistics. Road signs aid in the prevention of traffic accidents, assuring the safety of both drivers and pedestrians. Furthermore, traffic signals ensure that road users follow specified regulations, reducing the chances of traffic offences. The usage of traffic signals also helps with route guidance. All road users, including automobiles and pedestrians, should prioritize road signals. For a variety of causes, such as focus issues, tiredness, and sleep deprivation, we overlook traffic signs. Each individual, whether a traveller, driver, or walker, would have seen several sign boards along the side of the road that fulfil vital needs. These important pieces of street gear serve as course aides, warnings, and traffic controllers. Signs, as traffic management devices, require full study, regard, and appropriate motorist response. With the advent of automated traffic and the resulting increase in street congestion, many people have obtained graphic signs and have normalized their signs to enable worldwide travel when language differences would be a problem. In congested rush-hour traffic, the driver may fail to notice traffic signs, resulting in accidents. In such cases, pre-programmed street sign recognition comes in handy.

Poor vision, the impact of the outside world, and environmental circumstances are all factors that contribute to ignoring the signs. Using a technology that can recognize traffic lights and advise and alert the driver is even more crucial. To recognize signals, image-based traffic-sign recognition technology examines images acquired by a car's front-facing camera in real time. They assist the driver by issuing warnings to him or her. A vision-based traffic sign recognition system's primary components are the identification and recognition modules. The detection module finds the sign area in the image/video, and the recognition module recognizes it. During the detection procedure, the sign areas with the highest probability are chosen and sent into the recognition system to classify the sign.

Various machine learning techniques, such as SVM, KNN, and Random Forest, can be utilized to recognize traffic signs [6]. The main disadvantage of these algorithms is that feature extraction must be done individually; on the other hand, CNN will extract features on its own [3]. As a result, a convolutional neural network is used in the proposed system.

2. Proposed System

The street traffic pictures are caught by vehicle-mounted cameras introduced on the keen vehicles, and the traffic sign recognition means to remove the intrigued traffic sign areas from the current street traffic pictures adequately. Nonetheless, in various outer conditions, the characteristics of the obtained pictures are lopsided, and these characteristics must be successfully recognized after the inalienable qualities of traffic signs, for example, shading and shape. In this segment, it fundamentally incorporates two sections: traffic sign division dependent on the shading space and traffic sign location dependent on shape highlights.

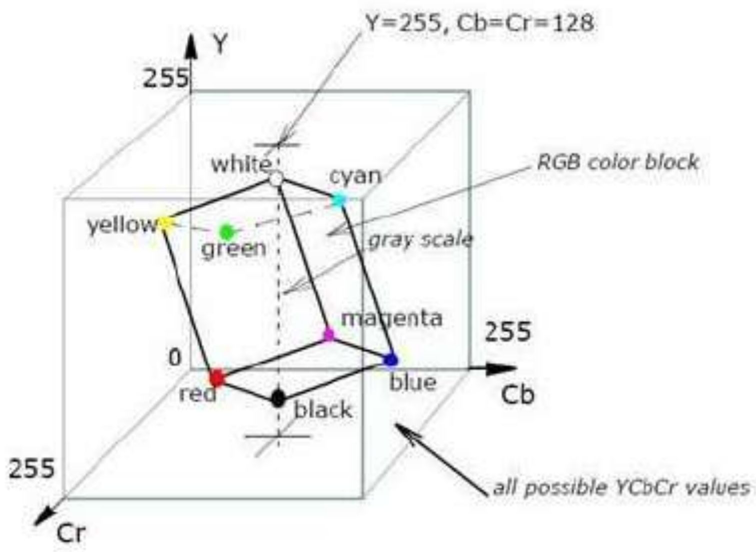

Figure 1 YCbCr Model

Shading is a significant element of traffic sign, and traffic sign can be immediately situated by shading division. Contrasted and RGB shading space and YCbCr shading space, the YCbCr shading space has a quicker recognition speed, less influenced by brightening, and has a best division advantage. Figure 1 shows the HSV shading space changed over from the RGB shading space. It speaks to the focuses in the RGB shading space by a reversed cone, where Y is the luminance, CbCr is the chrominance esteem. During the time spent limit division, the pixels inside the set edge go are set to white, else they are set to dark, and the picture is totally binarized.

Deep learning includes convolutional neural networks, which are widely used in image recognition. There are multiple layers in these convolutional neural networks. With the use of filters, a Conv2D layer is employed to extract features first. The number of filters is usually in the power of two, such as 32, 64, or 128. This layer employs an activation function. The activation function ReLU (Rectified Linear Unit) is commonly employed. The maximal ReLU function is defined as (0, x). The next layer is the max pooling layer, which is used to minimize the image's size. This is done in order to lower the amount of computing power necessary to process the image. The third layer is the dropout layer. This dropout layer is used to avoid overfitting and reduce the model's complexity. Some neurons are destroyed at random in this layer.

Feature learning phase refers to the combination of the first three levels. These three layers are applied several times in order to improve the training. The fourth layer is the flatten layer, which turns the 2-D input into a lengthy 1-D vector of features that can be fed into the neural network as a fully connected layer. The dense layer, which serves as an output layer, is the final layer. The number of nodes in the final layer is the same as the number of classes. The softmax activation function is used in the final dense layer. The Softmax function returns a probability value (between 0 and 1), allowing the model to forecast which class has the highest probability [3].

3. Flow Chart For Traffic Sign Recognition System

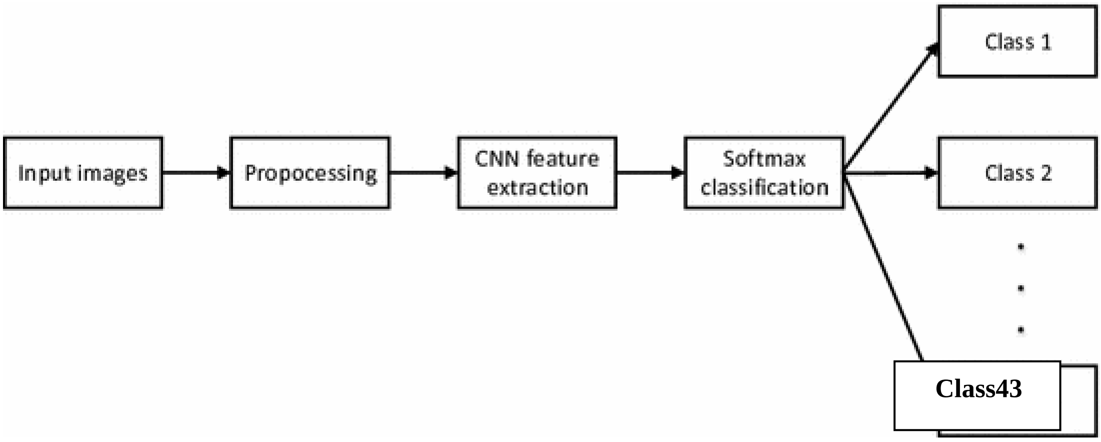

Figure 2 Flow graph for Traffic Sign Detection System

The flow graph shows the overall flow of the traffic signs recognition system using CNN network, starting with training the images from the test data set input those collected, trained images in the system. Next step is to preprocess the input images for noise reduction and to improve algorithm precision. The output from preprocessing stage is given to CNN feature extraction phase to extract the identified features and later on the images are given to the classification phase to identify class of image from a set of 44 different classes. Finally it detects the uploaded Traffic sign from the test dataset.

4. PREREQUISITES FOR SYSTEM IMPLEMENTATION

The system uses Deep learning Techniques and is implemented in Python with the version 3.7.9 and the required libraries and Frameworks to detect the traffic signs are installed. Python is a high-level object-oriented programming language with built-in dynamic semantics that is mostly used for web and app development. It has a lot of appeal in the field of Rapid Application Development since it allows for dynamic type and binding.

4.1 FEATURES OF DEEP LEARNING

- Reduce the requirement for feature extraction and automated jobs where predictions may be made in less time utilising Keras and Tensorflow to solve complex problems like audio processing in Amazon Echo, image recognition, and so on. Parallel computing can be done thus reducing overheads.
- Models can be trained on a large amount of data, and the model improves as more data is collected.
- When compared to humans, high-quality predictions can be made by training tirelessly.

4.2 MODULES/ Libraries USED

To implement the proposed system we installed the very familiar computer vision related libraries such as Open CV, PIL,matplotlib, Tensorflow etc.and some Deep learning frameworks Keras and Numpy which are described below.

4.2.1 NUMPY

The ancestor of NumPy, Numeric, was originally created by Jim Hugunin with contributions from several other developers. In 2005, Travis Oliphant created NumPy by incorporating features of the competing Numarray into Numeric, with extensive modifications. NumPy is open-source software and has many contributors.

4.2.2 TENSORFLOW

TensorFlow is a software library or framework, designed by the Google team to implement machine learning and deep learning concepts in the easiest manner. It combines the computational algebra of optimization techniques for easy calculation of many mathematical expressions.

4.2.3 KERAS

Keras is an open-source software library that provides a Python interface for artificial neural networks. Keras acts as an interface for the TensorFlow library.Up until version 2.3, Keras supported multiple backends, including TensorFlow, Microsoft Cognitive Toolkit, Theano, and PlaidML. As of version 2.4, only TensorFlow is supported. Designed to enable fast experimentation with deep neural networks, it focuses on being user-friendly, modular, and extensible.

4.2.4 PANDAS

Pandas are a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time.

4.2.5 MATPLOTLIB

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter.

Figure 3 Stop Traffic Sign

Figure 4 No Entry Traffic Sign

Figure 5 Speed Limit (30km/h) Traffic Sign

Figure 6 Go Straight or Left Traffic Sign

Figure 7 Dangerous Curve Left Traffic Sign

4.2.6 OPENCV

OpenCV is written in C++ and its primary interface is in C++, but it still retains a less comprehensive though extensive older C interface. All of the new developments and algorithms appearinthe C++interface.

4.3 PROCEDURE TO IMPLEMENT THE PROPOSED SYSTEM

Our approach to build this traffic sign classification model consists of the steps like Explore the dataset , Build a CNN model ,Train and validate the model and Test the model with test dataset. To detect the Traffic signs we first take an image as input and preprocess the image using preprocessing techniques to find out the noise in the image and reduce the nosie in the image if any. In the next step we use RGB based detection and shape based detection to detect the required traffic sign.

5. EXPERIMENTAL RESULTS

5.1 Detected Traffic Signs: The Following Traffic signs are detected in the implementation of proposed system

5.2 Probabilty of Detcting Traffic Signs: The following table depicts the detection probability for each Traffic Sign obtained by implementing the proposed system.

Table 1 from the data we have read, this is the result of each sign

Detected Traffic sign	Probability of Detection	Class number	Recognition Time(ms)
STOP	99.86	14	20 to 50
NO ENTRY	99.97	17	20 to 50
Speed Limit (30 kmph)	99.69	1	20 to 50
GO Straight or Left	99.99	37	20 to 80
Dangerous Curve to Left	99.62	19	20 to 80

5.3 Accuracy Obtained for various Techniques: The below table shows the percentage of accuracy derived when we tried with different algorithms. Here we used SVM, Naïve Bayes and CNN algorithms. Out of given techniques, CNN gives best results.

Table 2 Accuracy of different algorithms

Algorithm Used	Accuracy
SVM	83%

Naïve Bayes	85%
CNN	99%

5.4 Result Analysis: The following diagrams shows the graphical representation of accuracy and loss functions obtained from system while detecting the Traffic signs

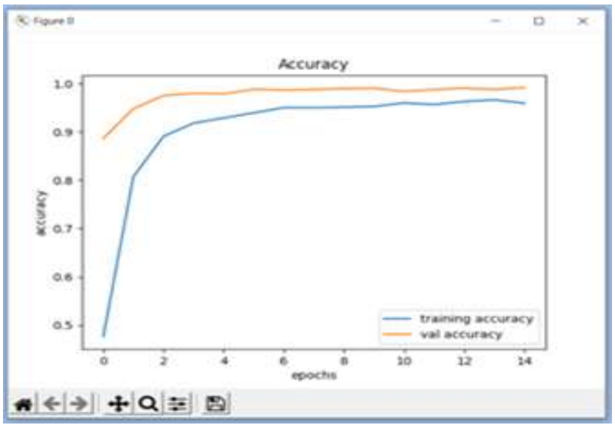

Figure 8 Accuracy in Detecting Traffic signs

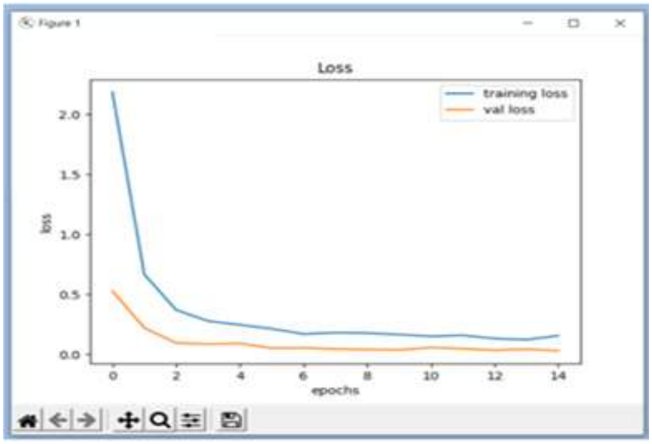

Figure 9 Loss in Detecting Traffic Signs

6. CONCLUSION

One of the most important responsibilities for intelligent vehicles, which are itself a future technology, is traffic sign recognition. Solving the TSR problem will have a significant impact on our driving safety, lowering the number of fatalities on the road. Computer-assisted traffic sign identification has a long history of academic and corporate research initiatives, and the first commercial implementations are already available. Although the model we provided will help us get closer to the ideal Advanced Driver Assistance System (Autonomous Car) or perhaps a completely driverless system, there is still a lot that can be done to improve it. This study is based on the colour and shape of a symbol in order to detect it. The CNN algorithm is prefered as it is more efficient over other algorithms.

REFERENCES

[1] Wali, S.B.; Abdullah, M.A.; Hannan, M.A.; Hussain, A.; Samad, S.A.; Ker, P.J.; Mansor, M.B. Vision-based traffic sign detection and recognition systems: Current trends and challenges. Sensors 2019, 19, 2093

[2] https://www.researchgate.net/publication/224255280_Realtimetraffic_sign_recognition_system

[3] Vaibhavi Golgire, 2021, Traffic Sign Recognition using Machine Learning: A Review, International National Journal Of Engineering Research & Technology (IJERT) Volume 10, Issue 05 (May 2021)
[4] https://medium.com/dataflair/class-data-science-project-for-2020-traffic-signs-recognition-12b09c131742
[5] https://ieeexplore.ieee.org/document/9036784
[6] Degui Xiao, Liang Liu, Super-resolution-based traffic prohibitory sign recognition ,2019.
[7] Zhang, Z.J.; Li, W.Q.; Zhang, D.; Zhang, W. A review on recognition of traffic signs. In Proceedings of the 2014 International Conference on E-Commerce, E-Business and E-Service (EEE), Hong Kong, China, 1–2 May 2014; pp. 139–144. International Journal of Future Generation Communication and Networking Vol. 13, No. 4, (2020), pp. 267–274 274 ISSN: 2233-7857
[8] Zhu, H.; Yuen, K.V.; Mihaylova, L.; Leung, H. Overview of environment perception for intelligent vehicles. IEEE Trans. Intell. Transp. Syst. 2017, 18, 2584–2601
[9] Shi, J.H.; Lin, H.Y. A vision system for traffic sign detection and recognition. In Proceedings of the 26th IEEE International Symposium on Industrial Electronics (ISIE), Edinburgh, UK, 18–2 June 2017; pp. 1596–1601
[10] I. Mato, Z. Krpi and K. Romi, "The Speed Limit Road Signs Recognition Using Hough Transformation and Multi-Class Svm," 2019 International Conference on Systems, Signals and Image Processing (IWSSIP), 2019, pp. 89-94.