International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 08 Issue: 06 | June 2021 www.irjet.net p-ISSN: 2395-0072

TRAFFIC SIGN RECOGNITION

Varshitha M S[1], Poornashree K A[2], A N Sunag Rao[3], Yashaswini P D[4], G Ramesh[5]

[1,2]U.G Student, Dept. of Computer Science and Engineering, The National Institute of Engineering, Mysuru, Karnataka, India.

Abstract - *A visual traffic sign recognition system can be integrated into the automobile with the objective of detecting and recognizing all emerging traffic signs. In case the driver refuses to heed traffic signs, the system will trigger an alarm. In this research, we aim to develop an efficient TSDR that can detect and categorize traffic signal into different classes in real-time with the help of Machine Learning techniques. The training process takes predefined traffic signals and has them 'learn' into a model. This in turn uses image processing totrain the model.*

Key Words **: Detecting and recognizing, TSDR, Machine Learning techniques, Image processing , model.**

1. INTRODUCTION

According to global road crash data, approximately 1.3 million people die in traffic accidents each year, averaging 3,287 deaths each day. Unfortunately, drunk driving, reckless driving, fatigue, and driver distraction continue to be the leading causes of road deaths. With today's traffic control technologies, there's a good chance the motorist will miss part of the traffic.

Fig 1: Samples from 8 of 10 traffic signs learnt by the model

An on-board computer vision system that can detect and identify traffic signs could help drivers avoid accidents in a variety of ways. The on-board vision technology might supplement reality by displaying forthcoming warning signs ahead of time, or even keeping them shown on a screen after the sign has past. This would make it less likely that the driver would miss an important sign. Traffic-sign recognition is a system that allows a vehicle to detect traffic signs placed on the road, such as "speed limit," "children," or "turn ahead." It detects traffic signs using image processing techniques.

1.1 Existing

Important information about road limitations and conditions is conveyed to drivers via visual signals, such as traffic signs and traffic lanes, in all countries throughout the world. Traffic signs are an important aspect of road infrastructure since they convey information about the current condition of the road, as well as limitations, bans, cautions, and other navigational aids. The visual features of traffic signs encode this information: shape, color and pictogram. Disregarding or failing to see these traffic signs may contribute to a traffic accident, either directly or indirectly. In bad traffic, however, the driver may fail to observe traffic signs on purpose or by accident. If a driver's inattention is compensated for by an automatic detection and recognition system for traffic signs, the system can reduce a driver's fatigue by assisting him in following the traffic sign, making driving safer and easier.

1.2 Proposed

With the introduction of augmented reality technologies in some current automobiles, the field of traffic sign identification has gotten a lot of attention. The system is being developed in three stages: picture preprocessing, detection, and recognition. The traffic sign is captured by the camera and then preprocessed, which includes scaling, rotation, grayscale conversion, and normalization, before feature extraction.

This information is then given to CNN, which predicts the proper traffic sign. This output is used to send a text message and a voice message to the driver.

2. METHODOLOGY

2.1 Neural Network Definition

The term neuronal network refers to a network or circuit of neurons, or in a modern sense, an artificial neural network, comprising artificial neurons or nodes for solving artificial intelligence problems. So a neuronal network is composed either of biological neurons, or artificial neurons. In a neural net, a computer learns to perform a task by analyzing examples of training data, with the examples usually having been labeled in advance. For example, an object recognition system may be fed thousands of images labeled with different traits, such as cars, houses, coffee cups, etc., and it can determine patterns in the images that are consistently related to particular attributes.

The PyTorch deep learning framework provides the greatest flexibility and speed while building and implementing Neural Networks.

2.2 Neural Network Model

Creating a Machine Learning algorithm entails creating a model that produces correct information when we offer it with accurate input data. The process through which the model learns to make sense of the incoming data is known as training, and it is an important concept in Machine Learning. After we've trained our model, we can simply feed it data to get a result. The basic logic of training an algorithm involves four ingredients

- **Data**
- **Model**
- **Objective Function**
- **Optimization Algorithm**

To begin, we must first create a set of data to train with. This is usually historical data that is easily available. With the data we've gathered, we can use a simple model like a linear model or a pre-trained model like a Neural Network.

So far, we've taken data, fed it into the model, and gotten a result.

However, we want this output to be as accurate as possible. The objective function is used to solve this problem. It calculates the accuracy of the model's outputs. The entire Machine Learning framework is based on this function being optimized. Finally, optimization algorithms are the mechanisms by which we change the model's parameters to optimize the objective function. As we know, the Machine Learning process is iterative, therefore we feed data into the model and compare accuracy using the objective function. Then we alter the model's parameters and repeat the process. When we no longer can or don't need to optimize, we've reached the end. Upon reaching a satisfactory conclusion, we would be able to resolve our problem.

In neural nets thousands or even millions of basic processing nodes are tightly interconnected, with a loose resemblance to the brain. In most cases, neural networks are feed-forward, which means that data flows through them only in one direction.

A node is in a layer may be connected to several nodes below it, from which it receives data, and to several nodes above it, from which it sends data. Nodes receive information over each of their connections and multiply it by the associated weight each time the system is operational. Once the results are added, one number is generated. In any case, no data is sent to the next layer if the number is less than a threshold. Neural networks send the sum of weighted inputs through all their connections when they reach a threshold value, fires.

Initialization of a neural net's weights and thresholds takes place at random during training. Data is fed into the input layer at the bottom of the structure, and it passes through the following layers, getting multiplied and added together by an activation function. A function called activation converts a total value calculated beforehand into a number between 0 and 1. To activate the neurons from an entire network, multiply each value by its respective coefficient, then add these values together. As soon as a neuron sends its new value to all the other neurons in a column, the neural network moves on to the next column. Lastly, we should obtain a means of determining the desired output. In Back Propagation, the weights and thresholds are continually adjusted until training data with similar labels consistently yield similar outputs; the Gradient Descent algorithm is used in this process.

2.3 OpenCV

An opensource library called OpenCV can be used for machine learning, image processing, and computer vision. Computer vision systems require this for real-time operation.

The video captured using the device camera is captured on the fly using OpenCV. After the video has been captured, the frames of that video are separated. By using OpenCV, color images are made into grayscale images, which can then be transformed into arrays of numbers. The constituent frames are then fed into an object detector, allowing them to be detected as objects. When a set of signs is detected, the bounding box surrounding the object is displayed with its class name.

2.4 System Implementation

Training takes place after a data set has been created and preprocessed with rotation, resizing, grayscale conversion, and normalization.

As the number of epochs is set, the model will be trained so that as the accuracy grows, the training loss will decrease. Having done this, we have completed the evaluation process. After the training is complete, we have a Neural Network model. The resultant Neural Network model was used to further recognize traffic signs in live streams. Using OpenCV libraries, we could detect any sign put in front of the camera by simply displaying a green box bound to the sign along with the traffic sign. We tested the model to ensure that it produces accurate results. It is then repeated with new data if necessary.

Not only does the model accurately recognize traffic signs but it also provides voice output to alert the driver.

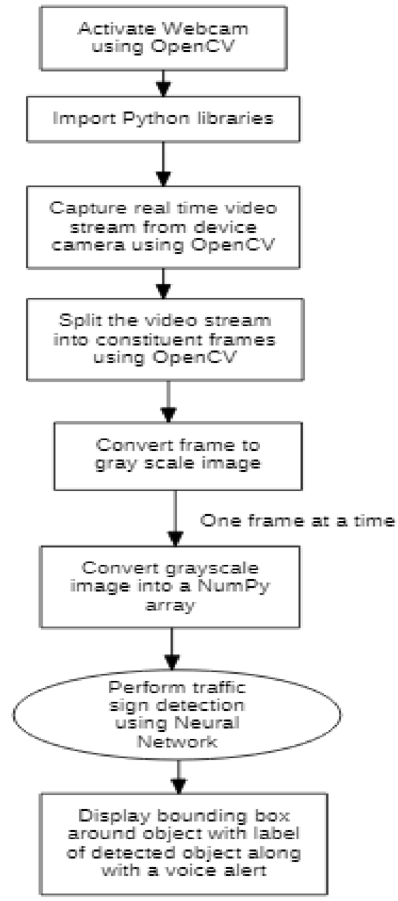

Fig 2: Steps followed in Traffic Sign Recognition

2.5 Results

Traffic sign recognition system recognizes those signs placed in front of the camera by bounding it with a green box and displaying the name of the sign above it. It also provides voice output of each recognized sign.

Test Case ID	TCI02
Test Case Name	Integration testing of Traffic Sign Recognition model
Description of Test Case	Check and test the end to end integration of the traffic sign recognition Model using Neural Network and TensorFlow to ensure that the required traffic sign recognition functionality is working properly and required output is being produced.
Feature being tested	Real time recognition of the Traffic Sign
Sample Input	Live video stream capturing the traffic signs displayed in the mobile
Expected Output	Recognise the traffic sign in the real time video and obtain audio output of the sign
Obtained Output	Obtained the name of the recognised traffic sign in text as well as audio format
Status	Success

Table-1: Integration testing for Traffic sign recognition model

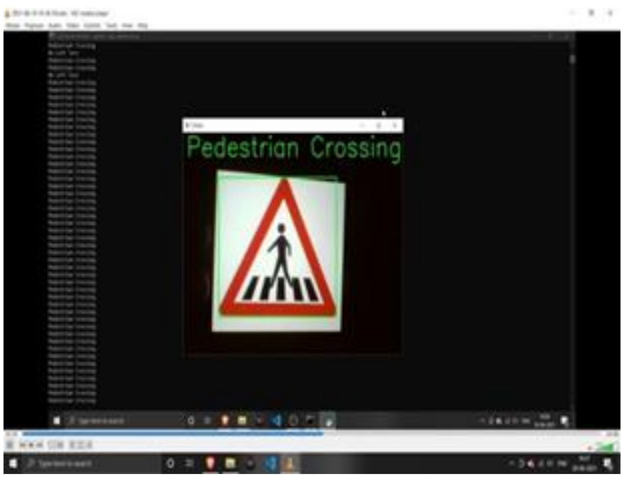

Fig 4: Pedestrian Crossing traffic sign being recognized

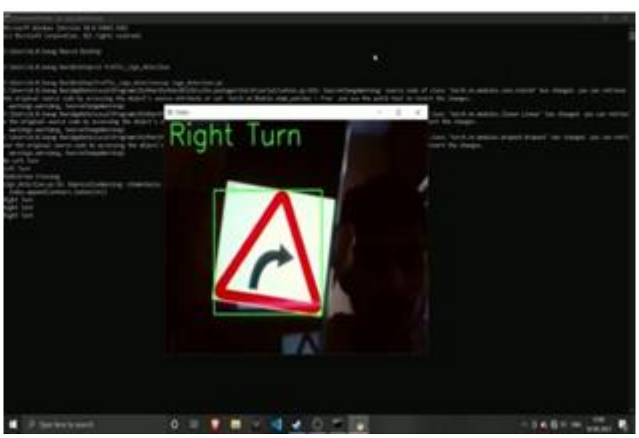

Fig 5: Right turn traffic sign being recognize

Fig 6: Left turn traffic sign being recognized

3. CONCLUSIONS

A neural network can learn independently as well as generate results that aren't limited to the input. Since it stores input in its own network, rather than a database, data loss has no effect on its operation. Instead of using a pre-defined neural network model, we created our own. Our model was more efficient because of the torch library instead of any other library. Using the torch technique as we are detecting live streams, the recognition process is faster and easier. We have used the neural network for traffic sign recognition due to its state-of-the-art accuracy. Therefore, we are combining computer vision and deep learning to develop a real time traffic sign recognition system. In this model, a voice alert signals the driver when the sign is detected. With this system, the driver never misses a traffic signal because he receives an alert before crossing the sign. Thus, the driver is more likely to drive on the road safely. Furthermore, it allows the driver to stay within the traffic laws.

ACKNOWLEDGEMENT

We would like to take this opportunity to express our profound gratitude to all those people who are directly/indirectly involved in the completion of this paper. We would like to thank **The National Institute of Engineering, Mysore**[1] for providing us this opportunity and resources towards the successful completion of the paper.

It also gives us immense pleasure to thank our Project Guide **Mr. Ramesh G**[2] and Project Co-Guide Mr. M. J Yogesh[3], for their constant motivation and support during the entire process.

REFERENCES

[1] Traffic Sign Recognition for Computer Vision Project-Based Learning David Gerónimo, Joan Serrat, Antonio M. López, Member, IEEE, and Ramon Baldrich, August 2019

[2] Road traffic sign detection and classification, A. De la Estalera, L.E.Moreno M.A.Salichs, J.M.Armingol, IEEE ,December 1997.

[3] An Automatic Traffic Sign Detection and Recognition System Based on Color Segmentation, Shape Matching, and SVM - Research Article,Safat B. Wali, Mahammad A. Hannan, Aini Hussain, and Salina A. Samad, November 2015.

[4] R. Szeliski, Computer Vision: Algorithms and Applications. New York, NY, USA: Springer, 2010.

[5] G. Bebis, D. Egbert, and M. Shah, "Review of computer vision education," IEEE Trans. Educ., vol. 46, no. 1, pp. 2–21, Feb. 2003.

[6] P. Reimer, A. Albu, and G. Tzanetakis, "Raydiance: A tangible interface for teaching computer vision," in Proc. 7th ISVC, Las Vegas, NV, USA, 2011, vol. 2, pp. 259–269.