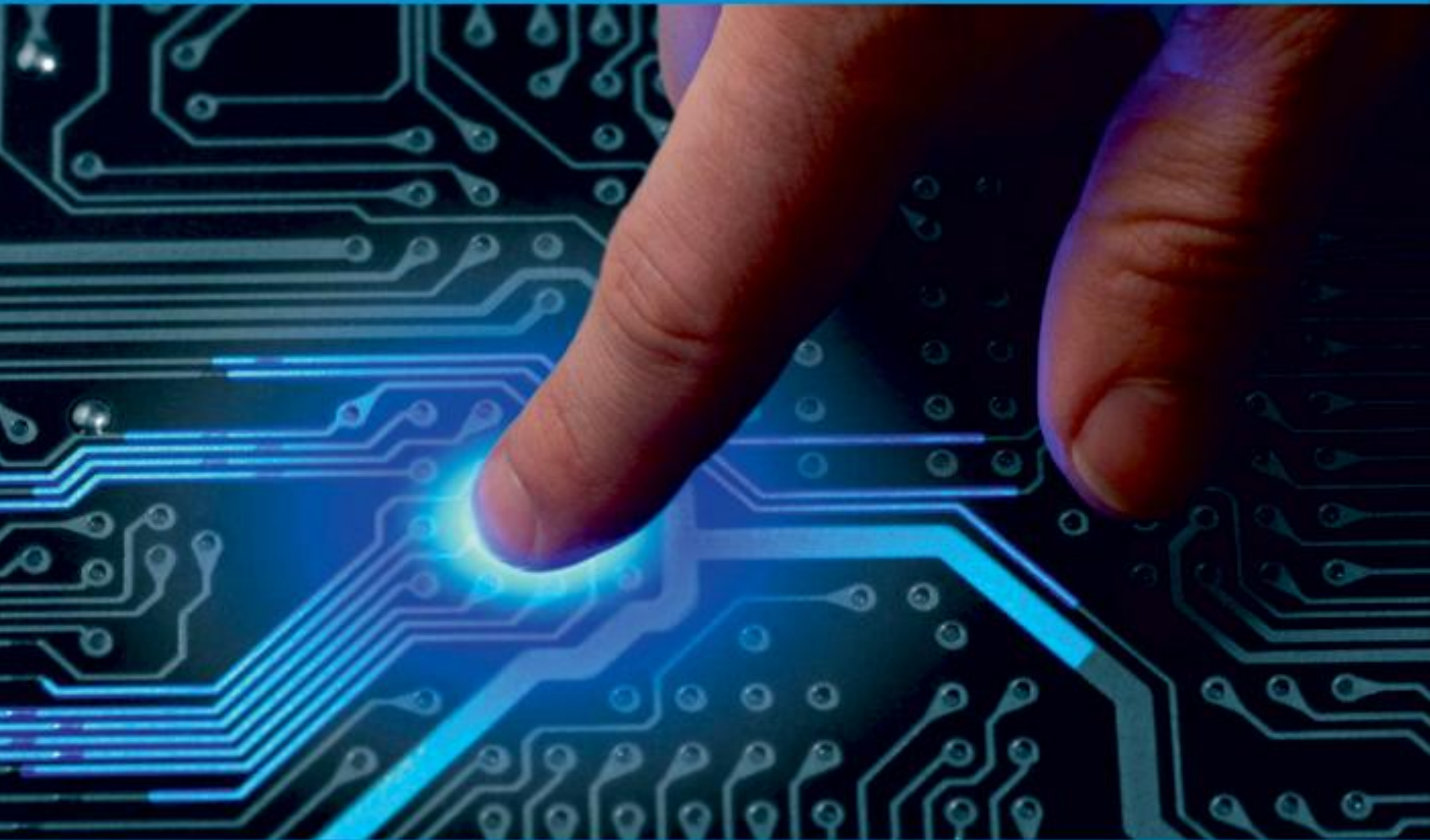
IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 9, Issue 7, July 2021

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 7.542

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 7, July 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0907192 |

Application for Real Time Traffic Sign Recognition on Indian Road

Adigouni Saketh, Bhavya N M, Chittipala Sirisha, Kuwant Raj Kisku, P Kokila

B.E Final Year, Dept. of ISE, The Oxford College of Engineering, Bengaluru, India

B.E Final Year, Dept. of ISE, The Oxford College of Engineering, Bengaluru, India

B.E Final Year, Dept. of ISE, The Oxford College of Engineering, Bengaluru, India

B.E Final Year, Dept. of ISE, The Oxford College of Engineering, Bengaluru, India

Asst. Professor, Dept. of ISE, The Oxford College of Engineering, Bengaluru, India

ABSTRACT: Vehicles have the potential to revolutionize urban mobility by providing sustainable, safe, and convenient and congestion free transportability. Development of safety features to prevent ignorance of traffic sign boards mounted on roads is one of the major technical challenges for cars. Ignoring traffic signs can lead to major road accidents. Therefore, assisting vehicles with warning and information of road signs can be of significant help in prevention of accidents. To overcome these problems the following methodology is adopted. An Android Application is implemented for traffic sign board detection and recognition. The input is taken in real time through the android application using a mobile phone camera which will then be preprocessed and fed into the Convolutional Neural Network (CNN) model. The model then outputs the classified traffic sign board. The application will then alert the system through voice command and a marker is plotted on the Google maps for reusability. The Indian sign boards are very prone to changing conditions and hence this project tries to adapt traffic sign detection systems for Indian sign boards. The accuracy of detection of Indian sign boards in vehicles is improved and can be adaptable to Indian road conditions and illuminations.

KEYWORDS: Android Application, Convolutional Neural Network, Traffic Sign Board Detection and Recognition, Voice Command.

I. INTRODUCTION

An autonomous car is a vehicle that is capable of sensing its environment and navigating without human input. Autonomous cars combine a variety of techniques to perceive their surroundings, including radar, laser-light, GPS (Global Positioning System), odometer, and computer vision. Advanced control systems interpret sensory information to identify appropriate navigation paths, as well as obstacles and relevant signage. Road signs give out a number of messages regarding the road and what a driver expects on the road. They keep the traffic flowing freely by helping drivers reach their destinations and letting them know entry, exit and turn points in advance. Pre-informed drivers will naturally avoid committing mistakes or take abrupt turns causing bottlenecks. Road signs, indicating turns, directions and landmarks, also help to save time and fuel by providing information on the route to be taken to reach a particular destination. Road signs are placed in specific areas to ensure the safety of drivers. These markers let drivers know how fast to drive. They also tell drivers when and where to turn or not to turn. In order to be a terrific driver, one needs to have an understanding of what the sign means. Indian sign boards are easy and comprehend whose size is approximately 60cm, clearly visible from a distance of around 100 m. Most of the commonly available sign boards are in red color, other sign boards are in blue color.

Development of safety features so as to prevent ignorance of traffic sign boards mounted on roads is one of the major technical challenges in the automobile industry. Ignoring traffic signs can lead to major road accidents. Therefore, using a driver assistance system to timely assist with warning and information road signs can be of significant help in prevention of accidents.

This project gives insight into mathematics and procedure involved in the classification of traffic sign boards and understanding of the critical aspects of the applying Convolution Neural Network and developing and developing an android application for the real time recognition of the sign boards. In machine learning, convolution neural networks are a class of deep, feed forward artificial neural networks that has been successfully applied to analyzing visual imagery. CNN s use little pre-processing compared to other image classification algorithms. This means that the

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 7, July 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0907192 |

network learns the filters that in traditional algorithms were hand engineered. This independence from prior knowledge and human effort in feature design is a major advantage. Convolution networks were inspired by biological processes in that the connectivity pattern between the neurons resembles the organization of animal visual cortex. Individual cortical neurons respond to stimuli only in the restricted region of the visual field known as receptive field. The receptive fields of different neurons partially overlap such that they cover the entire visual field.

II. RELATED WORK

In [1] Traffic Sign Detection and Recognition on German Traffic Sign Benchmarks along with Belgium datasets using image processing for the detection of a sign and an ensemble of Convolutional Neural Networks (CNN) for the recognition of the sign. The overall train set for triangles had 10582 images and test set had 3456 images. The train set for circles had 16106 images and the test set had5277 images. Upon training the CNN ensemble, an accuracy of 98.11% for triangles and 99.18% for circles was achieved. For identification of a sign, this paper depends on the color and shape of the sign. Advantage: Gives 98.11% accuracy and recognition is based color and shape of the sign. Disadvantage: Reflection on the sign which impacts its color ans also if the sign is chipped or cut off, the shape of the sign is impaired, thus resulting in no detection of the sign. In [2] Multiple Real-time object identification on a customized data set which is similar to the COCO data set using Single Shot Multibox Detection (SSD).This work eliminates the feature resampling stage and combines all calculated results as a single component.For detecting one single object it takes nearly 3s. Advantage: Using Single Shot Multibox Detection (SSD),eliminates the feature resampling stage and combines all calculated results as a single component. Disadvantage: objects need to be at the distance of 30 meters to the web camera. If the object is away from that distance, the detection rate is decreased. In [3] Traffic Light Detection and Recognition for self driving cars on a customised dataset collected in the urban streets and suburban roads of Pune, Maharashtra, India using deep neural network and faster region based convolutional network (R-CNN) Inception V2 model.The collection of images and video sequences is done by using a 16 megapixel camera which supports image resolution of 4616 × 3464 pixels and video recording resolution of 1920 × 1080 pixels at a frame rate of 30 fps. In total, 1237 frames have been taken in the daylight conditions. Advantage: The system can also be optimized for safe driving despite unclear lane markings. Also, it can be equipped with the ability to respond to spoken commands highway safety employees. Disadvantage:After training the model for 120,000 iterations which took nearly 12 hours it reported a loss in the range of 0.01 which is better compared to specified acceptable loss which is in the range of less than 0.05.In [4] Traffic sign detection for advanced driver assistance system on a customized data set using contour analysis approach. In this paper blue, green, red (BGR) to hue, saturation, value (HSV) conversion model and morphological filter for noise filtering are used to make the result more robust. The system has a voice trigger to alert the driver of road signs via audio message and thus helping in prevention of accidents. Advantage: results are 80 percent appropriate even with 90 degrees of rotation in sign boards. Disadvantage: Contour analysis fails at identifying boards with separate contours as with the case with narrow bridge which is a major drawback. In [5] Traffic sign detection method on the experimental datasets that are collected from the fields using Single Shot Multibox detector (SSD) detection algorithm.The scale of the entire data set was 1532, of which 1314 were training set images and 218 were test sets.The total number of traffic signs included 2741, including speed limit 20km/h, no horn, keeping right, no left turn, no motor vehicles, no parking. The mean Average Precision (mAP) is 0.7528, and has raised about 10% compared to the classical SSD, which verifies the effectiveness of the proposed algorithm. Advantage: The experiment verifies that improved SSD detection accuracy and positioning are significantly improved, and robustness is also enhanced. Disadvantage: The scale of the collected data set needs to be further increased to detect more traffic signs. And there is no visualize the detection process of the SSD model, to explain to a certain extent the working principle of the deep learning network. In [6] Real-time Large Scale Traffic Sign Detection using Novel YOLOv3 architecture and Convolutional Neural Network on a data set that was collected by private company for the purpose of maintaining inventory of traffic signs. A large number of images were captured from the RGB camera mounted on the vehicle in both urban and rural areas. Images that contain at least one traffic sign instance were kept in database and carefully annotated with bounding boxes. Annotations are done only for planar traffic signs.5000 pictures with more than 13k annotations, distributed in around 200 class was included in the database. The results showed that real-time TSD/TSR is possible even for the HD images, with a mean average precision of 88%. Advantage: traffic signs instances with compact aspect ratios are much easier to accurately detect and recognizeDisadvantage: Annotations are done only for planar traffic signs. Furthermore, signs smaller than 30 pixels arediscarded as well as classes with less than 20 samples. In [7]smart driver alert system for vehicle traffic on a customised dataset containing different sign board images using background subtraction,hough transform,morphological operations and template matching.The frame which consists of the traffic sign board is detected and processed whereas other frames without the traffic sign boards are discarded. The frame is converted to binary image and morphological operations are performed. Using Background subtraction foreground is extracted from the frame, the frame consists of objects, the object with red pixels is detected as red

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 7, July 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0907192 |

objects. Using Hough Advantage: The alert message, as voice output to the driver is given to avoid further fatalities or catastrophes.The proposed approach can be used in automatic vehicles, all kinds of vehicles to automate the process and to reduce road accidents. Disadvantage: Transform the objects are classified based on shape and compared with the standard data set in the database using template matching.The comparison of two images is done based on correlation coefficient technique.

III. METHODOLOGY

A. **Vehicle mounted video camera and input:**The input is taken in real time using the mobile phone camera through the android application that is developed. Android Studio can be used to take input through the camera api functions and process them at frame by frame.

B. **Input video conversion to frames:** Android Studio can be used to process input at frame by frame through the camera api functions. The ImageReader class allows direct application access to image data rendered into a Surface. Several Android media API classes accept Surface objects as targets to render to, including MediaPlayer, MediaCodec, CameraDevice, ImageWriter and RenderScript Allocations. The image sizes and formats that can be used with each source vary, and should be checked in the documentation for the specific API. The image data is encapsulated in Image objects, and multiple such objects can be accessed at the same time, up to the number specified by the maxImages constructor parameter. New images sent to an ImageReader through its Surface are queued until accessed through the acquireLatestImage() or acquireNextImage() call.

C. **Colour based segmentation:** We use HSV for colour based segmentation which defines colour space.HSV colour space has 3 components: Hue, Saturation,Value.The main intention behind HSV is to eliminate background part and extracts only sign from image.

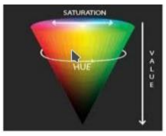


FIG 1 : HSV

D. **Image Augmentation:**Image augmentation artificially expand training images through different ways of processing or combination of multiple processing, such as random rotation, shifts, shear and flips, etcTo get the customized training and testing set, image augmentation was done. The images underwent multiple processing such as rotations, shifts, flips etc. This proliferated the dataset i.e.; one image got proliferated to 100 images as shown in the following Figure 6.6. Image augmentation parameters that are generally used to increase the data sample count are zoom, shear, rotation, preprocessing function and so on. Usage of these parameters results in generation of images having these attributes during training of Deep Learning model.

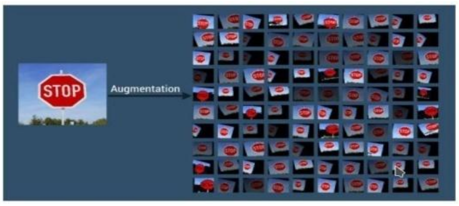


FIG 2 : IMAGE AUGUMENTATION

Image samples generated using image augmentation, in general results in increase of existing data sample set by nearly 3x to 4x times.

E. **Build CNN:** In machine learning, convolutional neural network is a class of deep, feed forward artificial neural networks that has been successfully applied to analyze visual imagery.Convolutional networks were inspired by biological processes in that the connectivity pattern between the neurons resembles the organization of animal visual cortex.convolutional and sub-sampling layers. Together the layers produce an approximation of input image data.The dataset from image augmentation and the collected dataset are fed to CNN as training dataset and testing dataset respectively.In CNN, Training shall be done by passing the input along with the expected result through which the model can compare and learn and classify properly.

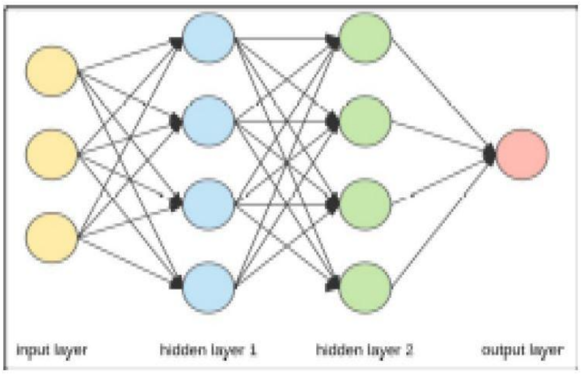


FIG 3 : Convolutional Neural

The comparison is done by giving priority to weights which is a communication link that contains input signal. Learning occurs through these weights and aims to minimize the errors between input and output layers.
The weight can be represented in terms of matrix. The weight matrix is also called a connection matrix. The weight matrix W is defined by

$$\mathbf{W} = \begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1n} \\ w_{21} & w_{22} & \cdots & w_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ w_{n1} & w_{n2} & \cdots & w_{nn} \end{bmatrix}$$

Fig 4: weight matrix

F. **Train CNN:** Backpropagation is used to train the CNN of chain rule method.In simple terms, after each feed forward passes through a network this algorithm does the backward pass to adjust the model parameters based weights and biases.CNN Model is trained a no of times with same training set. Each time model sees the entire training dataset in training process.After the training of CNN one classified image obtained by the output layer.CNN model is trained using the back propagation supervised learning algorithm. The model is trained a number of times with the same training set. Each time the model sees the entire training dataset in the training process is called an Epoch. One epoch is equal to one forward pass and one backward pass of all the training examples. The model is trained using the batch processing method where it does not take one patch of data at a time rather it sends data in batches of size 100 or 200 and so on.

G. **Convert model to tensorflow lite model:** TensorFlow Lite is designed to execute models efficiently on mobile and other embedded devices with limited compute and memory resources. Some of this efficiency comes from the use of a special format for storing models. TensorFlow models must be converted into this format before they can be used by TensorFlow Lite.Converting models reduces their file size and introduces optimizations that do not affect accuracy. The TensorFlowIn general, CNN has 3 layers

1.Convolutional layer: Input layer which takes image as input.

2.Fully Connected layer: Output layer from where we get trained output.

3.Polled layer: Intermediate layer, the network has a series of

reduce file size and increase speed of execution, with some trade-offs. The TensorFlow Lite converter is a tool available as a Python API that converts trained TensorFlow models into the TensorFlow Lite format. The following figure shows the TensorFlowLite Converter.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 7, July 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0907192 |

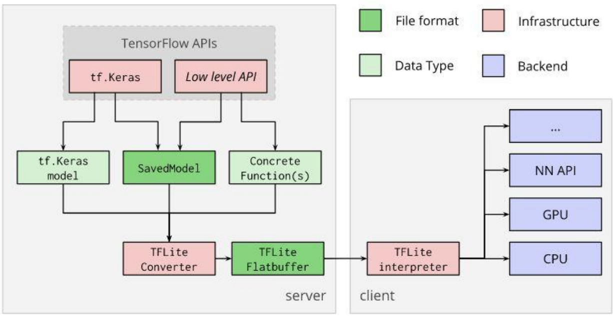


FIG 5: TENSORFLOW LITE CONVERTER

H. Run Inference

The term inference refers to the process of executing a TensorFlow Lite model on-device inorder to make predictions based on input data. To perform an inference with a TensorFlow Lite model, you must run it through an interpreter. The TensorFlow Lite interpreter is designed to be lean and fast. The interpreter uses a static graph ordering and a custom (less-dynamic) memory allocator to ensure minimal load, initialization, and execution latency.

This describes how to access to the TensorFlow Lite interpreter and perform an inference using C++, Java, and Python, plus links to other resources for each supported platform.

TensorFlow Lite inference typically follows the following steps:

Loading a model:You must load the .tflite model into memory, which contains the model's execution graph.

Transforming data:Raw input data for the model generally does not match the input data format expected by the model. For example, you might need to resize an image or change the image format to be compatible with the model.

Running inference:This step involves using the TensorFlow Lite API to execute the model. It involves a few steps such as building the interpreter, and allocating tensors, as described in the following sections.

Interpreting output:When you receive results from the model inference, you must interpret the tensors in a meaningful way that's useful in your application.

For example, a model might return only a list of probabilities. It's up to you to map the probabilities to relevant categories and present it to your end-user.

I. Voice alert system

Voice actions are an important part of the wearable experience. They let users carry out actions hands-free and quickly. A voice alert system is thus added in the android application developed in this project to give the output that is the detected sign board in voice format in order to alert the driver and thus the driver or system can take appropriate actions.

Android allows me to convert the text into voice. Android allows to convert the text into voice. Not only converting it but it also allows me to speak text in a variety of different languages. Thus in this project the text that is the label obtained from the CNN model to be converted into voice to give the result.

Android provides a TextToSpeech class for this purpose. Inorder to use this class, you need to instantiate an object of this class and also specify the initListener. Its syntax is given below

```
private EditText write;
tt obj=new TextToSpeech(getApplicationContext(), new TextToSpeech.OnInitListener() {
@Override
public void onInit(int status) {
Lite converter provides options that allow you to further
}
});
```

Thus this TextToSpeech function of android studio helps to give the result in voice format
and alert the system

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 7, July 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0907192 |

IV. SIMULATION RESULTS

The images shown represent the application of the algorithm with 70 epochs

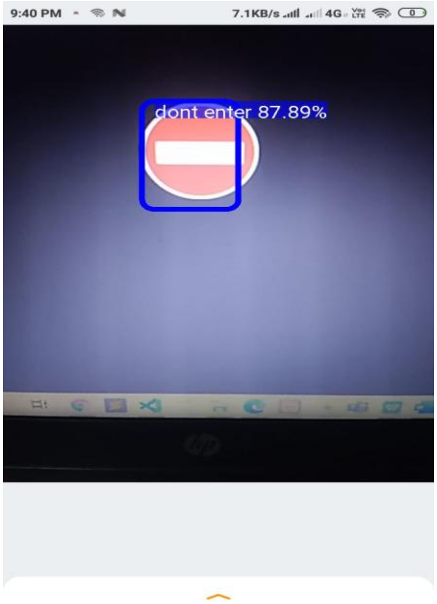


Fig 6 : Detection of Don't Entry Sign Board

Fig 7 : Detection of cross walk Sign Board

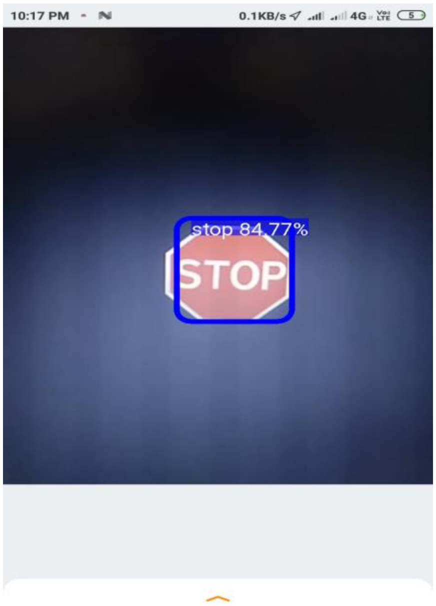


Fig 8 : Detection of stop Sign Board

V. CONCLUSION AND FUTURE WORK

With the increasing number of vehicles manufactured every year there's been a compounding interest for the safety necessity of vehicles and thus in this project an android application for detection and recognition of traffic sign boards on Indian roads is developed. In this project, the input is taken in real time through the android application using a

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 7, July 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0907192 |

mobile camera and the app gives the result in voice format. The focus of the project was mainly on the recent trend of the use of Convolution Neural Networks method for image recognition and developing an android application for the same. The first stage was to integrate an existing model into the android application by converting the model into tensorflow lite model. Further a new model was developed using SSD architecture and integrated with an android application to give the results in voice format in real time. The model is trained using Stochastic Gradient Descent (SGD) optimizer with initial learning rate 0.001 , batch size 32, no. of epochs as 70 SSD achieves mean accuracy precision (mAP 74.3%).

VI.FUTURE WORK

Scope of future work includes:
- This project's main aim is to get integrated with autonomous vehicles systems.
- The project needs to encapsulate all the traffic sign boards of Indian roads.
- The project can be integrated with google maps to plot the traffic sign boards location

REFERENCES

[1] 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.
[2] S. Kanimozhi, G. Gayathri and T. Mala, "Multiple Real-time object identification using Single shot Multi-Box detection," 2019 International Conference on Computational Intelligence in Data Science (ICCIDS), Chennai, India, 2019, pp. 1-5.
[3]R. Kulkarni, S. Dhavalikar and S. Bangar, "Traffic Light Detection and Recognition for Self Driving Cars Using Deep Learning," 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), Pune, India, 2018, pp. 1-4.
[4] P. S. K. Pandey and R. Kulkarni, "Traffic Sign Detection for Advanced Driver Assistance System," 2018 International Conference On Advances in Communication and Computing Technology (ICACCT), Sangamner, 2018, pp. 182-185.
[5] Z. Dongtao, C. Jie, Y. Xing, S. Hui and S. Liangliang, "Traffic Sign Detection Method of Improved SSD Based on Deep Learning," 2018 IEEE 4th International Conference on Computer and Communications (ICCC), Chengdu, China, 2018, pp. 1516-1520.
[6] A. Avramović, D. Tabernik and D. Skočaj, "Real-time Large Scale Traffic Sign Detection," 2018 14th Symposium on Neural Networks and Applications (NEUREL), Belgrade, 2018, pp. 1-4.
[7] S. Harini, V. Abhiram, R. Hegde, B. D. D. Samarth, S. A. Shreyas and K. H. Gowranga, "A smart driver alert system for vehicle traffic using image detection and recognition technique," 2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), Bangalore, 2017, pp. 1540-1543.
[8] D. M. Filatov, K. V. Ignatiev and E. V. Serykh, "Neural network system of traffic signs recognition," 2017 XX IEEE International Conference on Soft Computing and Measurements (SCM), St. Petersburg, 2017, pp. 422-423.