Traffic Sign Board Detection and Recognition using Augmented Reality

Akshata Anant Prabhu [1], Deepika V.D. [1], Muralikrishna .N[1], P. Vaishnavi Acharya[1], A.R.Manjula[2]

[1]*Students, Dept. of Telecommunication Engineering, Bangalore Institute of Technology, Bangalore, Karnataka, India*
[2]*Professor, Dept. of Telecommunication Engineering, Bangalore Institute of Technology, Bangalore, Karnataka, India*

---***---

Abstract- *Driving is a complex, continuous, and multitask process that involves driver's cognition, perception, and motor movements. The way road-traffic signs and vehicle information is displayed impacts strongly driver's attention with increased mental workload leading to safety concerns. Drivers must keep their eyes on the road, but can always use some assistance in maintaining their awareness and directing their attention to potential emerging hazards. Research in perceptual and human factors assessment is needed for relevant and correct display of this information for maximal road traffic safety as well as optimal driver comfort. In-vehicle contextual Augmented Reality (AR) has the potential to provide novel visual feedbacks to drivers for an enhanced driving experience. Here we present a new real-time approach for fast and accurate framework for traffic sign recognition [1]. Which superimposes virtual objects onto a real scene under all types of driving situations, including unfavorable weather conditions and gives a voice alert with the help of speakers. Show that the joint learning greatly enhances the capability of detection and still retains its real time performance*.

Key Words: **Augmented Reality (AR), traffic Sign Recognition, Voice Alert, Real time Approach, Display**

1. INTRODUCTION

Automotive active safety systems have become increasingly common in road vehicles since they provide an opportunity to significantly reduce traffic fatalities by active vehicle control. Traffic signs play a vital role in safe driving and in avoiding accidents by informing the driver of speed limits or possible hazards. The visibility of traffic signs is crucial for the driver's safety. In some cases, it may be very difficult to recognize traffic signs timely and accurately because the visibility may be reduced due to some environmental factors. Guiding the driver attention to an imminent danger, somewhere around the car, is a potential application [2]. After recognizing the traffic signs, a driver may be notified of the recognized traffic signs in a manner of audio or visual information. Traffic signs or road signs are an important part of the road environment as they provide visual messages not only for drivers; but for all road users. Traffic Sign Recognition (TSR) includes traffic sign detection and classification. Several detection algorithms are based on edge detection, making them more robust to changes in illumination [3]. Furthermore, several approaches and techniques for road sign detection and recognition have been introduced. The most common approach consists of two main stages: detection and recognition.

Recently the machine learning concepts is growing at a fast rate for the development of various models wherein the machine learns and predicts the future values or outcomes based on the initial values. Machine learning is learning based on experience. As an example, it is like a person who learns to play chess through observation as others play. In this way, computers can be programmed through the provision of information which they are trained, acquiring the ability to identify elements or their characteristics with high probability. The machine learning concept is used to extract the range of intensity values [4]. The linear regression model that comes under the supervised learning of machine learning concept is used to extract the range of intensity values and using these values the identification of the traffic sign is achieved.

2. LITERATURE REVIEW

This section discusses the various researches undertaken to identify the traffic signs, here we have mentioned few of the research work. In this literature review we discussed on traffic sign detection and recognition, deep learning concepts of convolution neural network and Haar- cascade algorithms for detection of traffic signs [5].

Machine learning classification methods which are a part of deep learning. Machine Learning is divided into 2 segments: Classification and Regression, out of which concepts like Haar Cascade [6], CNN (Convolutional Neural Network) were a part of Classification techniques. Our project implements Regression model in machine learning for Sign board detection and identification [7]. In Regression Model, there are few types like statistical regression model, logical regression model and linear regression model, out of which we are using Linear Regression Model [8]. According to the ease and efficiency of the algorithm, Regression model was simpler compared to the complex Classification Models.

3. MATERIALS AND METHODS

3.1 MATERIALS

3.1.1 Image Processing

Image processing is a method of performing some operations in an image to obtain an improved image or extract useful information from it. It is a type of signal processing in which input is an image and output may be image or characteristics / features associated with that image. Nowadays, image processing is among rapidly growing technologies. It forms core research area within engineering and computer science disciplines too [9].

As we know, images are represented in rows and columns we have the following syntax in which images are represented

$$f(x,y) = \begin{pmatrix} f(0,0) & f(0,1) & f(0,2) & \ldots & f(0,N-1) \\ f(1,0) & f(1,1) & f(1,2) & \ldots & f(1,N-1) \\ . & . & . & & . \\ . & . & . & & . \\ f(M-1,0) & f(M-1,1) & f(M-1,2) & \ldots & f(M-1,N-1) \end{pmatrix}$$

Figure 1: Image Identification of Matrix

$$f = \begin{pmatrix} f(1,1) & f(1,2) & \ldots & f(1,N) \\ f(2,1) & f(2,2) & \ldots & f(2,N) \\ . & . & & . \\ f(M,1) & f(M,2) & \ldots & f(M,N) \end{pmatrix}$$

Figure 2: Image Representation in MATLAB

3.1.2 Color Models in Image Processing

There are color codes or values written in specific color models.

YCbCr: Here Y is the luma component of the color. Luma component is the brightness of the color. That means the light intensity of the color. The human eye is more sensitive to this component. Cb and Cr is the blue component and red component related to the Chroma component. That means **"Cb is the blue component relative to the green component. Cr is the red component relative to the green component."** These components are less sensitive to the human eyes. Since the Y component is more sensitive to the human eye, it needs to be more correct and Cb and Cr is less sensitive to the human eye. Therefore, it needs not to be more accurate.

When in JPEG compression, it uses these sensitivities of the human eye and eliminate the unnecessary details of the image. The YCbCr color space is widely used for digital video. In this format, luminance information is stored as a single component (*Y*) and chrominance information is stored as two color-difference components (*Cb* and *Cr*). Cb and Cr represent the difference between a reference value and the blue or red component, respectively. (YUV, another color space widely used for digital video, is very similar to YCbCr but not identical.

Table 1: Attribute Description

Attribute	**Description**
Y	Luminance or brightness of the image. Colors increase in brightness as *Y* increases.
Cb	Chrominance value that indicates the difference between the blue component and a reference
Cr	Chrominance value that indicates the difference between the red component and a reference value.

The range of numeric values depends on the data type of the image. YCbCr does not use the full range of the image data type so that the video stream can include additional (non-image) information.

- For single or double arrays, *Y* is in the range [16/255, 235/255] and *Cb* and *Cr* are in the range [16/255, 240/255].
- For uint8 arrays, *Y* is in the range [16, 235] and *Cb* and *Cr* are in the range [16, 240].
- For uint16, *Y* is in the range [4112, 60395] and *Cb* and *Cr* are in the range [4112, 61680].

YCbCr is not an absolute color space. It is a way of encoding RGB information. The actual color displayed depends on the actual RGB primaries used to display the signal. The YCbCr image can be converted to/from RGB image. To convert from RGB to YCbCr, one variant of this color space (according to ITU-R BT.709): Y = 0.2126 *red+0.7152*green+0.0722*blue Cb = 0.5389*(blue-Y) Cr = 0.6350*(red-Y) Figure3: RGB colors cube in the YCbCr space

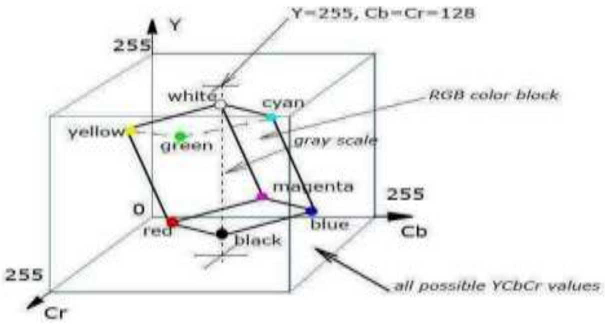

Figure 3: RGB colour cube to YCbCr space

3.2 METHODS

The basic flow the implementation is as follows.

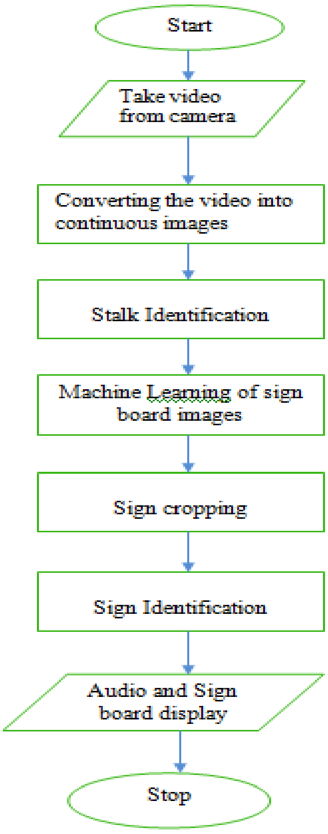

Figure 4: Flow of Implementation

1. Take video continuously from camera which is installed in the vehicle as the camera is always switched on when the engine of the vehicle is on.
2. Convert the video into continuous images.
3. Stalk is identified using image processing techniques.
4. Linear Regression model of machine learning is applied to the sign board images to get the minimum and maximum intensity values.
5. Sign board is cropped by taking the first white pixel of the stalk as the reference point. This is done to increase the accuracy/precision of the image obtained.
6. Now, the sign board is identified using the data from the two text files obtained in machine learning step.
7. Implement the above steps in Simulink.
8. Audio and display of the sign board identified is done through speakers and monitor respectively.

3.2.1 Stalk Identification

Figure 6 shows the images of stalk identification. Image is read using MATLAB function, and this RGB image is converted into YCbCr image. Cb component is used for the further process as the variation of the intensities in the stalk is high when compared to the background and so can be extracted easily. This image is converted into segmented image where our area of interest is on the stalk alone.

Figure 5: Stalk Identification Images

Intensity value ranges across the stalk are taken and are put inside a for loop to make the pixels between those particular range in white (255) and rest of the image in black (0). The background noise is eroded first and then the actual image without noise is again dilated. This way, the stalk is extracted from the entire image. The last step is to count the number of white pixels in order to get an idea as to the number of white pixels required for further processing.

3.2.2 Machine Learning

The concept of machine learning is used here in order to accept any number of images captured of the sign board. The system built should be able to handle even new set of images that are given as the input to the already existing code [10]. The images in RGB are converted into YCbCr once again and then the result is converted into Y component (shown in figure 6) as the brightness levels of a particular sign is highest in Y component of YCbCr.

Here, our region of interest is the sign board and not the stalk. We obtain the Y component and carry out the further process. As shown in figure 8, the image is converted into Y component and considering the intensities, the range of intensities are saved in two different text files which is further used in the process.

Figure 6: YCbCr Converted into Y Component

3.2.3 Sign Cropping

Sign cropping is particularly done in order to precisely map all the intensity values of the image for identifying the particular sign in the next process. This step helps in reducing the complexity of the system. Here we consider the count value i.e., particularly 6000 in our study, to get the clear image of the sign board and to find the co-ordinates as reference point. When the count is greater than 6000, the video is paused. Now we locate the first white pixel and make it as a reference point having coordinates(x, y1).This reference point is used to crop the image with the range given as: (x-160: x), (y1-70: y1+90) because the dimensions of the board is 160*160(rows, columns). With this process, the image is cropped and displayed.

3.2.4 Identification of Sign Board

Here we store all the higher and lower intensity value of the image in two different text file. And then stored in an array (named as arr1 and arr2 in our study) as shown in figure 7.

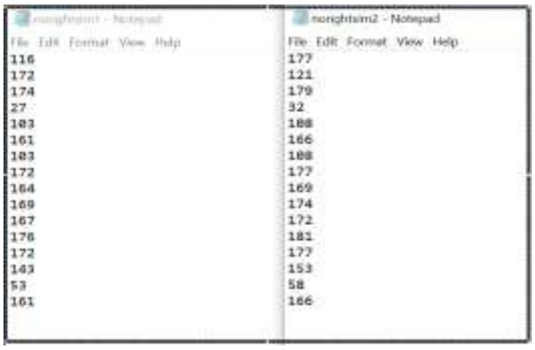

Figure 7: Text files

Whenever an input is read, it will compare the intensity values in these ranges. If the value lies between higher and lower intensity values, the image is correctly identified.

4. IMPLEMENTATION

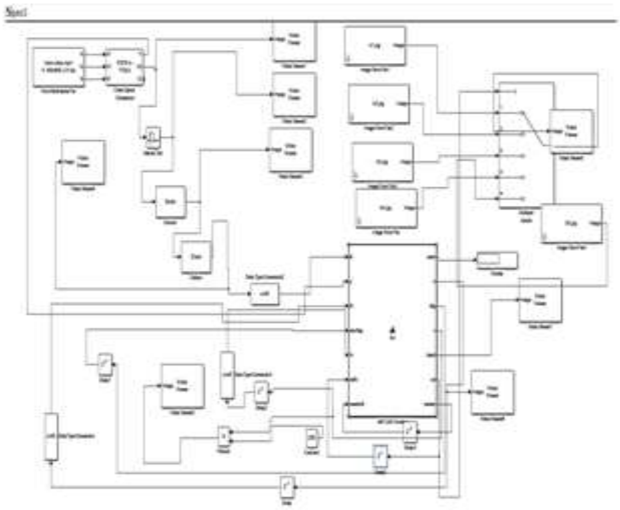

Figure 8: Implementation Model

We have implemented above model which shows the output.

Here multiport switch is particularly used in identifying images. This multiport switch gives the outcome through video viewer. We have also used text-to-speech conversion module to give the audio warning.

5. RESULT ANALYSIS

After implementing the above model, we have tested for different traffic sign boards. The figures followed shows the results derived for some of the sign boards

Figure 9: No-horn and No-right sign board output

6. CONCLUSION

The traffic sign detection and recognition system enables the user to stay relaxed and only alerts the user ahead of time regarding any sign board present on the path. This system also prevents further collision or accidents that may have occurred otherwise. We have used simple Linear Regression Model of Machine Learning, it reduces the complexity of the system leading to faster results which makes the system more efficient. The reduction in the number of hardware components makes it a more cost effective system. The system therefore, has the capacity to lower the statistical rates of death or accidents on roads significantly if implemented effectively in all the vehicles.

ACKNOWLEDGEMENT

We would like to express our heartfelt gratitude and respect to all those who guided us in the completion of this work. We would also like to extend our heartfelt gratitude to our guide Prof. A.R.Manjula for her constant support and guidance.

REFERENCES

[1]. A. Ruta, Y. M. Li, and X. H. Liu, "Detection, tracking and recognition of traffic signs from video input.

[2]. J. P. C. Pascal, Advanced driver assistance system based on computer vision using detection, recognition and tracking of road signs using machine learning.

[3].Abdi, Lotfi&Meddeb, Aref. Deep learning traffic sign detection, recognition and augmentation.IEEE, 2017

[4].EricTzeng,JudyHoffman,TrevorDarrellUCBerkelz.SimultaneousDeeptransferAcross Domains and Tasks. IEEE,2015.

[5]. P. Paclík, J. Novovičová, and R. P. W. Duin, "Building road-sign classifiers using a trainable similarity measure

[6].W. J. Jeon, G. A. R. Sanchez, T. Lee, Y. Choi, B. Woo, K. Lim, and H. Byun. Real-time detection of speed-limit tra_c signs on the real road using Haar-like features and boosts cascade. And Communication, page 93. ACM, 2014.

[7]. Y. Jail, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and S.Darrell.Caffe:Convolutionalarchitectureforfastfeatureembedding.InProceedingsofthe ACM International Conference on Multimedia, pages 675–678. ACM, 2014.

[8]. A. Mogelmose, M. M. Trivedi, and T. B. Moreland. Vision-based traffic sign detection and analysis for intelligent driver assistance systems: Perspectives and survey. Intelligent Transportation Systems, IEEE Transactions on, 13(4):1484–1497, 2012.

[9]. Digital Image Processing - A Remote Sensing Perspective, Jhon R. Jenson

[10]. Y. Bengio. Learning deep architectures for ail. Foundations and trends R in Machine Learning, 2(1):1–127, 2009