International Journal of Advance Research in Science and Engineering
Volume No.06, Issue No. 11, November 2017
www.ijarse.com
IJARSE
ISSN: 2319-8354

REAL-TIME INDIAN TRAFFIC SIGN DETECTION USING RASPBERRY PI AND OPEN CV

N Radhakrishnan[1], S Maruthi[2]

[1]*Assistant Professor, Dept. of EEE, CEG, Anna university, Chennai, Tamil Nadu, (India)*

[2]*PG Student, M.E.[EST], Dept. of EEE, CEG, Anna university, Chennai, Tamil Nadu, (India)*

ABSTRACT

The problem related to traffic is a complex one requiring proper design and planning for developing a proper solution. For safety and harmony in the flow of traffic, certain rules are established by governments all over the world and some of them are displayed by means of traffic signs. Drivers are expected to pay attention to identify, interpret and follow them while driving. Misinterpretation of traffic signs may lead to catastrophes. An automatic system in a car which detects, recognizes, interprets and gives warring to the driver would be a great help in reducing the number of road accidents and will be appreciated by all. In this paper a system is proposed for the automatic detection and recognition of Indian traffic signs from images captured by a camera which forms a part of Advanced Driver Assistance Systems (ADAS). It is implemented using Raspberry Pi 3 hardware - a credit card sized Single Board Computer (SBC) developed in the United Kingdom by the Raspberry Pi foundation, running Raspbian Stretch - a Debian based operating system optimized for Raspberry Pi which is officially provided by the manufactures themselves. Python integrated with OpenCV library is the programming environment used for implementing Image Processing algorithms related to the Traffic Sign Detection and Recognition. The entire system is implemented using open source hardware platform and open source software environment. Unlike some of the related work which considers static images, our system works on real time images.

Keywords: ***Traffic sign Detection, Raspberry Pi, OpenCV, HSV algorithm.***

I. INTRODUCTION

Every person, whether a passenger, driver, pedestrian would have noticed along the roadside various sign board that serve important purposes. These important road paraphernalia help us as route guides, warnings and traffic regulators. As control devices for traffic, signs need full attention, respect and appropriate driver's response.

The Road Signs we were around us date long back in history. The earliest road signs were milestones, giving distance or direction. In the middle ages, multidirectional signs at intersections became common, giving directions to cities and towns. With the advent of motorized traffic and its increasing pressure on road, many

have adopted pictorial signs and standardized their signs to facilitate international travel, where language differences would create barriers. In general it is used to help enhance traffic safety through appropriate caution, regulation and informatory signs. Most of them use symbols in place of words and have international recognition and acceptance. These signs were primarily evolved in Europe, and have been adapted by most countries.

In India, Motor Vehicle Act 1988 has laid down the uniform Road Signs in its Schedule I which comprehensively explains the shape and sizes of these road signs. Article 5 of Chapter II of the Convention on Road Signs and Signals held on 8th November 1968 lays down the classes of Road Signs, which were broadly categorized into:

a) Mandatory signs
b) Cautionary signs
c) Informatory signs

A further guide to the function of a sign is its color. Blue circles give a mandatory instruction such as "Compulsory Turn Left" etc. Blue Rectangles are used for information signs. All triangular signs are red. There are few exceptions to the shape and color rules, to give certain sign greater prominence. Examples are the "STOP" and "GIVE WAY".

This Section gives few road sign examples for Mandatory (Table 1.1), Cautionary (Table 1.2) and Informatory (Table 1.3) categories and a brief description of each sign is provided.

1.1 Mandatory Signs

These signs are obligatory on the traffic which used a specific area of road. These signs indicated what must one do, rather than must not do. Mandatory Road signs are generally round in shape with red border. Some of them are blue in color. "STOP" and "GIVE WAY" are octagon and triangular, respectively in shape. Violation of these signs attracts heavy fines and punishments. Importantly, violation of these could lead to major accidents also.

Table 1.1 - Some Examples of Mandatory Road Signs

Sl. No.	Sign	Meaning
1.		The Most important and prominent road sign. This sign indicates that driver should immediately stop. Usually Police, Traffic and Toll Authorities use this sign at check posts.

2.		This sign is used at roundabouts where a specific lane discipline is to be followed. This sign directs the traffic to give way to the fellow traffic on the right side of the driver.
3.		This sign notifies that entry is prohibited for all vehicles. Certain portions of an area or road are demarcated as 'NO ENTRY' areas for traffic. This could be entry to a restricted area or no traffic zone. So the driver should obey it and divert his route.
4.		This sign designates the speed of traffic on road. The Limit specified must be invariably followed to avoid penal action and accidents on the road.
5.		This sign indicates that the driver should drive in left lane for smooth traffic flow. The sign is installed mainly on the roads which do not have divider in between any two way traffic flows on the same road.

1.2 Cautionary Signs

These signs are meant to caution the driver about the hazards/situation lying ahead on the road. The driver should obey these for his safety. Though violation of these road signs do not attract any legal action, they are very important for the fact that avoiding them could result in major accidents. Cautionary signs are triangular in shape with red border.

Table 1.2 - Some Examples of Cautionary Road Signs

Sl. No.	Sign	Meaning
1.		This sign cautions about a sharp left turn on the road ahead. These are essentially erected on hilly roads. It gives time to reduce the speed to manage the turn and also sets eyes of the driver on turn.
2.		This road sign indicates that ther is stee ascent ahead and driver should get ready to climb and put the vehicle in relevant gear. Most of the times, these signs are found on hilly road where steep ascents and descents are normal part of travel.

3.		Sometimes road converges to a bridge which is of less width than that of road. This sign is erected before such bridges which are narrower than the road. The driver should reduce the speed and watch for oncoming traffic for safe drive.

1.3 Informatory signs

These signs are meant to provide information on direction, destination, roadside facilities, etc. to the road user. Following informative road signs helps a driver in saving time, reaching destination without wandering around. These signs are generally facilitators to the driver and signs are normally blue in color. The sign may have direction arrow and also the distance facility from the sign.

Table 1.3 - Some Examples of Informatory Road Signs

Sl. No.	Sign	Meaning
1.		This informatory sign indicates that there is a Petrol Pump ahead. Sometimes distance is also indicated on this sign which gives an idea about its location from the sign post.
2.		This sign indicates that there is Hospital nearby. The driver should be careful while driving through this stretch and should not honk unnecessarily.
3.		This sign indicates entry to a pedestrian underpass/subway. Pedestrians should invariably use these underpass/subway to cross the road.

II - HARDWARE PLATFORM AND OPERATING SYSTEM

2.1 Raspberry Pi

The Raspberry Pi is perhaps the most inspiring computer available today. Although most of the computing devices being used (including phones, tablets, and game consoles) are designed to stop people from tinkering with them, the Raspberry Pi is exactly the opposite. It allows the user to prod it, play with it, and create with it. It comes with the tools needed to start creating one's own software.

Lots of people are fired up about its potential, and they're discovering exciting new ways to use it. Dave Akerman and friends attached one to a weather balloon and sent it nearly 40 kilometers high to take pictures of the Earth from near space using a webcam.

Professor Simon Cox and his team at the University of Southampton connected 64 Raspberry Pi boards to build an experimental supercomputer, held together by Lego bricks. In the supercomputer the individual Raspberry Pi boards work together to solve a single problem. The project has been able to cut the cost of a supercomputer from millions of dollars to thousands or even hundreds of dollars, making supercomputing much more accessible to schools and students.

The Raspberry Pi is a family of credit card-sized Single Board Computer (SBC) developed in the United Kingdom by the Raspberry Pi Foundation. Raspberry Pi Foundation's Raspberry Pi was released in 2012. It was a massive hit and sold over two million units in two years. Subsequently, the Raspberry Pi Foundation revised versions of the Raspberry Pi. The latest one is Raspberry Pi 3 and it is shown in Fig. 2.1.

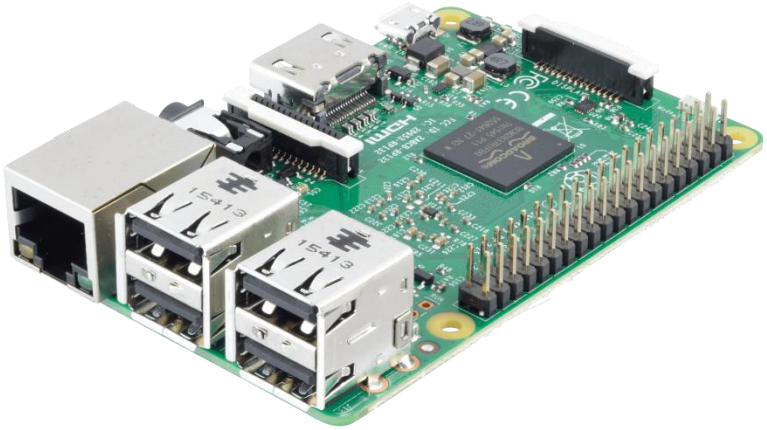

Fig. 2.1 - Raspberry Pi 3

The Technical Specifications of Raspberry Pi 3 is given in Table 2.1.

Table 2.1 Raspberry Pi 3 Technical Specifications

Architecture	ARMv8
SoC	BCM2837
CPU	1.2GHz 64-bit quad-core ARM Cortex-A53
GPU	Broadcom VideoCore IV (3D part of GPU @ 300MHz, video part of GPU @ 400MHz)
Memory	1 GB (shared with GPU)
Video	HDMI rev 1.3 and Composite Video RCA jack
USB	2.0 - 4 Ports
On-board Storage	Micro SDHC slot
On-board network	10/100 Mbps Ethernet, Bluetooth, and WiFi
Power source	5V via MicroUSB
Power rating	800 mA (4W)

2.2 Raspbian OS

Raspbian is a Debian-based computer operating system for Raspberry Pi. Since 2015 it has been officially provided by the Raspberry Pi Foundation as the primary operating system for the family of Raspberry Pi single-

board computers. Raspbian was created by Mike Thompson and Peter Green as an independent project. The initial build was completed in June 2012. The operating system is still under active development. Raspbian is highly optimized for the Raspberry Pi line's low-performance ARM CPUs.

Raspbian uses PIXEL, Pi Improved Xwindows Environment Lightweight as its main desktop environment as of the latest update. It is composed of a modified LXDE desktop environment and the Openbox stacking window manager with a new theme and few other changes. The distribution is shipped with a copy of computer algebra program Mathematica and a version of Minecraft called Minecraft Pi as well as a lightweight version of Chromium as of the latest version.

III - SOFTWARE ENVIRONMENT

3.1 Python

Python is a general purpose programming language developed by Guido van Rossum, which became very popular in short time mainly because of its simplicity and code readability. It enables the programmer to express his ideas in fewer lines of code without affecting the readability.

Compared to other languages like C/C++, Python is slower. But another important feature of Python is that it can be easily extended with C/C++. This feature helps the user to write computationally intensive codes in C/C++ and create a Python wrapper for it so that it can be used as Python modules. This gives two advantages: First, the code is as fast as original C/C++ code - since it is the actual C++ code working in background and Second, it is very easy to code in Python. This is how OpenCV-Python work, it is a Python wrapper around original C++ implementation.

The support of Numpy makes the task easier. Numpy is a highly optimized library for numerical operations. It gives a MATLAB-style syntax. All the OpenCV array structures are converted to-and-from Numpy arrays. Operations available in Numpy, can be combined with OpenCV. Besides that, several other libraries like SciPy, Matplotlib which supports Numpy can be used with this. Hence Python-OpenCV binding is an appropriate tool for fast prototyping of computer vision problems.

3.2 OpenCV

OpenCV (Open Source Computer Vision Library) is an Application Programming Interface (API) developed by Intel which can be used for many image processing and computer vision applications. OpenCV officially launched in 1999 and the project was initially an Intel Research initiative to advance CPU-intensive applications.

OpenCV library is a collection of algorithms and C/C++ functions and a few classes that implement some Image processing and computer vision algorithms. There is active development on interfaces for C, C++, Python,

Ruby, Matlab and other languages. OpenCV was designed for computational efficiency and with a strong focus on real time applications. OpenCV is written in optimised C and can take advantage of multicore processors. OpenCV contains over 500 function that span many areas in vision, including factory product inspection, medical imaging, security, user interface, camera calibration, stereo vision and robotics.

The principles behind the creation of the library is to aid commercial uses of computer vision in human-computer interface, robotics, monitoring, biometrics and security by providing a free and open infrastructure where the distributed efforts of the vision community can be consolidated and performance optimized. OpenCV support for vision is extensively including routine support for input, display, and storage of movies and single images

One of the OpenCV goals is to provide a simple-to-use computer vision infrastructure that helps people build fairly sophisticated vision applications quickly. There are several goals of OpenCV in outset which are following:

- Advance vision research by providing not only open but also optimized code for basic vision infrastructure.
- Disseminate vision knowledge by providing a common infrastructure that developers could build on, so that code would be more readily readable and transferable.
- Advance-vision based commercial applications by making portable, performance optimized code available for free- with a license that did not required commercial applications to be open or free themselves

OpenCV provide a set of image processing functions and computer vision applications. The functions are optimized for Intel architecture processors and are particularly effective with MMX technology. The OpenCV Library is a way of establishing an open source vision community that will make better use of up-to-date opportunities to apply computer vision in growing PC environment and mobile platform. The library is open and has platform independent interface and supplied with whole C sources.

OpenCV was designed to be portable. It was originally written to compile across Borland C++, Microsoft Visual Studio C++, and the Intel compilers. C and C++ code had to be fairly standard in order to make cross-platform support easier. OpenCV library is multi platform, and runs on both Windows and Linux Operating System. OpenCV is quickly gaining popularity for developing real-time computer vision applications. Some examples of applications include face recognizers, object recognizers, and motion trackers, just to name a few. The library has especially gained popularity in the computer vision research community. It allows researchers to get demos or research projects up and running quickly, and take advantage of the large collection of algorithms that are already available.

The use of term of computer vision and image processing is commonly interleaved. In contrary, there is a gap between computer vision and image processing. Image processing is of low-level processing of still or video image, while computer vision is high-level processing of still or video image. OpenCV is specifically designed to an advent to computer vision development. OpenCV aimed at providing the basic tools needed to solve computer vision problem. High level library will be sufficient to solve more complex problems in computer vision. Basic components in the library are complete enough to create complete solution to almost any computer vision problem.

OpenCV is broadly structured into five main components, four of which are shown in Fig. 3.1:

- CV component contains the basic image processing and higher-level computer vision algorithms
- MLL is the machine learning library, which includes many statistical classifiers and clustering tools.
- HighGUI contains I/O routines and functions for storing and loading video and images and
- CXCORE contains the basic data structures and content.

The above figure does not include CVAUX, which contains both defunct areas (embedded HMM face recognition) and experimental algorithms (background/foreground segmentation).

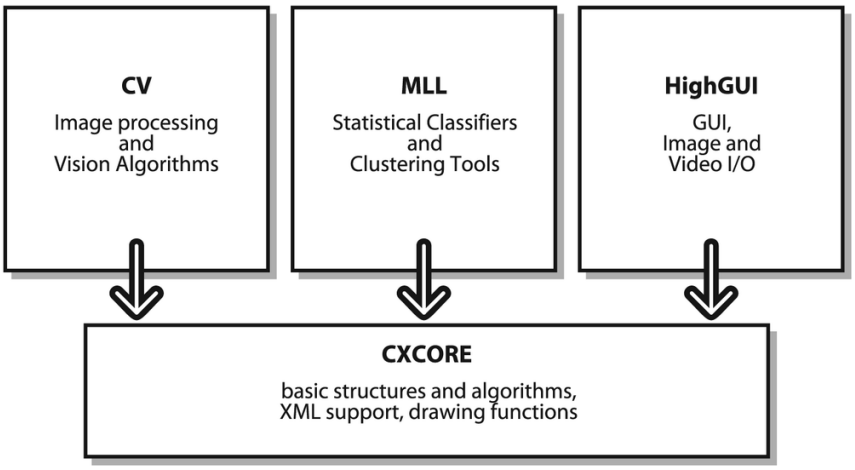

Fig. 3.1. - Components of OpenCV

IV - HSV ALGORITHM

The structure of the proposed system is shown in Figure 4.1. The inputs of the proposed scheme can be the video captured by any video capture device. And this paper uses a Pi Camera Module to capture videos. In experiments of this paper, Pi Camera Module which is connected to the Raspberry Pi installed in a moving vehiclee to real-time capture scenes in the front of this vehicle. These frames can be considered as input images of the traffic sign recognition scheme.

International Journal of Advance Research in Science and Engineering
Volume No.06, Issue No. 11, November 2017
www.ijarse.com
IJARSE
ISSN: 2319-8354

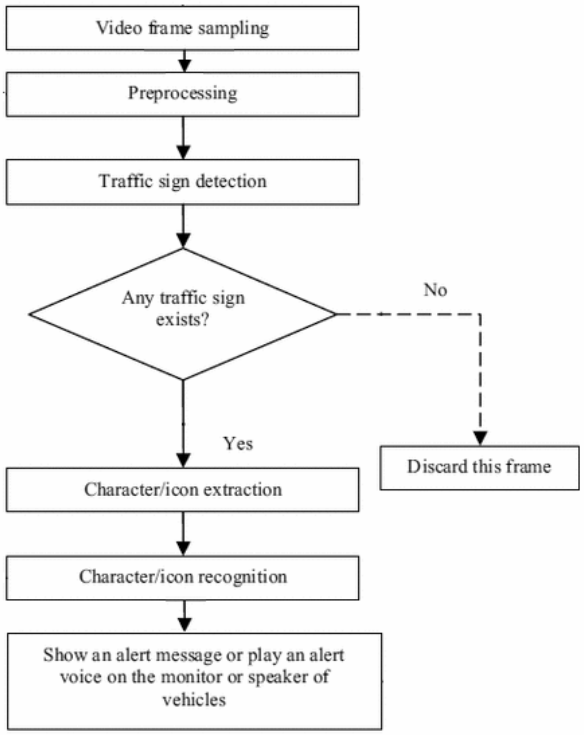

Fig. 4.1 - Structure of the Proposed System

The major recognition scheme can be performed on in-vehicle computing device, Notebook, or other computer devices installed in intelligent vehicles. In this paper, the proposed scheme can be divided into five stages: video and frame capturing, preprocessing, traffic sign detection, character/icon extraction and recognition. The first stage is to capture the front view of a vehicle as the dynamic video, and then sampling frames are extracted from the video in certain frame rate. These extracted frames are further are processes, and then recognized through the following stages. In this paper, all above works can be performed in Raspberry Pi 3. The preprocessing stage contains some image processing procedures to improve and normalize the image quality and format. The traffic sign detection is to detect the region of each traffic sign, and the last two stages are used to further recognize each detected traffic sign.

4.1 Preprocessing

The videos are real-time captured by the Pi Camera Module which is connected to the Raspberry Pi installed in a moving vehicle. Hence, the brightness, contrast, clarity and noises of scenes may have large difference when the weather or other conditions are changed by time and locations. However, these variables could increase recognition difficulty and affect the recognition results. In order to increase the robustness of the proposed scheme, some preprocesses are used to reduce the influence of variable conditions.

RGB color space is a very general color space for monitor displaying and image processing. However, when the brightness is changed, all color channels also would be changed. At the sign detection stage, all signs are detected when their colors are satisfied with a specific color condition. However, the specific color condition should be changed in different brightness conditions. In order to detect traffic signs by a fixed color condition, the RGB color space is transformed into other color spaces, such as, HSI and HSV in the preprocessing. Both HSI and HSV only record the brightness of images in I and V channels, respectively. Hence, other channels only are stored the color information. HSV and HSI values are calculated using the standard formulas.

4.2 Traffic Sign Detection and Extraction

Generally speaking, the mandatory and cautionary signs are most important for drivers and their contents must contain a red circle or triangle frame. Hence, color extraction is a very effective and efficient solution for selecting candidate sign regions in each image. However, each color of the human vision has a fuzzy range in computer color spaces. For example, the proposed scheme wants to extract all regions which are "red" as the frame of a sign from an image. In order to extract all "red" regions, some color conditions are needed to select red pixels by computers. In this paper, the color selection is performed in the HSV color space. This color space can be obtained from the RGB color space at the preprocessing stage itself. The proposed scheme defines thresholds for hue (H), saturation (S) and value (V) color channels, respectively. This paper detects colors in the HSV space. This color space only stores the brightness information in V color channel. Hence color thresholds can be defined for H and S color channels to reduce the effect of brightness. In HSV color space, the most brightness information is stored in V channel and color information is recorded in H and S channels. Hence, the influence of variable brightness can be reduced when the "red" pixel selection is determined by H and S values. H records the hue value of each pixel and the colors similar to the red are around 0°, this paper uses two thresholds ξ_1 and ξ_2 to define the range of the red. Another threshold ξ_{sat} is used to define the range of the satisfied saturation. The "red" pixel $f_{red}(x,y)$ can be defined as:

$$f_{\text{red}}(x,y) = \begin{cases} 1, & \text{if } H(x,y) \geq \xi_s, \\ & H(x,y) \leq \xi_1, \\ & \text{and } S(x,y) \geq \xi_{sat}, \\ 0, & \text{otherwise,} \end{cases}$$

The color selection is a suitable and simplest solution for detecting traffic sign candidate regions. However, the image may contain other red-likeness objects, such as, commercial signboards, vehicles, flowers or any object with red color. Hence, this color selection method only can be used to obtain all candidates of prohibitory and warning signs. This paper presents a simple traffic sign object filter which can be performed to remove most objects which are not traffic signs.

The traffic sign detection and extraction scheme can be used to obtain the traffic sign sub-images from each video frame. Besides, the shape recognition can provide the preliminary category information of each traffic sign object to help the following stages further recognize traffic sign more efficiently. In order to gain more performance this stage can be applied in the images with lower resolution. When any satisfied traffic sign cannot be detected in certain frame, this frame would be not processed in the following stages. The following stages focus on traffic sign content recognition.

V - RESULTS

In experiments of this paper, the devices contain a notebook computer and a smart phone to simulate the traffic sign recognition environment in an intelligent vehicle. The notebook computer is used to simulate the in-vehicle computing device. The smart phone is the video capturing device. This smart phone is equipped a camera, a wireless network, and Android operating system in the experiment. All sizes of video frames are 320 × 170 pixels and all patterns are normalized as 50×40 pixels. Some example frames are used to show the performance of the proposed scheme, these example frames are shown in Figure 5, and they are considered as source images in the proposed scheme. At the first stage, all images are transformed from RGB color space into HSV color space. Hence, the red regions of each image are detected by the hue and color saturation values of pixels. The detection of stop sign, speed brake sign and speed limit sign results are shown in Figure 5.1, 5.2 and 5.3 respectively

Fig 5.1 Stop Sign Detection

Fig. 5.2 Speed Brake Sign Detection

Fig. 5.3 Speed Limit Sign Detection

VI - CONCLUSION AND FUTURE WORK

The traffic sign recognition is a very helpful driver assistance technique for increasing traffic and driver safety. This paper proposes efficient real-time traffic sign detection scheme and integrates the proposed scheme with intelligence vehicles. The proposed scheme detects color by dynamic thresholds in HSV color space. And then the proposed scheme uses low computing complexity, adaptive and accurate mechanisms to extract portion of the image which contains the traffic sign. In Future algorithms for recognition of the extracted traffic signs can be implemented using appropriate techniques.

REFERENCES

[1] Ching-Hao Lai, Chia-Chen Yu, 'An Efficient Real-Time Traffic Sign Recognition System for Intelligent Vehicles with Smart Phones', IEEE International Conference on Technologies and Applications of Artificial Intelligence, 2010.

[2] P. Shopa, Mrs. N. Sumitha, Dr. P.S.K Patra. (2014), 'Traffic Sign Detection and Recognition Using OpenCV', International Conference on ICICES2014 - S.A.Engineering College, Chennai, Tamil Nadu, India.

[3] Dr. D. Y. Patil., 'A Road Sign Detection and the Recognition for Driver Assistance Systems' International Conference on Energy Systems and Applications (ICESA 2015).

[4] Prachi Gawande1,'Traffic Sign Detection and Recognition Using Open CV' International Research Journal of Engineering and Technology (IRJET), e-ISSN: 2395 -0056, p-ISSN: 2395-0072, 2017.

[5] Gary Bradski and Adrian Kaehler, Learning OpenCV 3 Computer Vision in C++ with the OpenCV Library (O'Reilly Media, 2016).

[6] Mark Lutz, Learning Python: Powerful Object-Oriented Programming: 5th Edition (O'Reilly Media, 2013).

[7] https://www.raspberrypi.org/ - The official website of Raspberry Pi.