# Autonomous Transportation Robot for Indoor as well as Outdoor Travel and Navigation using computer vision

Shweta Gajbhiye, Shrutika Tarekar, Gayatri Sakharkar, Shraddha Kharabe, Mrunali Kongre

*Computer Science and Engineering Department, Priyadarshini J.L College of Engineering - Nagpur, India*

**Abstract: -** Autonomous robot are a hot research topic in science and technology, which has a great influence on social and economic development. This paper proposes a design for an autonomous robot that is capable of autonomous navigation both Indoor as well as outdoor. Using the concept of CNN algorithm, computational geometry and computer vision will be used for detection of signboard for outdoor travel. The robot performed quite well outdoors avoiding both static obstacles as well as dynamic obstacles that presented themselves along the planned path. Lane tracking, assisting the vehicle to remain in the desired path, and it controls the motion model by using previously detected lane markers. The image processing used for lane detection is done using computer vision techniques. Along with lane detection, an autonomous robot must also be able to detect traffic signs and traffic signals. Ultra-sonic sensors are used to detect obstacles.

**Keywords: -** Automatic lane detection and tracking, Computer Vision, Convolutional neural network

## I. INTRODUCTION

Robots are designed to figure in any surroundings and perform the task on behalf of humans. There are several potential advantage of autonomous vehicles like saving lives as autonomous vehicles will considerably scale back the number of crashes, enhanced quality for young or disabled persons. The project relies on autonomous as suggests that of transportation for indoor and outdoor travel. Decision-based actions involve some basic tasks like starting, stopping, and maneuvering around obstacles that are in their way. The robot works autonomously with navigations to redefine ways and avoid obstacles and goes around on that and finds the most effective path.

The autonomous robot works on a convolutional neural network algorithmic program for out of door travel that involves traffic signal detection, signboard detection, lane detection, etc. The convolutional neural network is a type of artificial neural network which is used for image processing problems, computer vision tasks like segmentation, localization, obstacle detection, and speech recognition. Several analysis fields focus on CNN to get an accurate result. A CNN model includes four layers i.e convolutional layers, Relu activation, Polling layers, fully connected layer. If any of the layers fail to perform their task then the algorithm will never be executed. Convolution is the first step to creating the complete network. Here, Convolutional operates on two images in two-dimensional (2D) format. One as the input image, and another output image. It helps to grasp the feature from images by creating a relation between pixels [5]. Within the convolutional layer, we have to first line up the feature in conjunction with the image and then multiply the pixel value with the corresponding value of filter and then add them up driving them with the absolute value of pixels. Relu layer represents corrected layer unit generated by this layer is to remove all the negative value from the filtered image and after change it with zero. The image received from RELU is downscaled within the pooling layer. And the output is connected to a fully connected layer i.e the actual classification is done in a fully connected layer. We have to take the downscaled image and up that in a single list. And then we compare that image with our previously-stored list. The last is the output layer which supplies the output of the classified image [6].

In Autonomous Vehicle, lane detection is employed to help keep in a very specific lane. It plays a vital role in moving a vehicle to an alternate lane. Lane detection supported color region, line selection, edge selection, and also the Hough transformation. Image recognition computer code utilized in conjunction with cameras can enable the recognition of other vehicles, and also the detection and interpretation of traffic signs. Real-time traffic data can be used to determine the best route to be used to reach a destination.

## II. LITERATURE REVIEW

2.1 Comparison of Machine Learning Algorithm's on Self-Driving Car Navigation using Nvidia Jetson Nano: Many applied math reports pointed out that quite eightieth of accident causes came from direct human causes like violating the regulation, criminal passing, and suddenly cutting in. Therefore, the self-driving automotive was quickly developed by beginning a scaled RC-Car platform. During this analysis, they designed a self-driving automotive for collection knowledge. Nvidia Jetson Nano may be tiny silicon chipboard for developing and coaching models by mistreatment GPU 128-core Maxwell to quickly process AI frameworks and models for applications like image classification, object detection, and segmentation [1]. For the coaching model, they used the 3 models, that square measure Support Vector Machine (SVM), Artificial Neural Network Multilayer Perceptron (ANN-MLP), and Convolution Neural Network - Long Short Term Memory (CNN-LSTM) for comparison to finding the most effective accuracy for self-driving automotive model (SDCM). The SVM will encourage each classification and regression issue, together with the linear and non-linear hyper- plane by employing a kernel perform to scale back difficult feature areas. The ANN-MLP is a man-made neural network and it's a statistic figurer to use for classifying and detective work objects. Convolution Neural Network Long remembering (CNN-LSTM) is one in each of the models that are appropriate for fixing classification issues, that comprises 5 main layers: Convolution stage, Detector stage, Pooling stage, LSTM stage, and absolutely connected stage.

They propose 3-speed levels and three eventualities for comparison the accuracy of every algorithm: SVM, ANN-MLP, and CNN-LSTM. For the primary experiment, they originated the three-speed levels, that square measure one, 2, and 3 km/h, severally, while not an obstacle on the road. per 1st experiment, it will be seen that the share of the accuracy rate of the CNN-LSTM rule is the highest performance of all models at each speed level while not an obstacle on the road. Per the second experiment, though they add one condition to a state of affairs by adding AN obstacle, CNN-LSTM remains the most effective accuracy rule. Within the final experiment, it's apparent that even supposing we have a tendency to add a lot of the obstacles on the road, the share of accuracy rate of the CNN-LSTM rule is above the other rule within the experiment.

From the comparison algorithms of machine learning: SVM, ANN-MLP, and CNN-LSTM on totally different| completely different} eventualities and different speed levels. From the experiment, it will be seen that the share of the accuracy rate of the CNN-LSTM rule is that the highest potency, not solely with obstacles however conjointly while not obstacle.

2.1.1 Strength:

- CNN-LSTM, has been established terribly sure-fire in recognizing and classifying pictures for pc vision.
- Jetson Nano record {the pictures| the pictures| the photographs} and driving data resolution of pictures is 320 x 240 pixels that automotive model collected knowledge pictures contained over 3600 images.
- Adam optimizer technique decreases the training rate of some parameters to seek out the most effective accuracy model for SDCM.

2.1.2 Weaknesses:

- The share of the accuracy rate of all algorithms bit by bit drops once adding a lot of obstacles and a lot of high-speed levels.

2.2 Evaluating and enhancing Google Tango localization in indoor environments using fiducial markers: In this paper, the motion tracking capabilities with fiducial markers (QR Codes) are introduced with the methods of google project tango. Conducted all the measurements using slow and fast movements for all methods like motion tracking, motion with area learning, tango and marker. Method 1 (motion tracking) does not retain any memory of what it sees and does not need any previous information of the environment but it produced the worse result with the higher mean position and orientation errors for both types of movements. Method 2 (tango with area learning) needs to store what it sees and remember it. To take advantage of this learning mechanism the working area needs be previously scanned (learned) to produce an Area Description File (ADF). It produced the lower mean position error, providing to the best solution. Third method is motion tracking with markers. Objective of this method is to allow the recalibration of the position and orientation of the tracking when viewing markers hoping for an accuracy improvement. The white square which is a virtual marker of the current position given by the tango, motion tracking, and the red frame which is a marker of real-world position are overlapping which reinforces the accuracy level of the proposed model. After the re-calibration, the augmented content (white space) is better aligned with the real markers, reducing the positioning and orientation error accumulated with tango motion tracking. It allows to calibrate the position and orientation, improving the accuracy independency of the condition even in the dynamic condition. Study suggests that the accuracy of the 3D indoor navigation could benefit from a solution that combines method 2 and 3, area learning with markers, providing a mechanism to calibrate augmented content when the scenario does not have enough visual features or has changing characteristics regarding geometry or illumination, which compromise the accuracy.

2.3 AR Based Navigation using Hybrid Map: The Augmented Reality (AR) navigation system can provide a new experience for pedestrians compared to the conventional navigation on 2D map. The proposed system adopts Simultaneous Localization and Mapping (SLAM) to build a point cloud map, and performs positioning and navigation in a novel hybrid map which integrates the 3D point cloud map and floor map of the indoor environment. Basically, there are two components in the proposed system: Mapping and Navigation. The proposed system utilizes the RGB-D camera to observe the surrounding environment, and adopts ORB-SLAM to create a point cloud map. After that, the point cloud map and floor map are manually integrated together to become a hybrid map. The positioning and navigation are performed in the hybrid map.

2.3.1 Mapping:
ORB-SLAM is a real-time SLAM library for Monocular, Stereo and RGB-D cameras that computes the camera trajectory and a sparse 3D reconstruction (in the stereo and RGB-D case with true scale). It can detect loops and re-localize the camera in real time. ORB-SLAM provides both SLAM Mode and Localization Mode. This research firstly uses the SLAM mode to create a point cloud map. The created point cloud map includes the features of indoor environment. This point cloud feature map will be used to calculate the user position in the positioning and navigation step.

2.3.2 Positioning and Navigation:
Positioning and navigation are performed in the pre prepared hybrid map. In the positioning step, the user wears the head-mounted device equipped with a RGB-D camera, and walks in the corridor. The RGB-D camera outputs RGB image and depth information. These two image sources are used to re-localize the camera in the hybrid map by feature matching. This step is conducted based on Localization Mode of ORB-SLAM. The position and orientation of the camera can be calculated in the positioning step, and be converted to rotation matrix R and Translation matrix T.

2.3.3 Strengths:
The proposed system can provide the satisfied navigation information in 75% frames.

2.3.4 Weakness:
The hybrid map where the floor map and cloud map are aligned, is prepared manually.

2.4 Indoor and Outdoor Voice Based Navigation Detection using Light based Communication (LiFi): Indoor positioning system has made it easier for people to locate places and objects in easy manner and this technique has been undertaken for research for improvement and efficiency and the techniques used under them have various solution for obstacles been faced, some types of solutions would be WLAN, RFID, UWB and other Bluetooth based system. Every system user this procedure have their own performance analysis for the credibility of exactness that system gives to the end user. The Wi-Fi which is used in indoor determines the position based on the received signal strength. All repeaters will transmit signals simultaneously. And also utilizing invisible light source for the transmission of data from the transmitter to the receiver. The algorithm proposed allows to receive the identical repeater without actually determining the full length measurement of the repeaters. This will eliminate the impact on the dynamic position in performance where high switching can be avoided.

2.4.1 LiFi:
LiFi can be stated as visible light communication system transmitting wireless communication rates are about 224 GB/sec, also recognized as the world's fastest Wi-Fi. Encoded Polyline Algorithm: This encoded polyline algorithm is mainly used for storing a series of points in a string and it's a type of lossy compression algorithm. Then the points are sort out using their significant value. When there is a fixed point we are directed to our wish for using polyline

encoded interactive utility. The method is mainly utilized for the conversion of 1 and 0 into segment of similar codes accordingly to the ASCII characters.

2.4.2 Strength:
- It is open ended.
- Allows to store a series of coordinates as a single string.

2.4.3 Weaknesses:
It doesn’t connect last point to first.

## III. METHODOLOGY

CNN based Traffic Sign Detection and Recognition for outdoor travel: Generally, traditional computer vision methods were developed to detect and recognize traffic signs on signboards, but this method requires time-consuming manual work to extract important features in images. While applying deep learning to this project, we are going to create a model that efficiently classifies traffic signs images and learn to identify the most appropriate features from images on its own. While using deep neural networks methods the model will require a large dataset and a large number of matrix multiplication operations which requires more computer vision work. To tackle this problem we are using a Convolutional Neural Network (CNN) because it has been observed that CNN is more efficient and faster than a regular deep neural network for problems related to computer vision and images. Convolutional Nets models are easy and faster to train on images compared to the traditional models. To train and test the model we are going to use the German Traffic Sign Dataset (GTSRB) which contains more than 50,000 traffic sign images which are divided into 43 classes in total using python 3. This traffic sign dataset is enough which will help us to get results more accurately and to train the model [4].

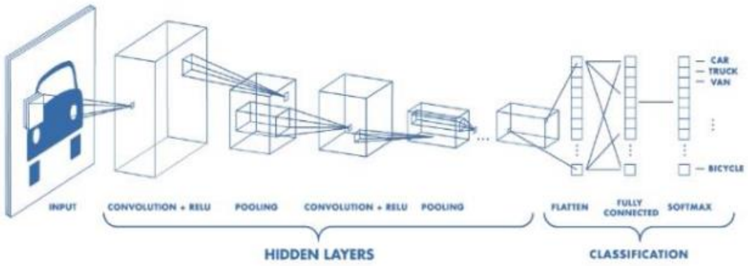


Fig. CNN Model

Steps Involved –
Getting Data
Download the German traffic sign dataset.
We are aiming to use libraries like OpenCV, PyTorch, Numpy, Pandas, OS.
Numpy library is use to calculate summary statistics of 0074he traffic signs data set and for multi-dimensional array, matrices multiplication operations and Pandas version 1.x is used to for reading and writing CSV file.

Dataset exploration, preprocessing and visualization:
First, we’ll check the dimension of all the images captured by the camera in real-time so that we can process the images into having similar dimensions and stored them in a micro SD card. Due to images having a varying range of dimensions which ranges from 16*16*3 to 128*128*3 hence cannot be passed directly to the CNN model. Firstly we need to flip the captured image on the y-axis to get the real image and downscale the image using interpolation. We need to decide the dimension which is in between and save the accurate image data to reduce more compression of the image. . So we have decided to downscale every image to 32 x 32 x 3 dimensions. Next, we will convert these

images to augmented images which will help our model to find exact features in the images. The RGB image is converted into HSV and find only HSV masking is done on that image Hence preprocessing is an important step as it reduces execution time and finds the exact region of interest by ROI (region of extraction). Then after preprocessing we will get the downscaled image. This downscaled image is converted into monochrome (either black or white-colored image). Next, we will pass this image into CNN architecture as the input image.

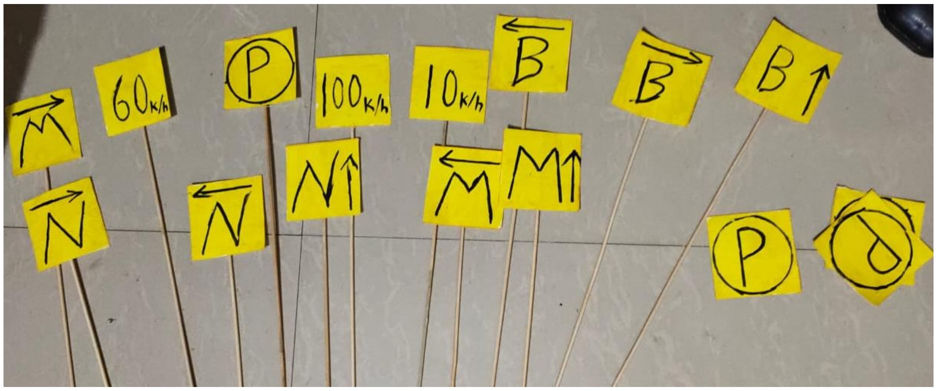


Fig. Signboards for testing

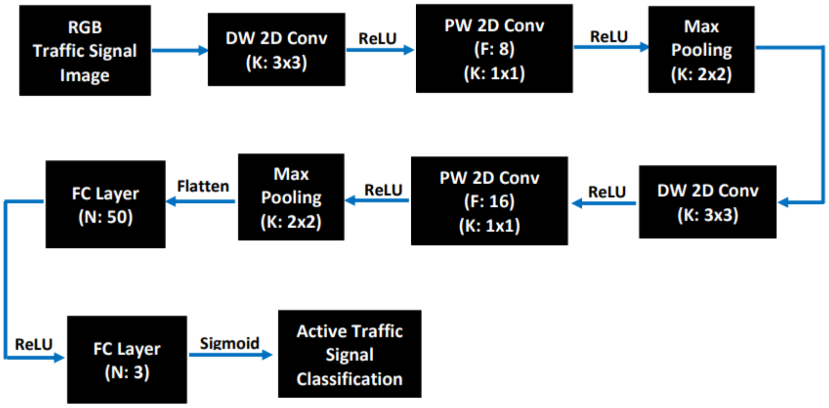


Fig. Traffic signboard detection flowchart

**Model Architecture**

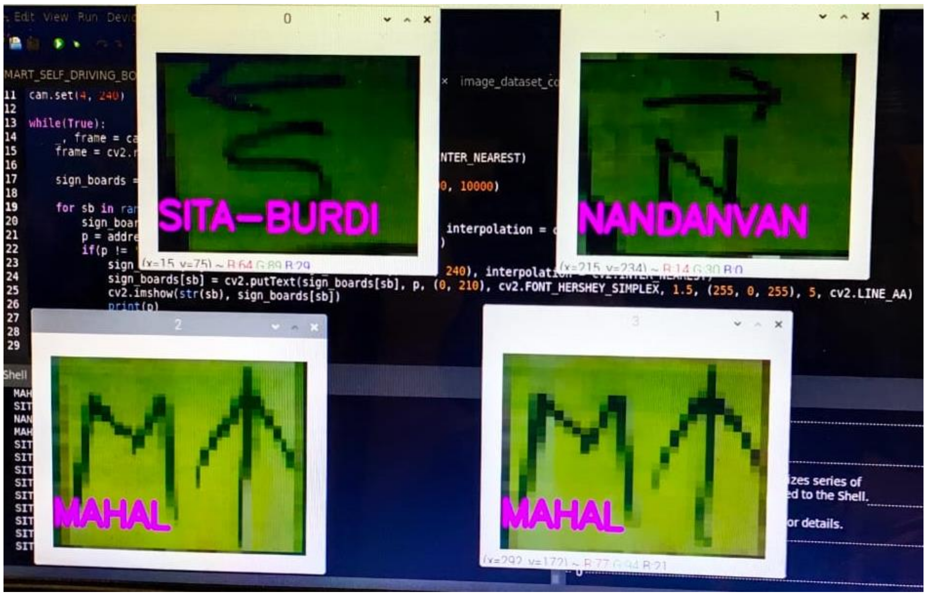
The CNN architecture consists of three types of layers: Convolutional Layer, Max Pooling Layer, and Fully-Connected Layer.

1. INPUT layer consists of a 3-D array of pixel values of the image.

2. DW 2D CONV layer will compute the dot product between the kernel and sub-array of an input image the same size as a kernel. Then it will aggregate all the values resulting from the dot product and this will be the single pixel value of an output image. This process is repeated till the whole image is covered. In this process, we will get a single feature map.

3. RELU (Rectified Linear Unit)layer will apply an activation function max (0, x) on all the pixel values of an output image which skip all negative values and we will get a positive feature map of kernel size 1*1 and 16 feature maps.

4. POOL layer is having a 2*2 kernel size. POOL layer will perform down sampling along the width and height of an image resulting in reducing the dimension of an image by a factor of 2.
5. Next we will pass this image to DW 2D CONV, PW 2D CONV, max pooling. The output of these layers is going to flatten to convert multidimensional images into 1D arrays.
6. Fully-Connected layer with 50 hidden neurons undergoes RELU activation again and get 10 neurons. Finally we are passing it to sigmoid activation which will give different signboard classifications.
7. In this way, CNN transform the original image layer by layer from the original pixel values to the final class values. Parameters at FC and Convolutional layer are trained with the help of gradient descent optimizer [4].



```
cam.set(4, 240)

while(True):
    _, frame = ca
    frame = cv2.r                                    NTER_NEAREST)

    sign_boards =                                    0, 10000)

    for sb in ran
        sign_boar                                    interpolation = c
        p = addre                                    )
        if(p != '
            sign_                                    240), interpolati
            sign_boards[sb] = cv2.putText(sign_boards[sb], p, (0, 210), cv2.FONT_HERSHEY_SIMPLEX, 1.5, (255, 0, 255), 5, cv2.LINE_AA)
            cv2.imshow(str(sb), sign_boards[sb])
        print(p)
```

3.2 Automatic Lane detection and tracking: Lane detection, which will capture one frame from the camera, that frame goes on process of interpolation and clipping then it will convert RGB image into a grayscale image. In that grayscale image, we will add some Gaussian blur. After blurring this image, we will apply a canny egde detection algorithm, that is going to extract all real edges of images (it will view only edges). Once we get that image, apply ROI (region of interest) extraction e.g. lane detection, region of interest will be below part of the image or the part where the road will present.

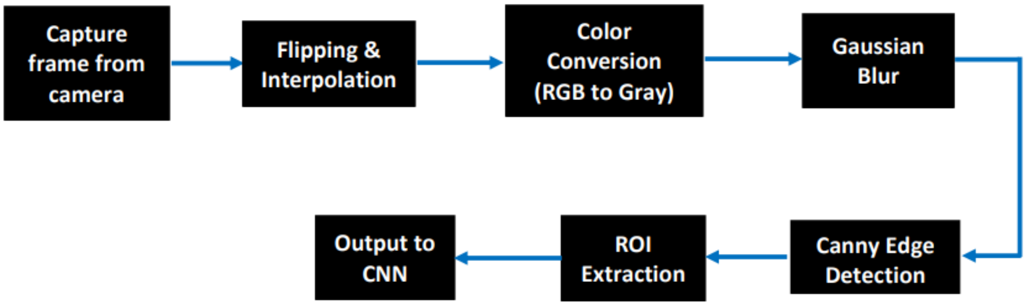
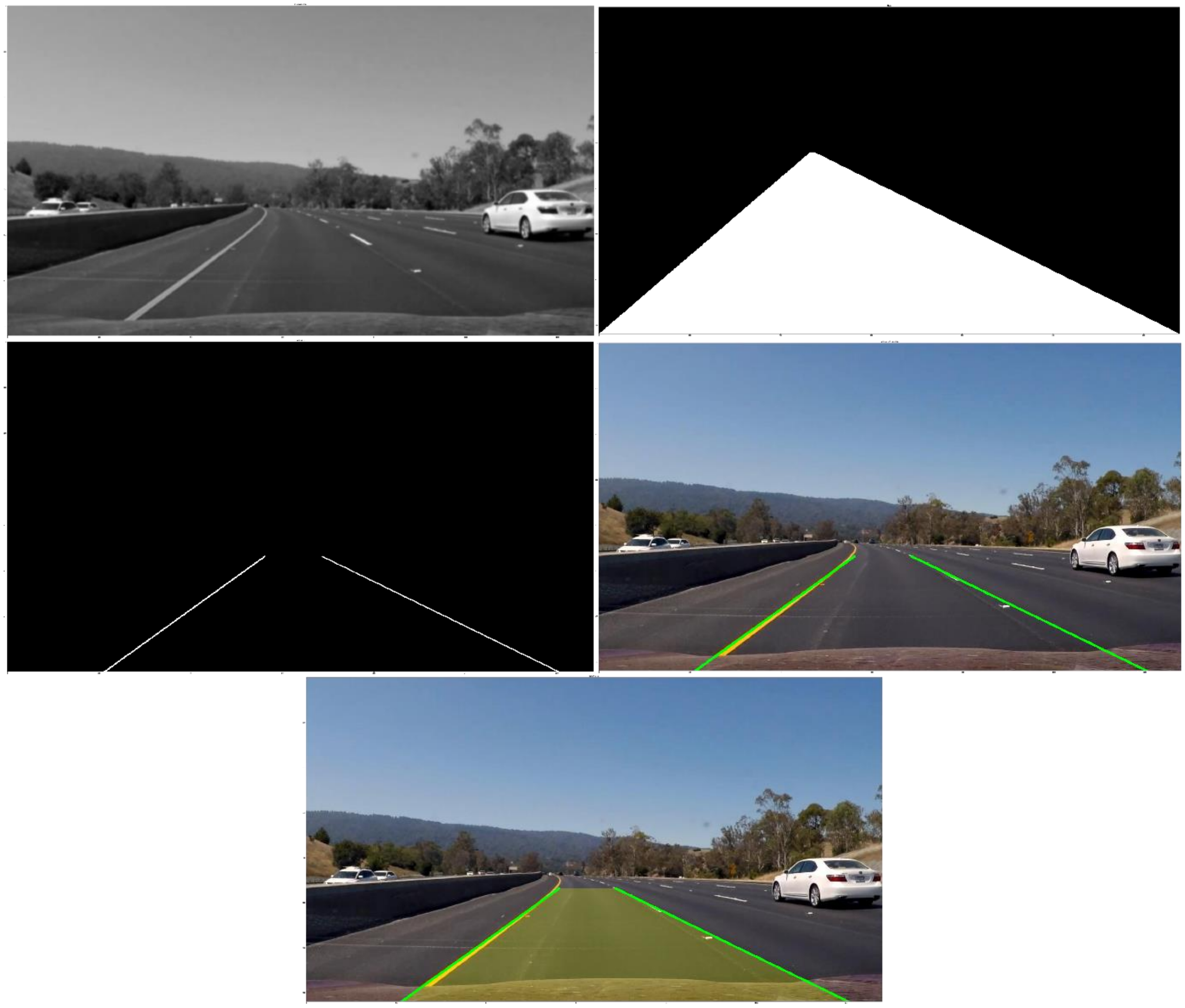


Fig. Lane Detection flowchart

3.2.1 Lane Following: As already detected lane/road now wants to follow this. Robots follow the lane automatically. It is having a controller Raspberry Pia with computer vision, it will be able to control the motor and its speed can move a robot in any direction. In a neural network, the first layer is depth-wise convolutional in this layer, our input image will convert into feature maps. For example, apply the grayscale image as input, it is having only a single channel. If grayscale passes into a depth convolutional network that neural network only has a single filter. Whatever output comes, applying some activation functions like rectifiers linear unit, which will skip all the negative values. Input channels will pass through 16 filters. Then we downscale using max pulling. Once downscaling is done again it will pass this to depth-wise max pulling. Once get the last feature map from the last pulling layer we are going to flatten them. This flatting function converts a multi-D feature map array into one D array. So flatten array fed into the fully connected layer, the first fully converted layer which is having input equal to flattened array and no of output hidden neurons are 100. The first fully connected layer is going to get activated using some activation function and the second fully connected layer is the second hidden layer which is having an output of 3 neurons. This is the output layer, having 3 neurons and those neurons finally passing them through some sigmoid activation which will convert in range 0 and 1. Whatever classification comes from this neuron will move our robot accordingly.



3.2.2 Automatic traffic light monitoring and control: A high-speed camera that operates at 500 fps on a car because it would capture five images during a single period of lamp blinking. The proposed detection system consists of six modules that include loading, band-pass filter, binarization, buffering, detection, and classification.
The binarization module first estimates the state of the traffic light dynamics, which includes the blinking amplitude, offset, and phase. It uses the Kalman filter for state estimation. Subsequently, it determines an appropriate threshold for binarizing the filtered image based on the estimated state and finally binarizes the filtered image. The buffering module relays the image, which has stronger signals compared to the previous images. This is because recognizing colors and areas from an image with non-maximum brightness is difficult. The detection module extracts the contours from the peak binarized image. Further, the size and shape are used to exclude candidates to prevent false detections. The classification module classifies the lamp color using the contours and RGB images

3.2.3 Band-Pass Filter Module: This module enhances the area blinking at a specific frequency by applying the band-pass filter to the gray-scaled image over time. The used of the IIR filter, which have steep frequency characteristics even in small dimensions, to reduce the amount of computation required for real-time processing.

3.2.4 Binarization Module: For the investigation, binarizing the image that was applied to the band-pass filter was necessary to efficiently extract the blinking area. There are two methods of binarization: one is to use a common threshold for the entire image, and the other is to adaptively use the variable threshold for each pixel according to the surrounding pixels.
In particular, the appearance of the traffic light on the image captured during daytime and night differs significantly. The state of traffic light was estimated. It included the amplitude, offset, and phase of the blinking traffic light. Calculating the appropriate threshold removed the disturbance while retaining the traffic lights in the image

3.2.5 Buffering Module: This module performs two tasks:
(1) It seeks the local maximum image from the last images, thus simplifying processing for subsequent modules.
(2) It compensates for the phase delay of the binarized image to aid its synchronization with the RGB image.

## IV. REFERENCE

[1] Bernardo Marques, Raphaël Carvalho, Paulo Dias, Miguel Oliveira, Carlos Ferreira, Beatriz Sousa Santos , *Evaluating and enhancing Google Tango localization in indoor environments using fiducial markers , 18th IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC) April 25-27 2018, Torres Vedras, Portugal.*

[2] Wuttichai Vijitkunsawat, Peerasak Chantngarm , *Comparison of Machine Learning Algorithm's on Self-Driving Car Navigation using Nvidia Jetson Nano , 2020 17th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON).*

[3] Mrs. Deepali Patil, Ashika Poojari, Jayesh Choudhary, Siddhath Gaglani , *CNN based Traffic Sign Detection and Recognition on Real Time Video, International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181 Published by, www.ijert.org NTASU - 2020 Conference Proceedings.*

[4] Nazmul Hasan, Tanvir Anzum, and Nusrat Jahan, *Traffic Sign Recognition System (TSRS): SVM and Convolutional Neural Network , Conference Paper · September 2020 DOI: 10.1007/978-981-15-7345-3_6.*

[5] Yanlei Gu, Woranipit Chidsin, Igor Goncharenko , *AR-based Navigation Using Hybrid Map , 2021 IEEE 3rd Global Conference on Life Sciences and Technologies (LifeTech 2021).*

[6] Chirag Sharma, S. Bharathiraja, G. Anusooya , *Self Driving Car using Deep Learning Technique , International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181 org Vol. 9 Issue 06, June-2020248.*

[7] Tej Kurani, Nidhip Kathiriya, Uday Mistry, Prof. Lukesh Kadu, Prof. Harish Motekar , *Self Driving Car Using Machine Learning , International Research Journal of Engineering and Technology (IRJET) Volume: 07 Issue: 05 | May 2020.*

[8] Ahmed Hechri, Mtibaa Abdellatif , *Lanes and Road Signs Recognition for Driver Assistance System , November 2011 International Journal of Computer Science Issues Vol. 8(Issue 6, No 1).*

[9] Deepika Manimaran, Maria Anu, Christina Steffi.F , *Integration of Indoor and Outdoor Voice Based Navigation Detection using Light based Communication (Lifi) & IoT , Proceedings of the Fifth International Conference on Computing Methodologies and Communication (ICCMC 2021) IEEE Xplore Part Number: CFP21K25-ART.*

[10] AnuraagVelamatia, Gopichand Gb , *Traffic Sign Classification Using Convolutional Neural Networks and Computer Vision , Turkish Journal of Computer and Mathematics Education Vol.12 No.3(2021), 4244-4250.*

[11] Nitin Kanagaraj, David Hicks, Ayush Goyal, Sanju Tiwari, Ghanapriya Singh , *Deep learning using computer vision in self driving cars for lane and traffic sign detection, International Journal of Systems Assurance Engineering and Management · May 2021.*

[12] Wenhong Zhu, Fuqiang Liu, Zhipeng Li, Xinhong Wang, Shanshan Zhang , *A Vision Based Lane Detection and Tracking Algorithm in Automatic Drive , 2008 IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application.*