IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

Artificial Intelligence Based Self Driving Car Using Raspberry AI

N.Nivetha[1]

[1]**Assistant professor**

Dept of Electronics and Communication Engineering

Unnamalai Institute of Technology

Tamil Nadu , India

Abstract - A self-driving car is a car which is capable of sensing the environment and moves safely without human inputs. A need for automated vehicles is increasing every day due to rapid growth in road traffic and increased number of accidents. In developed as well developing countries the traffic is the biggest problem everyone is facing now. This traffic not only waste our time but also it also increase the chance for accidents. About 0.5 million people dies in road accident every year in India and so many more peoples were injured. About 94 % of accidents occurred is due the human errors such as carelessness, negligence and mental condition of the driver. The main goal of this project is to replace the error prone human with error free machines controlled by artificial intelligence. These self-driving are capable making its own decision in highly complex scenarios. Artificial intelligence is the back bone of this system. These machines not only senses the obstacles but also it can learn the environment, and can derive the path without GPS. These machines doesn't use GPS or Google maps to derive its route but it will study the environment and remember the places and route as we human do. To achieve this we use machine learning Techniques where machine can learn, improvise and exhibit its actions in a sensible way. If more number of driverless cars were employed it can reduce thetraffic and provide smooth flow of traffic.

Keywords: automation, self driving, artificial intelligence, GPS, machine learning

1. INTRODUCTION

Modern age is the age of technology and innovation. Rapid advances in technology have made it possible for humans to make their work easier and faster in many ways. Scientists and technologists are always busy in working to innovate and improve new technology products to facilitate and make human life easier not only to save time and effort but also to ensure safety. Driverless car is one out of many such innovations intended to help human beings in their daily life. Work on car automation started as early as in 1920s. More encouraging results in this development were seen in 1950s, which encouraged the scientists to keep working and first truly automatic cars were seen in 1980s ("History of autonomous car,". However, According to Marks research, meaningful developments in driverless cars were exhibited by US Defence Advance Research Project Agency (ARPA) by organizing competitions of such cars in desserts in 2004 and 2005 and in urban area in 2007. Many research organizations are involved in the development of driverless cars, however, Google is far ahead of others in its research on this particular technology.

1.2 NEED OF THE PROJECT

Driverless cars bring many advantages to human beings over the traditional cars. Google aimed at changing the car technology to bring about driverless cars into the market in order to reduce number of road accidents, save people's time, as well as carbon emissions. According to research 10.8 million accidents per year happen only in USA. These accidents not only cause more than 36000 deaths but also around $300 billion costs annually. One major benefit of driverless cars is that they are programmable and they don't possess unpredictable behavior like humans do. The high customizability of the program of the car, these can be easily adjusted to follow local traffic laws, and obstacles that it encounters. For example, Google's car is programmed to follow traffic rules and signs and responds to signs and obstacles faster than humans can do to help avoid accidents and thus save lives. The inches forward feature of Google Car is very efficient to indicate to other vehicles about its intentions at crossings and intersections. These features make driverless car safer than traditional cars. Google cars estimate to be able to not only save 30,000 lives but also $2.7 billions of cost and more than 2 million injuries. Bureau of Transportation 2009 statistics cited that Google car can reduce the average commuting time of Americans from fifty two to only five minutes daily. The suggested approach is that being intelligent, autonomous cars can drive close to each other with less errors and hence can efficiently utilize road space on freeways. In addition, to saving on commuting time, driverless cars can avoid congestion thus saving 4.8 billion hours and 1.9 billion gallons of fuel annually (US Energy Information Administration, 2012). Driverless cars make carpooling an easy and realistic option due to its technological capabilities. These cars are also capable of self-parking and retrieving thus saving time in these activities. The intelligent map techniques and more interaction with centralized systems can also help not only to detect or forecast congestion routes but also to select and use shortest routes between destinations thus reducing fuel consumption even more. All the car-sharing and optimized route selection is estimated to cut the gas consumption by 80 percent. This will not only reduce the number of self-owned cars on the road but will also help save energy to meet more demands in future thus adding to the economy of the country

2. LITERATURE SURVEY

Despite the rapid developments and technological advances in the accuracy and reliability of driverless cars, they are still not in common use and have not been presented for public transport yet. All the carmakers have been in a race, busy in testing their prototypes for quick delivery into the markets. Google has been ahead of all in this race. By 2013, Google car completed around 200,000 miles of accident free driverless test drives (Poczter & Jankovic, 2013). According to Mui (2015), more than 20 Google driverless cars have completed test drives of more than 1.7 million miles out of which around one million miles test drives were in driverless mode. To speed up entry in to the market, Google is driving its autonomous cars around 10,000 miles per week in real environments, without controlled environments. Google is also doing simulated driving of its driverless cars for around 3 million miles a day. It is expected that Google's driverless cars will soon be into the market for public use.

3.PROPOSED SYSTEM

Driverless cars are standard passenger cars with additional capabilities of replacing the driver by an intelligent autonomous system to run the car. In such cars, GPS receivers and mapping technology are installed to navigate paths and destinations. Radar system is used to detect obstacles that the driverless car may encounter. A laser ranging system scans the outside environment in three dimensions. The car also comes with a video camera to identify various objects like signs, lights, humans, and other cars. The system of the car uses all the information to decide the behaviour of the car in a particular situation (Waldrop, 2015). Thus, driverless cars are capable of performing all the functions of a human driver through automatic processing units.

4. SYSTEM REQUIREMENTS

4.1HARDWARE COMPONENTS

- Raspberry pi
- Pi camera
- Ultrasonic sensors

- Lm298 motor module
- Geared motor

4.2 SOFTWARE COMPONENTS

- Python
- Libraries
- GPIOZERO
- CV2 (OPENCV)
- PICAMERA
- NUMPY

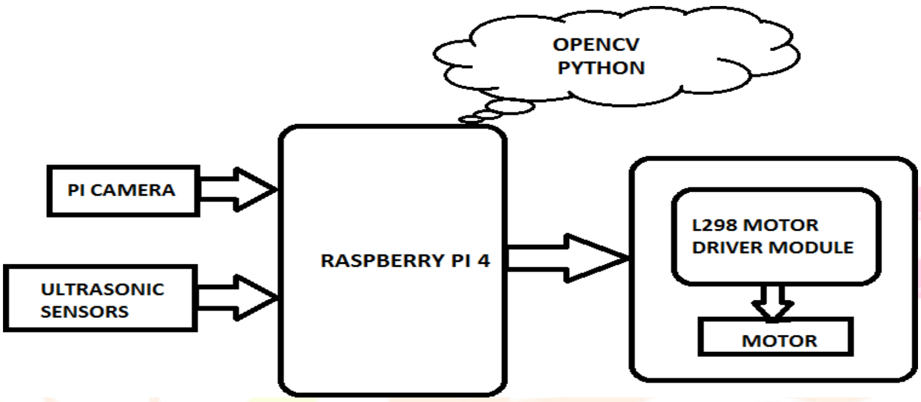


Fig 4.1 block diagram of the proposed system

4.3 MODULE 1- RASPBERRY PI 4

Raspberry Pi is a series of small single-board computers developed by the Raspberry Pi Foundation in association with Broadcom. The Raspberry Pi project originally leaned towards the promotion of teaching basic computer science in schools and in developing countries. The original model became more popular than anticipated, selling outside its target market for uses such as robotics. It is widely used in many areas, such as for weather monitoring, because of its low cost, modularity, and open design. It is typically used by computer and electronic hobbyists, due to its adoption of HDMI and USB devices.

HARDWARE	INTERFACE	
Quad core 64-bit ARM-Cortex A72 running at 1.5 GHz	802.11 b/g/n/ac Wireless LAN	1x Raspberry Pi camera port (2-lane MIPI CSI)
1, 2 and 4 Gigabyte LPDDR4 RAM options	Bluetooth 5.0 with BLE	1x Raspberry Pi display port (2-lane MIPI DSI)
H.265 (HEVC) hardware decode (up to 4 Kp 60)	1x SD Card	
H.264 hardware decode (up to 1080p60)	2x micro-HDMI ports supporting dual displays up to 4Kp60 resolution	
Video Core VI 3D Graphics	2x USB2 ports	
Supports dual HDMI display output up to 4 Kp 60	2x USB3 ports	
	1x Gigabit Ethernet port (supports PoE with add- on PoE HAT)	

4.4 SOFTWARE REQUIREMENTS

- ARMv8 Instruction Set
- Mature Linux software stack

4.5 MECHANICAL SPECIFICATIONS

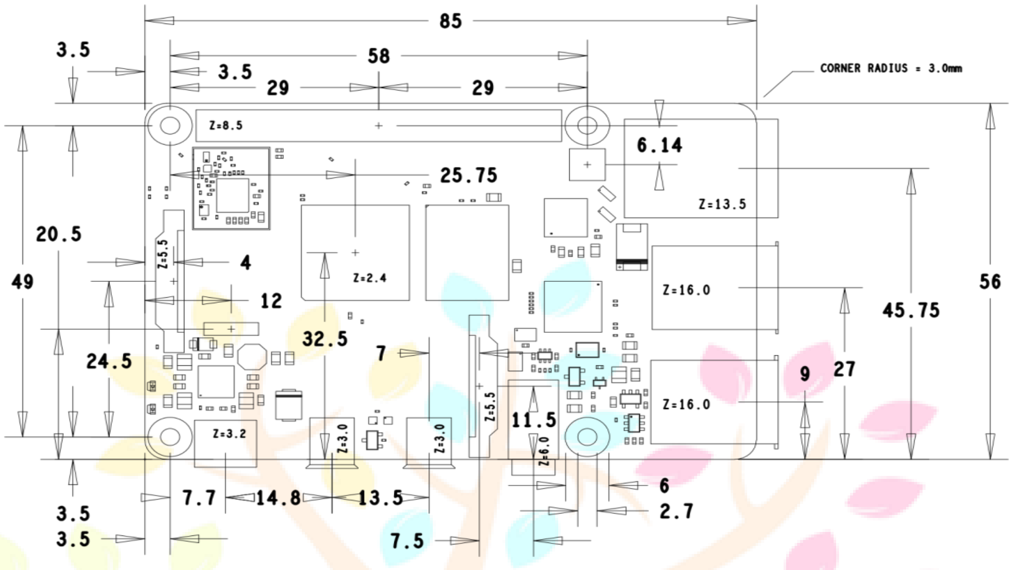


FIG 4.2 Mechanical Dimensions

4.6 POWER REQUIREMENTS

The Pi4B requires a good quality USB-C power supply capable of delivering 5V at 3A. If attached downstream USB devices consume less than 500mA, a 5V, 2.5A supply may be used.

5. PERIPHERALS

5.1 GPIO INTERFACE

The Pi4B makes 28 BCM2711 GPIOs available via a standard Raspberry Pi 40- pin header. This header is backwards compatible with all previous Raspberry Pi boards with a 40-way header. As well as being able to be used as straightforward software controlled input and output (with programmable pulls), GPIO pins can be switched (multiplexed) into various other modes backed by dedicated peripheral blocks such as I2C, UART and SPI. In addition to the standard peripheral options found on legacy Pis, extra I2C, UART and SPI peripherals have been added to the BCM2711 chip and are available as further mux options on the Pi4. This gives users much more flexibility when attaching add-on hardware as compared to older models.

GPIO	Pull	ALT0	ALT1	ALT2	ALT3	ALT4	ALT5
0	High	SDA0	SA5	PCLK	SPI3 CE0 N	TXD2	SDA6
1	High	SCL0	SA4	DE	SPI3 MISO	RXD2	SCL6
2	High	SDA1	SA3	LCD VSYNC	SPI3 MOSI	CTS2	SDA3
3	High	SCL1	SA2	LCD HSYNC	SPI3 SCLK	RTS2	SCL3
4	High	GPCLK0	SA1	DPI D0	SPI4 CE0 N	TXD3	SDA3
5	High	GPCLK1	SA0	DPI D1	SPI4 MISO	RXD3	SCL3
6	High	GPCLK2	SOE N	DPI D2	SPI4 MOSI	CTS3	SDA4
7	High	SPI0 CE1 N	SWE N	DPI D3	SPI4 SCLK	RTS3	SCL4
8	High	SPI0 CE0 N	SD 0	DPI D4	-	TXD4	SDA4
9	Low	SPI0 MISO	SD 1	DPI D5	-	RXD4	SCL4
10	Low	SPI0 MOSI	SD 2	DPI D6	-	CTS4	SDA 5
11	Low	SPI0 SCLK	SD3	DPI D7	-	RTS4	SCL5
12	Low	PWM0	SD4	DPI D8	SPI5 CE0 N	TXD5	SDA5
13	Low	PWM1	SD5	DPI D9	SPI5 MISO	RXD5	SCL5
14	Low	TXD0	SD6	DPI D10	SPI5 MOSI	CTS5	TXD1
15	Low	RXD0	SD7	DPI D11	SPI5 SCLK	RTS5	RXD1
16	Low	FL0 _	SD8	DPI D12	CTS0	SPI1 CE2 N	CTS1
17	Low	FL1 _	SD9	DPI D13	RTS0	SPI1 CE1 N	RTS1
18	Low	PCM-CLK	SD10	DPI D14	SPI6 CE0 N	SPI1 CE0 N	PWM0
19	Low	PCM FS	SD11	DPI D15	SPI6 MISO	SPI1 MISO	PWM1
20	Low	PCM DIN	SD12	DPI D16	SPI6 MOSI	SPI1 MOSI	GPCLK0
21	Low	PCM DOUT	SD13	DPI D17	SPI6 SCLK	SPI1 SCLK	GPCLK1
22	Low	SD0 CLK	SD14	DPI D18	SD1 CLK	ARM TRST	SDA6
23	Low	SD0 CMD	SD15	DPI D19	SD1 CMD	ARM RTCK	SCL6
24	Low	SD0 DAT0	SD16	DPI D20	SD1 DAT0	ARM TDO	SPI3 CE1 N
25	Low	SD0 DAT1	SD17	DPI D21	SD1 DAT1	ARM TCK	SPI4 CE1 N
26	Low	SD0 DAT2	TE0	DPI D22	SD1 DAT2	ARM TDI	SPI5 CE1 N
27	Low	SD0 DAT3	TE1	DPI D23	SD1 DAT3	ARM TMS	SPI6 CE1 N

Table 5.1: Raspberry Pi 4 GPIO Alternate Functions

5.3 DISPLAY PARALLEL INTERFACE (DPI)

A standard parallel RGB (DPI) interface is available the GPIOs. This up-to-24- bit parallel interface can support a secondary display.

5.4 SD/SDIO INTERFACE

The Pi4B has a dedicated SD card socket which suports 1.8V, DDR50 mode (at a peak bandwidth of 50 Megabytes / sec). In addition, a legacy SDIO interface isavailable on the GPIO pins.

5.5 CAMERA AND DISPLAY INTERFACES

The Pi4B has 1x Raspberry Pi 2-lane MIPI CSI Camera and 1x Raspberry Pi 2- lane MIPI DSI Display connector. These connectors are backwards compatible with legacy Raspberry Pi boards, and support all of the available Raspberry Pi camera and display peripherals.

5.6 USB

The Pi4B has 2x USB2 and 2x USB3 type-A sockets. Downstream USB current is limited to approximately 1.1A in aggregate over the four sockets.

5.7 HDMI

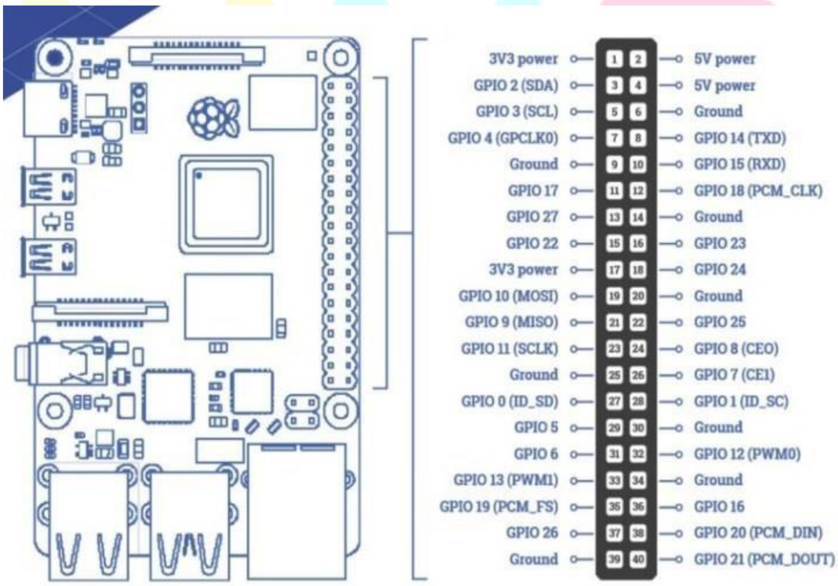
The Pi4B has 2x micro-HDMI ports, both of which support CEC and HDMI 2.0with resolutions up to 4 Kp 60.

5.8AUDIO AND COMPOSITE (TV OUT)

The Pi4B supports near-CD-quality analogue audio output and composite TV- output via a 4-ring TRS ’A/V’ jack.The analog audio output can drive 32 Ohm headphones directly.

5.9 TEMPERATURE RANGE AND THERMALS

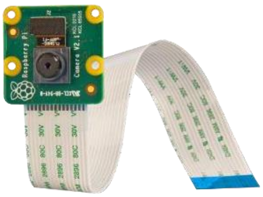
The recommended ambient operating temperature range is 0 to 50 degrees Celcius. To reduce thermal output when idling or under light load, the Pi4B reduces the CPU clock speed and voltage. During heavier load the speed and voltage (and hence thermal output) are increased. The internal governor will throttle back both the CPU speed and voltage to make sure the CPU temperature never exceeds 85 degrees C. The Pi4B will operate perfectly well without any extra cooling and is designed for sprint performance expecting a light use case on average and ramping up the CPU speed when needed (e.g. when loading a webpage). If a user wishes to load the system continually or operate it at



a high termperature at full performance, further cooling may be needed.

Picture 5.1 Raspberry Pi Pin Diagram

6. PI CAMERA MODULE



Picture 6.1 Pi Camera

6.1 Product description

The Raspberry Pi Camera Modules are official products from the Raspberry Pi Foundation. The original 5-megapixel model was released in 2013, and an 8- megapixel Camera Module v2 was released in 2016. For both iterations, there are visible light and infrared versions. A 12-megapixel High Quality Camera was released in 2020. There is no infrared version of the HQ Camera,however the IR Filter can be removed if required.

6.1.1 Hardware features

Available	Implemented
Chief ray angle correction	Yes
Global and rolling shutter	Rolling shutter
Automatic exposure control (AEC)	No - done by ISP instead
Automatic white balance (AWB)	No - done by ISP instead
Automatic black level calibration(ABLC)	No - done by ISP instead
Automatic 50/60 Hz luminancedetection	No - done by ISP instead

Frame rate up to 120 fps	Max 90fps. Limitations on frame size for the higher frame rates (VGA only for above 47fps)
AEC/AGC 16-zonesize/position/weight control	No - done by ISP instead
Mirror and flip	Yes
Cropping	No - done by ISP instead (except1080p mode)
Lens correction	No - done by ISP instead
Defective pixel cancelling	No - done by ISP instead
10-bit RAW RGB data	Yes - format conversions available via GPU
Support for LED and flash strobemode	LED flash
Support for internal and external frame synchronisation for frame exposure mode	No
Support for 2 × 2 binning for better SNR in low light conditions	Anything output res below 1296 x 976 will use the 2 x 2 binned mode
Support for horizontal and verticalsub-sampling	Yes, via binning and skipping
On-chip phase lock loop (PLL)	Yes
Standard serial SCCB interface	Yes
Digital video port (DVP) paralleloutput interface	No
MIPI interface (two lanes)	Yes
32 bytes of embedded one-time programmable (OTP) memory	No
Embedded 1.5V regulator for corepower	Yes

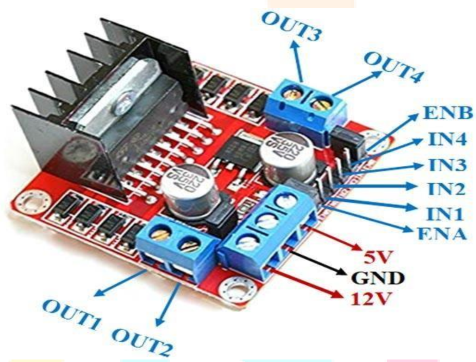
6.1.2Software features.

PARAMETER	SETUP
Picture formats	JPEG (accelerated), JPEG + RAW, GIF, BMP,PNG, YUV420, RGB888
Video formats	raw h.264 (accelerated)

Effects	negative, solarise, posterize, whiteboard, blackboard, sketch, denoise, emboss, oilpaint, hatch, gpen, pastel, watercolour, film, blur, saturation
Exposuremodes	auto, night, nightpreview, backlight, spotlight, sports, snow, beach, verylong, fixedfps, antishake,fireworks
Metering modes	average, spot, backlit, matrix
Automatic white balance modes	off, auto, sun, cloud, shade, tungsten, fluorescent, incandescent, flash, horizon
Triggers	Keypress, UNIX signal, timeout
Extra modes	demo, burst/timelapse, circular buffer, video with motion vectors, segmented video, live preview on 3D models

6.2 L298N MOTOR DRIVER MODULE

This L298N Motor Driver Module is a high power motor driver module for driving DC and Stepper Motors. This module consists of an L298 motor driver IC and a 78M05 5V regulator. L298N Module can control up to 4 DC motors, or 2 DC motors with directional and speed control. The L298N Motor Driver module consists of an L298 Motor Driver IC, 78M05 Voltage Regulator, resistors, capacitor, Power LED, 5V jumper in an integrated circuit.



Picture 6.2 L298N Motor Driver Module

6.3 L298N MODULE PIN CONFIGURATION:

<table>
<tr><th>Pin Name</th><th>Description</th><th>Module Features & Specifications</th></tr>
<tr><td>IN1 & IN2</td><td>Motor A input pins. Used to control the spinning direction of Motor A</td><td rowspan="7">Driver Model: L298N 2A
Driver Chip: Double H Bridge L298N
Motor Supply Voltage (Maximum): 46V
Motor Supply Current (Maximum): 2A
Logic Voltage: 5V
Driver Voltage: 5-35V
Driver Current:2A
Logical Current:0-36mA
Maximum Power (W): 25W
Current Sense for each motor
Heatsink for better performance</td></tr>
<tr><td>IN3 & IN4</td><td>Motor B input pins. Used to control the spinning direction of Motor B</td></tr>
<tr><td>ENA</td><td>Enables PWM signal for Motor A</td></tr>
<tr><td>ENB</td><td>Enables PWM signal for Motor B</td></tr>
<tr><td>OUT1 & OUT2</td><td>Output pins of Motor A</td></tr>
<tr><td>OUT3 & OUT4</td><td>Output pins of Motor B</td></tr>
<tr><td>12V</td><td>12V input from DC power Source</td></tr>
</table>

5V	Supplies power for the switching logic circuitry inside L298N IC	Power-On LED indicator
GND	Ground pin	

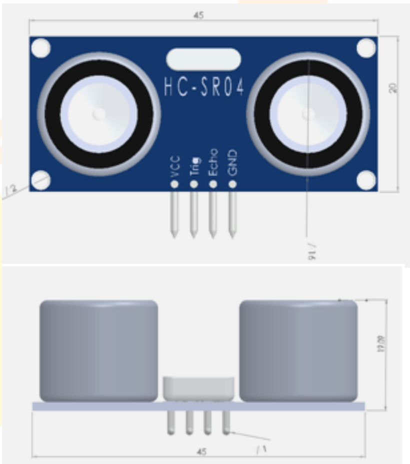
6.4 ULTRASONIC SENSOR

6.4.1 PIN CONFIGURATION

Pin Number	Pin Name	Description
1	Vcc	The Vcc pin powers the sensor, typically with +5V
2	Trigger	Trigger pin is an Input pin. This pin has to be kept high for 10us to initia measurement by sending US wave.
3	Echo	Echo pin is an Output pin. This pin goes high for a period of time which be equal to the time taken for the US wave to return back to the sensor.
4	Ground	This pin is connected to the Ground of the system.

6.4.2 HC-SR04 SENSOR FEATURES

- Operating voltage: +5V
- Theoretical Measuring Distance: 2cm to 450cm
- Practical Measuring Distance: 2cm to 80cm
- Accuracy: 3mm
- Measuring angle covered: <15°
- Operating Current: <15mA
- Operating Frequency: 40Hz



Picture 6.1 2-D Model Of Ultrasonic Sensor

7. MACHINE LEARNING FOR TRAFFIC ASSESMENT

This paper concentrates on how Image processing can be used in vehicles to drive the automotive industry to completely autonomous and high security pathways. A real time embedded system environment is inevitable in an automotive application. Also, the scale of the industry is very high so the solutions should be cost efficient, fast and reliable. This paper intends to highlight these key points.

7.1 Autonomous Vehicle Control Using Image Processing

In autonomous vehicles, the driving commands from a human driver are replaced by a controller or a microcomputer system that generates these commands from the information it gets as it's input. Since this paper deals with the applications of image processing in autonomous control of a vehicle, the input given to the microcomputer system is the visual information obtained from a camera mounted on the vehicle. This section explains in detail, some of the important factors in Autonomous vehicles such as Lane Detection, Traffic Sign Detection, Speed Bump Detection, Steer By Wire System etc., which uses the processing of received image inputs and the algorithms used in them. Lane Detection represents a robust and real time detection of road lane markers using the concept of Hough transform in which the edge detection is implemented using the Canny edge detection technique and Spiking neural network technique. Traffic Sign Detection includes the recognition of road traffic signs in which, the concept of Polynomial approximation of digital curves is used in the detection module. Speed Bump Detection represents the detection and recognition of speed bumps present in the road which alerts the vehicle to control the speed automatically and Steer-By- Wire system represents an autonomous steering system using an Electronic Power Steering (EPS) module.

7.2 Lane Detection

Lane detection is one of the main part in the self-driving car algorithm development. On board cameras are kept in and around the cars to capture images of road and surrounding of the car in real time [1]. When the vehicle appears to deviate from the lane or vehicle safety distance is too small, it can timely alert the driver to avoid dangerous situations. The basic concept of lane detection is that, from the image of the road, the on- board controller should understand the limits of the lane and should warn the driver when the vehicle is moving closer to the lanes. In an autonomous car, lane detection is important to keep the vehicle in the middle of the lane, at all-time, other than while changing lanes. Lane departure warning systems have already crawled into most of the high-end passenger cars currently in market.

7.2.1Lane Algorithm Steps

- Select the ROI (Region of Interest)
- Image Preprocessing (gray range/image noise subtraction)
- Get the edge information from the image (edge detection)
- Hough Transform (or other algorithms) to decide the lane markings

Step 1 select the ROI

The images collected by the on-board camera are color images. Each pixel in the image is made up of R, G, and B three color components, which contains large amount of information. Processing these images directly makes the algorithm consume a lot of time. A better idea for this problem is to select the region of interest (ROI) from the original image by concentrating on just the region of the image that interests us namely the region where the lane lines are generally present. Processing on only the ROI can greatly reduce the time of algorithm and improve the running speed.

Step 2: Image Preprocessing

Most of the road images have a lot of noise associated. So, before we do any image processing steps we need to remove those noises. This is typically done through Image Preprocessing. Image preprocessing includes grayscale conversion of color image, gray stretch, median filter to eliminate the image noise and other interference information. Gray stretch can increase the contrast between the lane and the road, which makes the lane lines more prominent. Equation (1) represents the function which is to be applied to an RGB image to convert it to Gray Scale.

$$L(x,y) = 0.21\,R(x,y) + 0.72G(x,y) + 0.07\,B(x,y) \qquad (1)$$

Where

R - Red component of the image

G - Green component of the image

B - Blue component of the image

X,y - position of a pixel

The method of image filtering includes the frequency domain filtering and the spatial domain filtering [2]. Spatial domain filtering is simpler and faster than the frequency domain filtering. Spatial domain filtering can remove the salt and pepper noise from the original image and preserve the edge details of the image. Its main principle is to use the middle value of every pixel in the neighborhood of one pixel instead of current pixel value.

Step 3: edge detection on the output of the preprocessing.

It is basically to detect the lines around the objects in the images. One of the common methods of edge detection is called Canny Edge Detection introduced by John F Canny, University of California, Berkeley in 1986. It basically uses multiple steps including Gaussian filters, intensity gradient changes to determineIn recent researches, one of the main goals are to develop higher efficient edge detection algorithm for better detecting edges from varying image quality. For that purpose, an alternative approach to edge detection called Spiking Neural Network [1] is used, which is claimed to be a much efficient method than cannyedge detector..

Step 4: Hough Transform

Hough transform uses simple mathematical model of a straight line in a two- coordinate system, to predict straight lines or other simple shapes from an image.Equation (2) represents the basis for performing Hough transform.

$$r = x \cos \theta + y \sin \theta \qquad (2)$$

From the edge detection algorithm, we get a set of points which are considered part of the lane markings, Hough transform is used to generate a line through these points from the image. It connects points in the 2 dimensional polar coordinates of r and θ (theta), and generates multiple lines from each point. After generating all the different combination of lines of r and θ (theta) from each point, there will be one specific r and θ (theta) which intersects which is common to allthe lines generated. This line will pass through all the points of interest. Thus, a line is generated.

Hough transform is a widely-used method in the lane detection. It has very good suppression of noise performance and is not sensitive to the target which is partially occluded and covered in the image. But, because of the complexity of Hough transform, the computation speed is very slow, and the false detection rate is large. It cannot meet the real-time requirements accurately. The detection speed is improved by limiting the polar angle and the polar diameter in Hough transform space as described in Q. Wang [4]. Z.L.Jin [5] detected the lane lines by finding the best threshold of the image. Random sample consensus (RANSAC) algorithm is used to detect the lane lines in J.Guo [6], but it has high wrong rate. J.R.Cai [7] used the peak point inverse transform to extract line segments in Hough transforms space. All the methods mentioned above are modified in the process of Hough transform, but do not consider the appropriate processing methods in the image before applying Hough transform, to improve the detection accuracy and shorten the calculation time. X.Li [1] considered all these discrepancies and integrates a set of image processing techniques and spiking neural network to improve accuracy of the lane detection. Initially, the ROI is set on the image captured from the on-board camera. This can greatly reduce the running time of the algorithm and improving the detection speed; Secondly, a set of image preprocessing techniques is applied to the ROI region, such as transform from RGB image to grayscale, gray stretch and median filtering. Edge detection based on the spiking neural network with biological mechanism is used to extract efficient edge lines. At last, the Hough transform is used to detect the lane. The new ideas are that the ROI preprocessing is used to speed up the detection time and spiking neural network is used to obtain better edge lines to improve the detection accuracy.

7.3 Traffic Sign Detection

Traffic sign detection is key to semi and fully autonomous vehicles. The purpose of the technique is to identify the sign boards on the side of the road and identify what they imply and let the car computer know The system has two components, one is the preprocessing module, which is based on image processing techniques such as color segmentation, threshold technique, Gaussian filter, canny edge detection and contours detection [8]. The second is the

Detection module based on Polynomial Approximation of digital curves technique applied on contours to correctly identify the signboard. It uses a similar technique of converting Image from RGB to Gray to Binary and then finding contours. Based on the shape of the sign board, the algorithm identifies what Traffic Signal it is.

7.4 Speed Bump Detection

Speed bump detection is one of the key action which the vehicle need to understand in-order to improve user comfort, as part of the Advanced driver assistance system (ADAS) in the vehicle.

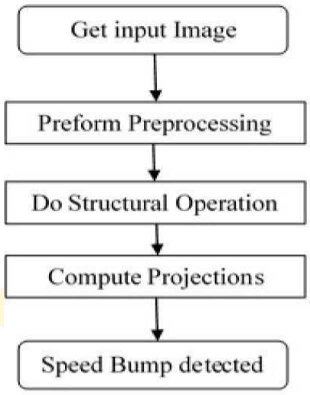


FIGURE5. 11: Flow Chart of Speed Bump Detection

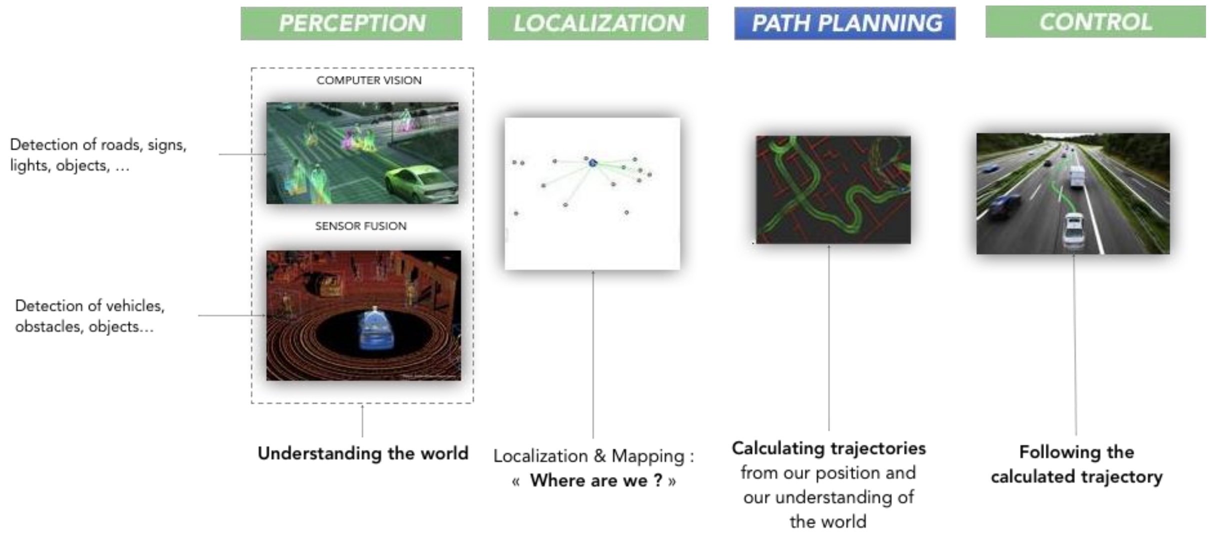
It is also a safety feature for the vehicle and the driver. A similar approach as the previous system is taken here as well. Figure 11 shows the flow chart used for the implementation criteria in this system. First the image is converted from RGB to gray using the Gray scale function. Then the Grayscale is converted to binary image. A morphological image processing is performed on this binary image [11]. A morphological processing is a set of operations that transform images per the size, shape, connectivity using the concept of set theory and other function. The process includes three steps, Opening, Area opening and Filling. After which a spread over Horizontal projection is performed. Projection is a one-dimensional representation of an image computed along horizontal and vertical axis. It's the summing up of pixel value in rows and columns. The results show that this type of speed bump detection is ideal for all the bumps with zebra crossing pattern.

7.5 Reinforcement learning

A more and more popular approach is reinforcement learning. This machine learning technique consists of learning from experience. If we want to turn right, we ask the car to make a random choice, if it is good, it receives a positive reward, if not, a negative. Over the course of the training, the car is able to learn what has caused a positive reward, and to reproduce it. This technology is at this day the one that comes closest to human learning.

7.6 Autonomous Navigation

This is the Path Planning step in which our car uses its knowledge of the environment and its position to plan trajectories.



7.8 Decision-Making

Once we have an estimate of the future of the environment, we can make a decision. How to brake if a pedestrian is detected? How to accelerate orchange lanes?

The first thing we have to do is environment classification. The choices are not the same whether we are driving on a highway or in a parking lot. Several criteria are taken into account when generating a trajectory, in particular safety, feasibility, efficiency and legality. Other variables can also betaken into account such as passenger's comfort.

7.8.1 Finite-State Machine

The first decision-making method that can be used is a finite state machine. The principle is to define, according to the situations, the possible states of a car. On a highway, the state of a car may be to stay in a lane, change lanes to the left,or change lanes to the right. Depending on the traffic conditions, we change state to, for example, overtake a car.The choice of states is usually made using cost functions. For each possible scenario, we calculate independent costs (distance to obstacles, legality) and addthem up. The lowest cost scenario wins.

Here we define what is important. We cannot do an impossible or dangerousmaneuver.

Total_Cost = Feasibility_Cost * 5+ Security_Cost * 4 + Legality_Cost * 3+Comfort_Cost * 2 + Speed_cost*1

Fig 7.8.1 graphical representation of speed cost

7.8.2 Speed Cost Function

In the cost function for speed, we do not want our vehicle to be too slow or exceed the maximum speed allowed. We therefore define a decreasing cost according to the speed then maximum after the speed limit. Decision making is a very delicate subject when talking about autonomous driving. We must take into account the current situation and decide on everything that can be done from this point. Then we must weigh the pros and cons of each possibility and finally choose the best solution.

7.9 Trajectory generation

The final step is trajectory generation. In this step, it is necessary to use a different coordinate system than the Cartesian coordinate system. For good reason, the cartesian coordinate system takes into account the dimension (x; y) but does not make sense if we want to find one's bearings in relation to the road. The Frenet coordinates contains two axes, an s axis indicating the advance relative to the track and a d axis indicating the distance to the center of the lane. This marker is the one on which we base ourselves to estimate if our trajectory deviates from the center of the lane or if a vehicle is ahead of us orbehind us.

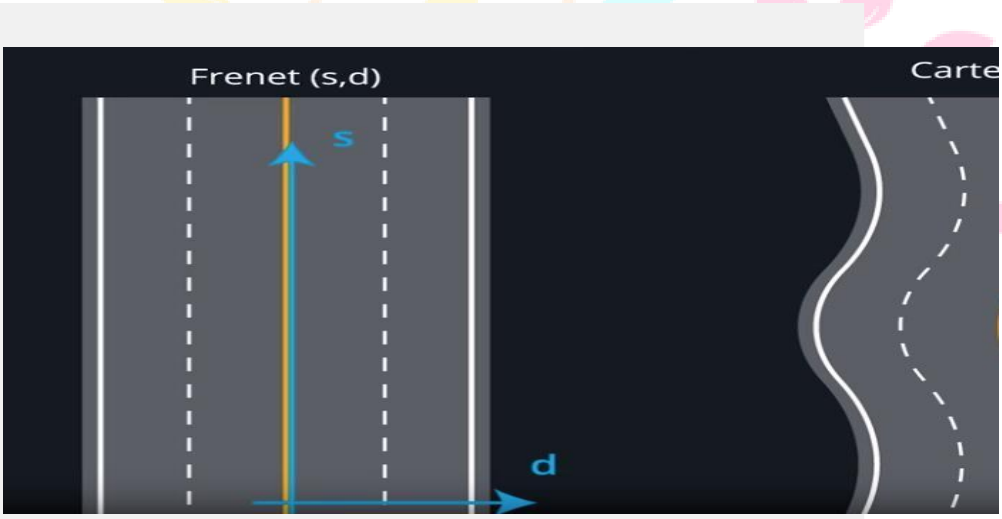
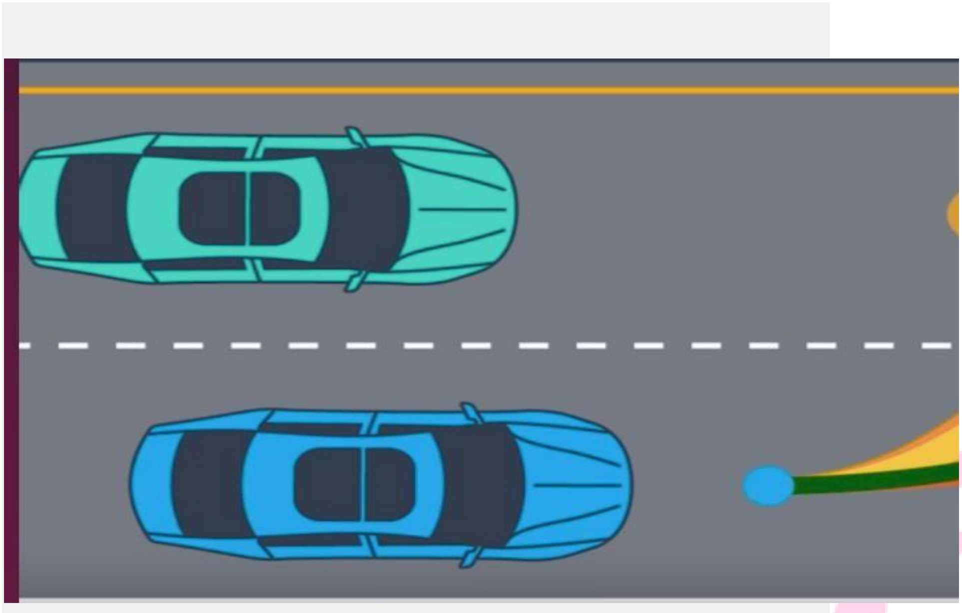


Fig7.9 Frenet vs Cartesian coordinates

When we take the decision to overtake a vehicle, the algorithm generates several trajectories for a decision and chooses the best one according to the criteria of feasibility, safety, legality, efficiency, comfort, …

In this case of overtaking, the red / orange trajectories are dangerous, the yellows are acceptable but incomplete, the green is the most efficient and safe.



7.10 Trajectory choice

A trajectory is a curve that goes through all these points. These points are positioned in space and time. They tell us when to move to a specific (x;y) position and how fast. If you want to brake at a pedestrian crossing, we create points up to the pedestrian crossing and set a decreasing velocity to the speeds ofthe points up to the stopping zone.

To generate this trajectory, we create a level five polynomial that passes through waypoints. Waypoints are points on the way that contain 3 dimensions:

S: The longitudinal distance

D: The lateral distance

T: The moment at which one must pass through this point; giving speed

8.Result

8.1 LANE DETECTION

Original Image	
ROI Area	
RGB To Gray Conversion Of Original Image	
Gray Stretch Image	
Median Filtered Image	

Canny Edge Detection Output Image	
Spiking Neural Network Output Image	
Lane detection output after Hough transform	

9. CONCLUSION

Putting all of these described blocks together leads to the construction of an automated vehicle. Automation is the outcome of a series of processes, which loop from the acquisition of environmental data, to data processing, and the usage of processed data to control thevehicle. Motion control can occur according to a user predetermined orvehicle generated path by sending a command to the vehicle actuator. This loop ensures the vehicle moves in the desired direction. Motion changes environmental data, which is acquired by the sensors to restart the series of processes. Adaptation to the environment following set objectives is fundamental to robotics, as it is to life. This common behavior brings the machine closer to biological systems, as the vehiclelearns by reacting to its environment over time. It is predicted that the vehicle will become smarter, following a more complex decisional pattern and taking into account more information. In this way an automated vehicle will be able to perform tasks of increasing complexity. Looking into the future of human-machine interaction, our vision of the machine can only evolve in tandem with machine complexity. Vehicle automation, from this viewpoint, is indeed awindow into the evolution of self and technology.

10-References

[1] Autonomous Driving: Technical, Legal and Social Aspect Book by Barbara Lenz, Chris Gerdes, and Markus Maurer

[2] Driverless: Intelligent Cars and the Road Ahead Book by Hod Lipson and Melba Kurman

[3] Autonomy: The Quest to Build the Driverless Car and How It Will Reshape Our World Book by Christopher Shulgan and Larry Burns

[4] Deep Learning Book by Aaron Courville, Ian Goodfellow, and Yoshua Bengio

[5] Principals of digital image processing Book by Wilhelm burger,mark.j.burge