ISSN : 2347 - 2243

Indo - American Journal of Life Sciences and Biotechnology

www.iajlb.com
Email : editor@iajlb.com or iajlb.editor@gamil.com

Indo-Am. J. of LifeSc & Bt.,2021- *ISSN 2347-2243 www.iajlb.com*

Indo-American Journal of Life Sciences and Biotechnology

Vol.18, Issue.1, feb 2021

Vision Based Robotic Car

Mir Shafeeq Ur Rehman1, Dr P.V. Sivakumar2

Abstract

According to this study, the implementation of a Vision Based Robotic Car, which uses Raspberry Pi and a mounted Pi-camera, is explained. Input photos from the Pi-camera are processed using a vision library that uses a circle recognition algorithm to track round objects, allowing the automobile to drive itself. It also has an Ultrasonic sensor that helps the car avoid obstructions. DC motor steering is accomplished by the use of a twin H bridge motor driver installed on the automobile. An autonomous vehicle based on vision is the focus of this article. A number of tests are carried out in the surrounding area, and the findings are positive. "

Keywords: Python, Flask, Flask server, and a Dual H bridge for circle detection are some of the tools that make up this system.

1**Introduction**

2. Robotics nowadays are so advanced that they enable self-awareness, wireless connectivity, biological sensing capabilities, integrating all information processing and transferring to distant systems, monitoring and sensing mechanisms, and so forth. It's no surprise that consumers see automated cars as a trendy technology improvement since they desire to have more comfort and security when driving. Human mistake is to blame for a large number of mishaps in the modern world. Most road accidents are caused by drivers who fail to observe traffic laws, resulting in a collision with an item in front of them. Autonomous driving can be achieved with a combination of cutting-edge technologies like as Computer Vision and high-performance sensors that minimise the possibility of human mistake. In certain circumstances, it may be more convenient to conceive of a car that can connect over a network and be driven through an android application, rather than a car that can only be accessed by a human in the immediate vicinity of the vehicle. Th

1M.Tech. Student, Department of Computer Science and Engineering, VNR Vignana Jyothi Institute of Engineering and Technology, Hyderabad, India

2Associate Professor, Department of Computer Science and Engineering, VNR Vignana Jyothi Institute of Engineering and Technology, Hyderabad, IndiaShafeeqrehman309@gmail.com Sivakumarpasupuleti@gmail.com

3. Development of an android-controlled vision-capable robotic automobile is the primary focus of this article. Users may switch between Auto Mode and Manual Mode in Android applications. Manual mode gives the user access to navigation and live streaming on the user's monitor. As long as the user has Auto mode chosen, Pi-Camera will transmit live video to the user by taking photographs as input, processing, and alerting the user when a circle-like item is spotted. An ultrasonic sensor helps the car steer away from obstructions in its path, while the Raspberry-Vision Pi's library and Circle identification algorithm allow the car to follow a circle-like object, which in some ways makes it self-driving.

5. Related work

Autonomous vehicle research has been going on since 1920, however the first promising testing took place in the 1950s and 1960s. The first self-driving automobile was developed in 1980 by Carnegie Mellon University's navlab and ALV [1, 2]. Autonomous vehicle research is now possible because of this. For the first time, in July 2013, Vislab demonstrated the autonomous driving of BRAIVE on a mixed traffic route. 3] A forward-facing camera and a sensor are attached to the real car, and the distance between the car and the object is measured via Inverse Perspective Mapping (IPM). This method is used to warn drivers of potential collisions, but it cannot send commands to take control of the vehicle. In this paper, an effort is made to develop an autonomous system that can work without human intervention. 4] [4]. A relatively new development is the self-driving automobile developed by Google. The algorithm for the Google self-driving car was created by Stanford University professor Sebastian Thrun and his colleagues. Google self-driving cars are intended to securely navigate metropolitan streets. Objects, including people and automobiles, may be detected from a distance of up to two football fields away, thanks to sensors installed in such vehicles. This product will be on sale in 2020. Fully driverless Tesla motors compete with Google's self-driving automobiles. To date, these prototypes have been equipped with a real-time driver for collecting test data. If a virtual driver can take the place of a real-life one, then the danger to human life can be reduced. In order to process traffic lights and the edges of lanes using HAAR feature-based cascade classifiers, a lane detection and edge detection method and computer vision library are used. [6] Live video streaming is provided by using the flask framework to serve the car, and an Arduino micro controller with a distance sensor is used to determine the distance, give commands to the car, but in this paper, we use the flask framework for live video streaming, which also shows the circle detection on the user's window screen in the form of a circle. The efficient processing of pictures of objects discovered is made possible by the employment of a powerful GPU jet-son TK1 for image processing object tracking. [7]. The notion of critical points and critical lines, where sensor nodes are installed to monitor an area, is created as a divide-and-conquer approach to environmental monitoring. Areas are split into smaller regions by crucial lines and the surveillance problem is likewise broken down into subproblems of boundary monitoring for each area. Introduced to assist in the monitoring and tracking of objectives [8] are robotic assistants. In order to

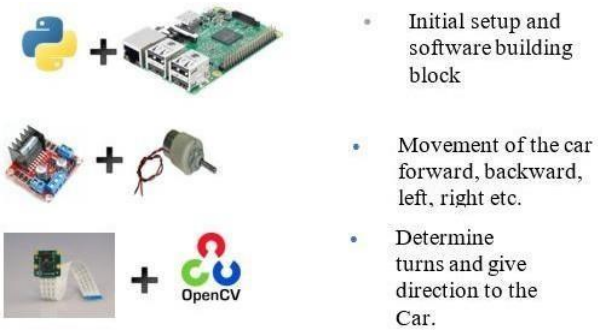
extract features from a picture using python programming and train these photos for circle identification using pi-camera, the ideas of OpenCV and its libraries are used. [10] In this study, they built a Bluetooth-enabled robot automobile that runs an Android app.

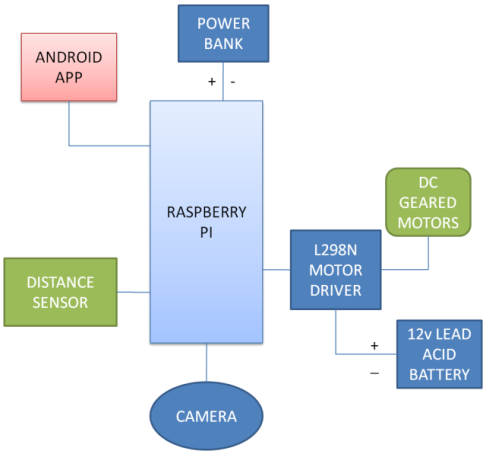
A temperature sensor, humidity sensor, and a distance sensor were all employed. Using the Wi-Fi connectivity provided in the Raspberry Pi and an Android application integrated with an IP address of Robotic, we were able to connect to the Robotic Car using Wifi technology, which has a greater range than Bluetooth and is more efficient.

6. Methodology

a. Defintion of Initial Configuration:

Figure 1: Initial Setup.





Block Diagram

Figure 2: Block diagram of proposed work.

As illustrated in the Fig.1. The first setup starts with configuring Raspberry-Pi installing Rasbianlinux based operating system; this is done by using the software programme Win32DiskImager it assists in writing the OS image to SD card which has minimum 16GB of storage capacity. Once the OS image is written on SD card it is ready to mount on to the Raspberry Pi's given SD card port. The Raspberry Pi boots up the operating system from SD card. In our work we utilise Python as programming language. Raspbian OS comes with pre-installed python IDE, which is used for basic setting. In order to utilise Pi- Camera we must install OpenCV and also import numpy, cv2 libraries which enables Pi- Camera to read, show and store picture into the Raspberry-Pi. As illustrated in the Fig.2. We utilised two DC geared motors which is controlled by Dual H-bridge L298N motor driver along with power supply 12v LEAD ACID rechargeable battery. An extra distance sensor, HC-SR04 ultrasonic sensor is also fitted for obstacle avoidance.

- **Raspberry Pi:**

6. Hardware Description:

The Raspberry-Pi 3 model-B is utilised in our work which comes with 1GB of RAM operates on Quad Core 1.2GHz Broadcom BCM2837 64bit CPU, Bluetooth Low Energy (BLE)-BCM43438 wireless LAN on board connectivity and also includes an Ethernet Base connectivity. It contains 40-GPIO(General purpose Input/Output) pins tailored for interacting with components, these pins may be controlled with software application. These pins output 3.3 volts when set to HIGH and no voltage when set

to LOW. It also features four USB 2.0 ports for interacting with mouse, keyboard and enables media storage connection. A DSI 15 pin HDMI connector is supplied for video output and a 15 pin MIPI port for camera communication. It obtains 2.5A of power supply may be supplied using micro USB connector given on it. This hardware component functions as a crucial module in our work as the functionality of the Robotic vehicle fully depends on this, as it act as means of communication between the android app and automobile. All components must connect with this module such as camera, motor driver and distance sensor.

• L298N Dual H-Bridge motor driver:

It is a module used for regulating speed and directions of various sorts of motors. In our job we employ two DC geared motors. A H-bridge is a circuit that can drive a current in either polarity and be controlled by pulse width modulation. It can drive up to 2 bi-directional DC motors that have voltage from 5V to 35V.

• DC Geared Motors:

A DC geared motor is generally used device which coverts electrical pulse into mechanical energy as the name says it comes with add on called Gear assembly assists in boosting torque and reducing the speed for any desired number. The speed of the engine is counted as RPM-Rotation Per Minute, using this the wheels of the automobile may revolve both in clockwise and anticlockwise orientations.

• HC-SR04 ultrasonic distance sensor:

This sensor employs non-contact ultrasound sonar to measure the distance and comprises of a control circuit, ultrasonic transmitters and receiver. The transmitters generate a high frequency ultrasonic sound, which bounce off any surrounding solid objects, while the receiver listens for any back echo. That echo is then analysed by the control circuit to compute the time difference between the signal being broadcast and received. The time gained is utilised in estimating the distance between objects. Its potential to determine the distance is between 2cm to 400cm. But this sensor cannot be utilised

straight onto to the Raspberry-Pi, it emitsoutput of 5V, has to lower to 3.3V since GPIO pins can tolerate up to 3.3v.

This is done by utilising resistors which can aid in decreasing the 5V output to 3.3V.

• Pi-Camera:

The Raspberry Pi Camera Module v2 is 8-megapixel Sony IMX219 image sensor specially made add-on board for Raspberry Pi, having a fixed focus lens. In order to operate in Raspberry Pi, the camera interfacing configuration is to be enabled then it is to be mounted to Pi by way of one of the tiny sockets available on the board top surface and uses the specialised CSI interface. This is the second key module as it takes photos, recognises the objects and classifies the needed characteristic from the image captured.

7. Software Description:

• Android Studio

We use this for developing our android application. It is an Integrated Development Environment [IDE] for Android app development, based on IntelliJ IDEA which is a powerful code editor and developer tool. It requires a minimum of 3GB of RAM, 2 GB of available disc space. It also requires Java

Development Kit [JDK] to be installed in to the system as all activities in our app are the Java classes. [SDK] software developer kit manager helps in downloading and installing the required Android framework and packages. Android Virtual Device [AVD] manager is an emulator which lets us assume it to be physical device where we develop, install and debug the app in AVD. Once the application is developed an Android Package [APK] file is generated, which has to be installed on android device.

• Python as software building block.

A programming language which allows both objects oriented and as well as procedural paradigms. It has a large library which can be utilised for machine learning, developing GUI application, image processing, Multimedia etc. Python is used to programme our Vision Based Robotic Car where each GPIO pin is programmed through terminal provided in Raspbian OS we can use either python IDE or terminal to write software code. The configuration of camera is also done using the above said terminal. libraries. The image processing libraries for Computer Vision are purely based on Python programming. The algorithm we use for Circle like object detection is well supported by the python functions and its

8. Working Model

As shown in the figure 3.1 and 3.2, An Assembled Vision Based Robotic Car, to get started, connect Raspberry-Pi to the power source as in our case we use a power bank, On getting started it boots up the OS and runs our Python Script automatically as we programmed it to be loaded on Boot-Up process, as the app.py script is running simply establish connection between the Car and Android Device

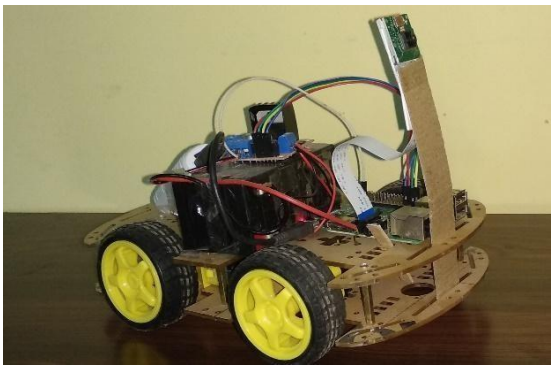


Figure 3.1: Assembled Robotic Car.

Figure 3.2: Assembled Robotic Car.

i. Establishing connection with android application: First the user opens the android app and inputs the IP address and port number on which our Raspberry- Pi by way of Flask server hosting the Vision Based Robotic Car android application. Once the connection is established the user is supplied with two modes Auto mode and Manual mode. On choosing Auto mode the user is supplied with live streaming of the area using Pi-Camera and the automobile operates itself without human involvement.

ii. Object detection Using OpenCV:

As illustrated in the Figure:4 The Pi-Camera placed on Raspberry-Pi with OpenCV assists in capturing the photos of the surroundings. As the objective of this work is to recognise circle like shapes and pursue those items. The Pi-Camera classifies the photos from the ongoing video it scans the images from video, frame by frame and coverts the image into a grayscale image as it assists in extracting the characteristicsof the image simpler comparison to a coloured image. In our script we use NumPy to store the (x, y) coordinates i.e., tracked location as a list, imutils library and its methods assists in positioning, scaling, translation, rotation, Skeletonization and displaying Matplotlib of the tracked picture, after the object is tracked, CV2 function Gaussian blur assists minimizing the noise from tracked image, then it is transformed into HSV [Hue, Saturation, Value] colour space. The functions CV2.range, CV2.dilate, CV2.erode aids in masking and removal of blobs in the picture. The Contour function helps in determining the Radius and Centroid of the detected object and creates a boundary around the item detected, the CV2.line function helps in altering the thickness of the boundary created on the object. CV2.imshow method eventually presents the frame to the screen which is live streamed on the Android Application given window and then it follows that item.Figure 4: Object detection using Pi-Camera.

Table 1. Experimental Instructions and Result

Instruction from Android application	Raspberry-Pi Instruction Fetch and Service	Motor Operation		CAR Function
		Left Motor	Right Motor	
Video Stream	CV2.imshow()	-	-	Streaming video
Right	GPIO.setup(pinRight)	LOW	HIGH	Turn RIGHT
Left	GPIO.setup(pinLeft)	HIGH	LOW	Turn LEFT
Forward	GPIO.setup(pinForward)	HIGH	HIGH	FORWARD
Backward	GPIO.setup(pinBackward)	HIGH	HIGH	BACKWARD

Conclusion

Using vision-based robotics, we've created a vehicle with a high degree of automation and network connectivity that's dependable and quick. To put it another way, using Python's built-in methods, OpenCV was able to analyse images quite well, correctly recognising objects, categorising them, and drawing a circle around each one so that the automobile could track it. The prototype that was constructed was tested in the real world and found to perform well when given a command, as well as being able to operate independently of user input. With this prototype, we're learning about robotics and image processing in preparation for using it to build our own self-driving automobiles in the real world.

REFERENCES

[1] Autonomous land vehicle project at cmu, by T. Kanade, C. Thorpe, and W. Whittaker, Proceedings of the ACM 14th annual conference on Computer science, 1986. Pp. 71–80 in ACM, 1986

[2] First findings in robot road following were published in the International Journal of Computational Intelligence (IJCAI), pp. 1089–1095.

[3] For an automobile environment, Distance Determination for an Automobile Environment using Inverse Perspective Mapping in OpenCV," was presented at the UCC 2010 conference in Galway, Ireland, in 2010.

[4] J.Levinson,J.Askeland,J.Becker,J.Dolson,D.Held,S.Kammel,J.Z.

[5] Kolter,D.Langer,O.Pink,V.Prattetal.,“To wardsfullyautonomous

[6] system and algorithm." in Intelligent Vehicles Symposium (IV), 2011 IEEE, pp. 163–168.

[7] IOT-based autonomous vehicle design using computer vision, Mohammad Rubaiyat Tanvir Hossain Md. Asif Shahjalal and Nowroz Farhan Nur., "ECCE 2017: Design of an IOT-based autonomous vehicle," February 16-18, 2017.

[8] A Raspberry Pi-based prototype for an autonomous car was developed by Tiple Anjali Hemant and Hemant P and Gurav Sagar Hanumant at the International Conference on Autonomous Vehicles (ICV) [6].

[9] Electronic and Communication Engineering Research Journal (IJERECE), April 2018.

[10] In "Object Tracking Bot Using On-Board Jetson TK1," Vikrant Patil, ManasaNavalgund, Ashwin Sahasrabudhe, and Shwetha Shinde. Big data, Internet of Things, and Data Science (BID) 2017: an international conference.

[11] "Surveillance and Tracking System with Collaboration on Robots, Sensor Nodes and RFID Tags" by Yang Xiao Yanping Zhang, University of Alabama, Tuscaloosa.

[12] "A quick introduction to OpenCV" by Ivan Culjak, David Abram, Tomislav Pribanic, HrvojeDzapo and MarioCifrek at IEEE 2012 in Opatija.

[13] The following individuals deserve special mention: Debarun Chakraborty, Kanhku Sharma, Ram Kishore Roy, and Hidam

[14] TulshiBezboruah, and Kumarjit Singh are the performers. Sensors put on a Bluetooth-enabled robotic car may be used to monitor and control its movement using the Androud Application. The Advances in Electrical, Electronic, and System Engineering Conference 2016 will be held in Putrajaya, Malaysia, on November 14-16.

[15][15] https://www.raspberrypi.org/documentation/usage/camera/

[16][16] https://developer.android.com/studio/intro

[17][17] https://www.pyimagesearch.com/2015/09/14/ball-tracking-with-opencv/