Tuijin Jishu/Journal of Propulsion Technology
ISSN: 1001-4055
Vol. 45 No. 3 (2024)

Next-Generation Autonomous Driving Using Deep Learning and OpenCV

[1] Thejaswini S, [2] Mamatha K R,[3] Ramesh Babu N, [4] Dr. Girish H

[1] *Dept. of ECE, B M S Institute of Technology and Management Bengaluru, Karnataka, India*

[2] *Dept. of ETE, B M S Institute of Technology and Management Bengaluru, Karnataka, India*

[3] *Dept. of CSBS, K S School of Engineering and Management Bengaluru, Karnataka, INDIA*

[4] *Professor, Department of ECE, Cambridge Institute of Technology, Bangalore*

Abstract: The objective of this paper is to design and conceptualize an autonomous car by integrating advanced technologies from key domains, namely Image Processing, Embedded Systems, and Neural Networking. As society increasingly demands both comfort and cutting-edge technology, our model seeks to merge these expectations by creating a vehicle that autonomously navigates complex environments with precision and safety.In this model, the autonomous car employs sophisticated image processing algorithms to interpret and respond to its surroundings, utilizing cameras as its primary sensors. These cameras continuously capture real-time data, which is then analyzed to recognize and react to various road conditions, obstacles, and traffic signals. This visual data serves as a critical input for the car's navigational system, which is guided by an embedded system designed to execute decisions in real-time. Furthermore, the car's decision-making process is powered by a neural network, trained to handle the diverse and dynamic challenges of real-world driving. This neural network enables the vehicle to learn from its experiences, improving its ability to predict and react to unforeseen circumstances over time. By incorporating machine learning techniques, the car adapts to different driving scenarios, ensuring a high level of safety and reliability. In conclusion, this paper presents a comprehensive approach to the development of autonomous vehicles, emphasizing the seamless integration of image processing, embedded systems, and neural networking. This integration aims to enhance human comfort by providing a technologically advanced solution for modern transportation needs.

Keywords: Autonomous Driving, Deep Learning, Neural Network, OpenCV

Introduction

A self-driving car, also known as an Autonomous Vehicle (AV) or robotic car, represents the pinnacle of modern transportation technology. These advanced vehicles are designed to navigate and operate independently, relying on a sophisticated network of sensors, cameras, radar systems, and other cutting-edge technologies. By continuously monitoring and interpreting their surroundings, self-driving cars can make real-time decisions, ensuring safe and efficient travel with minimal to no human involvement.

At the core of this innovation is a complex integration of multiple disciplines, including image processing, control systems, deep learning, and advanced software engineering. These components work in harmony to create a robust system capable of handling the dynamic and often unpredictable nature of real-world driving environments.

The ultimate goal of this technology is to achieve Level-5 Automation, a stage where the vehicle can handle all driving tasks under any conditions, without the need for human oversight or intervention. This represents a transformative shift in how we approach transportation, promising a future where travel is not only safer and more efficient but also more accessible to everyone.

1. Objective

The main objective of shaping a self-driving car is to consolidate both technology and networking to fabricate the finished model for the solace of human and environment. A road full of autonomous cars would also change the quality of driving [1]. Computerized vehicles could coordinate their movements with each other, drastically

improving their efficiency. The result would be a faster commute that allows people to simultaneously socialize, read or work while on the road and also autonomous systems ensures reduction in traffic deaths, reduction in harmful, emissions, improves fuel economy, reduction in travel time.

2. Problem Identification

To bring out the adaptability of the model to Indian roads where dimension/curvature of roads varies drastically, Potholes present themselves challenging to AI. That's because the current systems that self-driving vehicles rely on cameras, radar and really don't recognize potholes, almost no matter how big they are.

Taking the above statement as a major challenge in India, we have implemented the detection of potholes. The pothole detection is basically implemented by using blob detection technique which scans the entire digital image on searching the blob shapes. We have also proposed a method that detects the upcoming marked speed hump/bump in real time using deep learning techniques and gives the distance the vehicle is

away from it. The captured video is then processed to detect the potential bumps. Therefore, appropriately controlling the speed of the model and to achieve Level-5 automation [5].

3. Design and implementation

Firstly, the gear motors of the BOT are to be assembled and to supply the power, we can use a power source and accelerate the motors. Since the motors can be supplied with 3-6V DC voltage, we have to supply and maintain power within this range for the efficient operation of the autonomous system.

We need a microcontroller to control the gear motors and hence we employ Arduino which also acts as a slave device of the system. They receive inputs from various sensors mounted on the vehicle and converts them into actions by actuating and controlling various functions to perform the driving tasks. The Master device employed is the microprocessor Raspberry Pi module. H-bridge driver circuit supervise the movement of the motors in clockwise or anticlockwise direction upon encountering control signals from Raspberry Pi controller.

Raspberry pi camera module which captures the images of the route and guides towards the destination. It uses OpenCV which is a cross-platform library utilizing which we can develop real-time computer vision applications. We make use of C++/Python codes and set the camera for capturing pictures and videos [4]. Image processing is a method to perform some operations on an image, in order to get an enhanced image and or to extract some useful information from it. OpenCV can process images and videos to identify objects, faces, or boards, traffic signals.

The data processed by the OpenCV is fed into an end-to-end deep learning model that would recognize the lanes, detect and analyze the traffic signal and stop sign. Also detect another car in-front and taking necessary measures to maneuver itself around the track [6]. To describe the adaptability of our model to the Indian roads, we have implemented the detection of potholes. The pothole detection is basically implemented by using blob detection technique which scans the entire digital image on searching the blob shapes. We have also proposed a method that detects the upcoming marked speed hump/bump in real time using deep learning techniques and gives the distance the vehicle is away from it. The captured video is then converted to images. These images are then processed to detect the potential bumps. Therefore, appropriately controlling the speed of the model and detection is another trait that is of interest. It uses edge detection concept and prevent the driver/car system from drifting off the driving lane.

Lastly, we establish communication between the master and the slave systems while processing in info from the environment thus operating the car without the need for a driver thus, ensuring comfort and safety of the individuals.

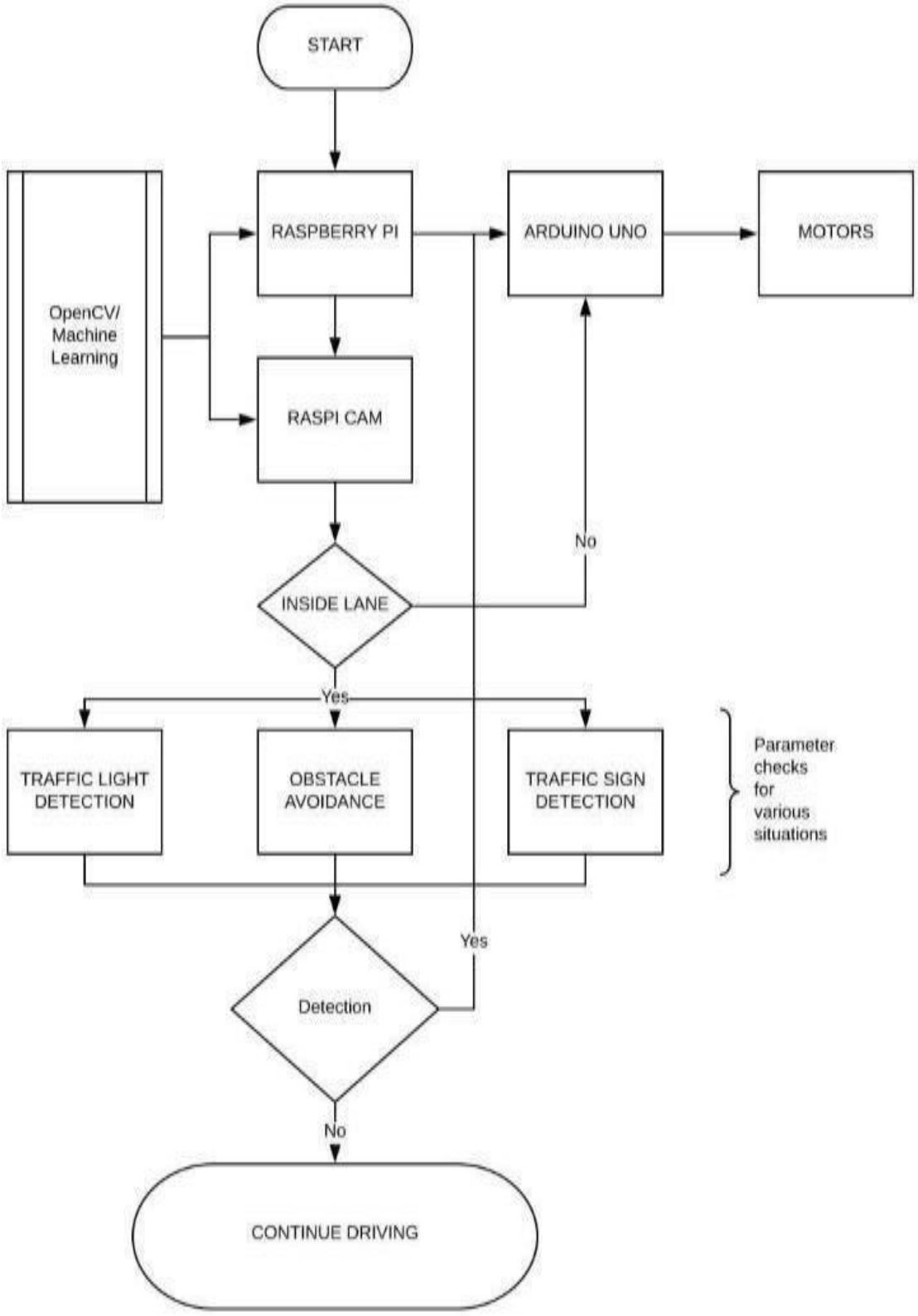


Fig 1. Flowchart

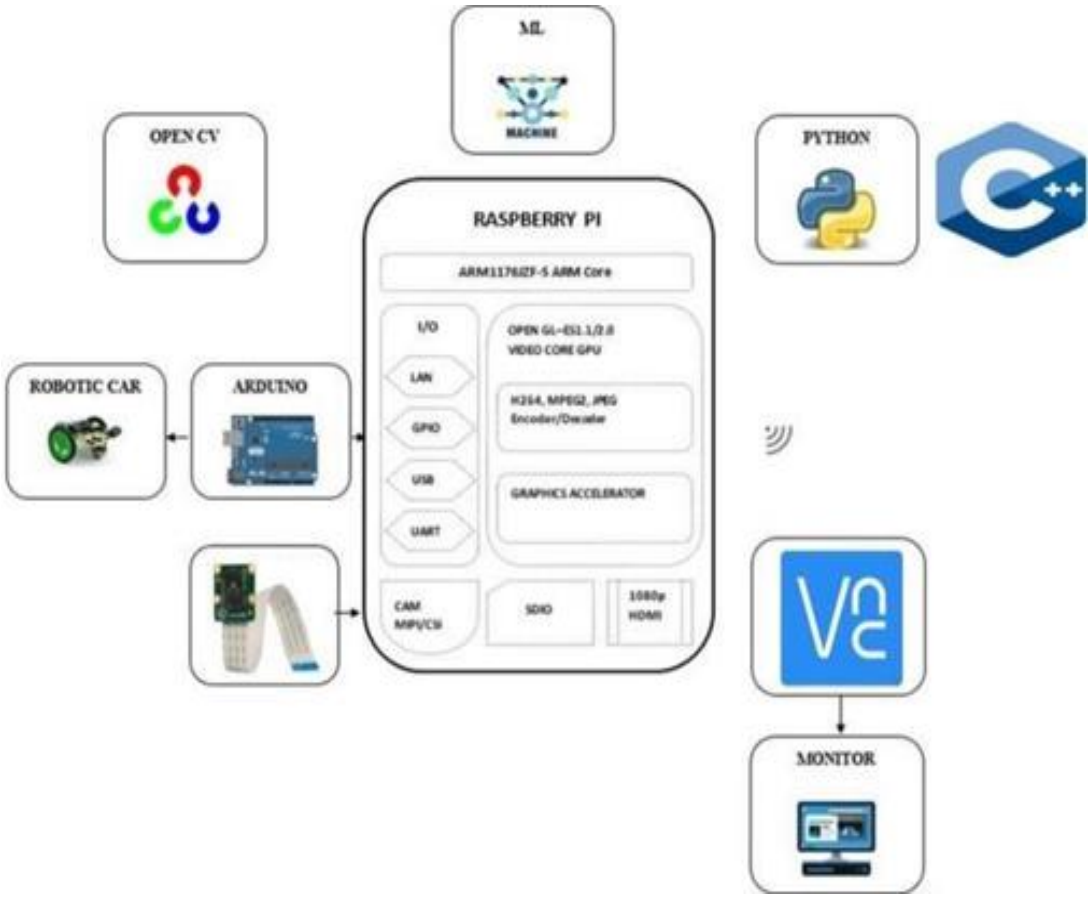


Fig 2: Block diagram

The car thus drives without any driver, identifying the obstacles, other vehicles, detects the traffic signals, sign boards, speed limits and performs accordingly those obtained in the remaining directions, if the edge strength is greater than the current pixel, the current pixel value will be discarded and the new edge strength value will be prevalent. Thus, by setting a threshold value, we can ensure safe travel. If the edge strength value is greater than the threshold value, we can declare it as a danger curve and take remedial actions.

4. Algorithms

i. Canny Edge Detection

This method is employed for lane detection to identify the precise path, ensuring a safe journey. Initially, a Gaussian filter is applied to the image to smooth it, eliminating noise and light fluctuations. By using four filters corresponding to horizontal, vertical, and diagonal directions, the intensity gradient of the image can be determined, providing angle values in every direction that signal sharp curves on the road. The edge strength of the current pixel is then compared with adjacent pixels to identify edges.

ii. Haar cascade model

This cascade model is used for detecting the obstacles as desired by the user, for detecting the traffic boards and signals.It is a machine learning based technique where a cascade function is trained from positive and negative images [7]. The first step is to alloy all the Haar features, collect those features and assemble them into positive (those with obstacles) and negative images (without obstacles). To select the best features in an image, a concept called Adaboost is used which both selects the best features and trains the classifiers that use them. Each Haar feature is considered as a weak classifier and hence a large number of those features are necessary to describe an object with accuracy. Therefore, they are organized into Cascade classifiers to form a strong classifier. Cascade classifiers technique can be ensembled on the OpenCV platform, where different classifiers for each stage are

allotted, therefore differentiating the positive and negative images and the obstacles will be identified.

iii. Blob detection

For an image, we set the minthreshold, thresholdstep and the maxthreshold values where the thresholdstep values will be incremented by one, while analyzing the image. In a binary image, we set the white pixels (normal path without any humps or potholes) and black pixels (humps and potholes detected). We combine all the white pixels and discard them while the black pixels are computed to note the blob size, height and the width of the blob.

5. Methodology

i. Hardware

a. Raspberry Pi

The Raspberry Pi is a compact, affordable single-board computer, originally developed by the Raspberry Pi Foundation in the UK. It is a fully functional mini-computer capable of handling tasks similar to a standard desktop computer. The Raspberry Pi 3 Model B is the third generation of this series and surpasses its predecessors, including the original Raspberry Pi Model B+ and Raspberry Pi 2 Model B. While retaining the same popular board format, the Raspberry Pi 3 Model B offers a more powerful processor—10 times faster than the first generation—and features built-in wireless LAN and Bluetooth, making it an excellent choice for connected and high-performance applications.

b. Raspberry Pi Camera

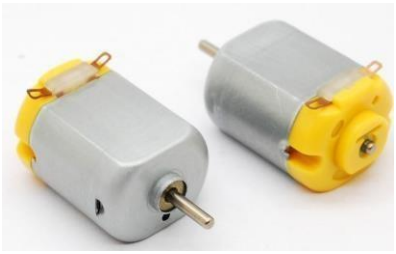
The Raspberry Pi Camera Module is a 5MP CMOS camera with a fixed-focus lens capable of capturing both still images and high-definition video. The camera board features a 5-megapixel sensor and connects to the Raspberry Pi via a ribbon cable to the CSI connector. It captures still images at a resolution of 2592 x 1944, and supports video recording at 1080p at 30 FPS, 720p at 60 FPS, and 640x480 at either 60 or 90 FPS. The camera consists of a compact circuit board (25mm by 20mm by 9mm) that connects to the Raspberry Pi's Camera Serial Interface (CSI) bus connector through a flexible ribbon cable. The image sensor has a native resolution of five megapixels and comes with a fixed-focus lens. The camera's software supports full-resolution still images up to 2592x1944 and video resolutions of 1080p30, 720p60, and 640x480p60/90.

c. H-Bridge Motor Driver(L298)

The L298 is a monolithic integrated circuit available in 15-lead Multiwatt and PowerSO20 packages. It is a high-voltage, high-current dual full-bridge driver designed to interface with standard TTL logic levels and control inductive loads such as relays, solenoids, DC motors, and stepper motors. The circuit includes two enable inputs, allowing independent control of the device regardless of the input signals. The emitters of the lower transistors in each bridge are connected, and an external terminal is provided for connecting an external sensing resistor. Additionally, a separate supply input is available to operate the logic at a lower voltage.

d. DC Motor

A DC motor is a type of rotary electrical machine that converts direct current (DC) electrical energy into mechanical energy. Most DC motors operate based on the forces generated by magnetic fields. Almost all DC motors incorporate an internal mechanism, whether electromechanical or electronic, to periodically reverse the direction of current flow in certain parts of the motor. DC motors were among the first widely adopted motors because they could be powered by the direct current lighting power distribution systems that were already in place.



e. Arduino Uno

The Arduino Uno is an open-source microcontroller board designed by Arduino.cc, built around the Microchip ATmega328P microcontroller. It features a set of digital and analog input/output (I/O) pins that can be connected to various expansion boards (shields) and other circuits. The board includes 14 digital I/O pins, six of which support PWM output, and 6 analog input pins. It is programmable using the Arduino IDE (Integrated Development Environment) via a type B USB cable be powered by the USB cable or by an external 9- volt battery, though it accepts voltages between 7 and 20 volts. It is similar to the Arduino Nano and Leonardo.

ii. SOFTWARE

a. Open CV

OpenCV (Open-Source Computer Vision Library) is a free, open-source library for computer vision and machine learning, designed to provide a standardized infrastructure for developing computer vision applications and to accelerate the integration of machine perception in commercial products. As a BSD-licensed project, OpenCV allows businesses to easily use and modify its code. The library contains over 2,500 optimized algorithms,

encompassing both classic and cutting-edge computer vision and machine learning techniques. These algorithms can perform a wide range of tasks, such as face detection and recognition, object identification, human action classification in videos, camera movement tracking, moving object tracking, 3D model extraction, 3D point cloud generation from stereo cameras, image stitching for high-resolution scene panoramas, image similarity searching within databases, red-eye removal, eye movement tracking, scenery recognition, and overlaying markers for augmented reality. With a user community of over 47,000 and more than 18 million downloads, OpenCV is widely used in industry, research, and government projects. It offers interfaces in C++, Python, Java, and MATLAB, and supports multiple operating systems, including Windows, Linux, Android, and Mac OS. Primarily geared toward real-time vision applications, OpenCV leverages MMX and SSE instructions where available and includes actively developed CUDA and OpenCL interfaces. The library, written in C++, features a templated interface that integrates smoothly with STL containers and comprises over 500 algorithms and thousands of supporting functions.

b. Raspberry Pi OS

Raspberry Pi OS (formerly known as Raspbian) is a Debian-based operating system designed specifically for Raspberry Pi devices. Since 2015, it has been the official operating system provided by the Raspberry Pi Foundation for the Raspberry Pi family of compact single-board computers. Originally developed as an independent project by Mike Thompson and Peter Green, the first version of Raspbian was released in June 2012. Earlier versions of Raspberry Pi OS were 32-bit and built on the Raspbian core, thus carrying the name Raspbian. However, with the introduction of recent 64-bit versions that no longer use the Raspbian core, the operating system was renamed Raspberry Pi OS for both 64-bit and 32-bit editions. As of August 1, 2020, the 64-bit version remains in beta and is not recommended for general use.

Raspberry Pi OS is specifically optimized for the Raspberry Pi line, which features ARM CPUs. It utilizes a modified version of the LXDE desktop environment, combined with the Openbox stacking window manager, a new theme, and a few other modifications. The distribution includes several pre-installed applications, such as the algebra software Wolfram Mathematica, a version of Minecraft known as Minecraft Pi, and a lightweight version of the Chromium web browser in its latest release.

c. Arduino IDE

The Arduino Integrated Development Environment (IDE) is a cross-platform application available for Windows, macOS, and Linux, designed for writing and uploading programs to Arduino-compatible boards. It is primarily written using functions from C and C++, and can also support other development boards through third-party cores. The source code for the IDE is released under the GNU General Public License, version 2. The Arduino IDE supports C and C++ languages, using specific rules for code structuring. It includes a software library from the Wiring project, offering many common input and output functions. User-written code needs only two basic functions: one for initializing the sketch and another for the main program loop. These are compiled and linked with a main() program stub into an executable cyclic executive program using the GNU toolchain, which is included with the IDE. The IDE then converts the executable code into a text file encoded in hexadecimal, which is uploaded to the Arduino board by a loader program in the board's firmware. By default, the Arduino IDE uses the avrdude tool to upload user code to official Arduino boards.

d. VNC Viewer

Virtual Network Computing (VNC) is a computing system that enables graphical desktop sharing by using the Remote Frame Buffer (RFB) protocol to remotely control another computer. It works by transmitting keyboard and mouse inputs from one computer to another, while simultaneously sending graphical screen updates back to the controlling computer over a network. VNC is platform-independent, with clients and servers available for a variety of GUI-based operating systems, including Java. Multiple clients can connect to a VNC server simultaneously. Common applications of VNC include remote technical support and accessing files on a work computer from home or vice versa. VNC was originally developed at the Olivetti & Oracle Research Lab in Cambridge, UK. The original VNC source code, along with many modern derivatives, is open-source under the GNU General Public License. There are various VNC variants that offer additional functionalities, such as

optimizations for Microsoft Windows or file transfer capabilities (not part of the standard VNC). Despite these differences, many of these variants are compatible with the standard VNC, allowing a viewer of one type to connect with a server of another.

e. Geany Program Editor

Geany is a lightweight graphical text editor built with Scintilla and GTK, offering basic IDE functionality. It is optimized for fast load times and has minimal dependencies on additional packages or external libraries in Linux environments. Geany has been adapted for various operating systems, including BSD, Linux, macOS, Solaris, and Windows. However, the Windows version lacks an embedded terminal window and certain external development tools available in other versions.

available on Unix systems unless the user installs them separately. Geany supports a wide range of programming and markup languages, including C, C++, C#, Java, JavaScript, PHP, HTML, CSS, Python, Perl, Ruby, Pascal, Haskell, and many others. Unlike traditional Unix-based editors like Emacs or Vim, Geany is more similar to programming editors typically found on Microsoft Windows, such as Notepad++, which also utilizes the Scintilla editing component

6. Conclusion

The Autonomous Cars if made prevalent in India will surely ensure safer commute of people and will lead to many activities in the field of Research and Development in the Telecommunication, Automobile, Machine Learning and Artificial Intelligence domain. Furthermore, the transition from self-driving cars with varying levels of autonomy to fully autonomous vehicles is yet to be made. In light of such rapid changes in intelligent transportation systems, the education system must align itself with these emerging technologies. Thus, we wish to frame a model which is fully automated, ensuring human comfort, safer environment and pave way for more research on the same. In the near future, Autonomous vehicles will be an indispensable part of modern transport systems. This proposed model, provides a methodological framework to model the Autonomous car which can respond to every practical obstacle on the road just like a human's intellect.

References

[1] Arduino - ArduinoBoardUno", Arduino.cc, 2016. [Online]. Available: https://www.arduino.cc/en/main/arduinoBoardUno. [Accessed: 05- Mar- 2016].

[2] "XBee radio communication between PICs - Northwestern Mechatronics Wiki",Hades.mech.northwestern.edu, 2016. [Online]. Available:http://hades.mech.northwestern.edu/index.php/XBee_rad io_communication_between_PICs. [Accessed: 15- Apr-2016].

[3] "L293D Motor Driver IC | L293D Datasheet - Engineers Garage", Engineersgarage.com, 2016. [Online]. Available:

[4] http://www.engineersgarage.com/electroniccomponents/ l293d-motor-driver-ic.

[5] H. Girish, T. G. Manjunat and A. C. Vikramathithan, "Detection and Alerting Animals in Forest using Artificial Intelligence and IoT," 2022 IEEE Fourth International Conference on Advances in Electronics, Computers and Communications (ICAECC), 2022, pp. 1-5, doi: 10.1109/ICAECC54045.2022.9716679.

[6] A. Devipriya, H. Girish, V. Srinivas, N. Adi Reddy and D. Rajkiran, "RTL Design and Logic Synthesis of Traffic Light Controller for 45nm Technology," 2022 3rd International Conference for Emerging Technology (INCET), 2022, pp. 1-5, doi: 10.1109/INCET54531.2022.9824833.

[7] S. Vaddadi, V. Srinivas, N. A. Reddy, G. H, R. D and A. Devipriya, "Factory Inventory Automation using Industry 4.0 Technologies," 2022 IEEE IAS Global Conference on Emerging Technologies (GlobConET), 2022, pp. 734-738, doi: 10.1109/GlobConET53749.2022.9872416.

[8] T G Manjunath , A C Vikramathithan , H Girish, "Analysis of Total Harmonic Distortion and

implementation of Inverter Fault Diagnosis using Artificial Neural Network", Journal of Physics: Conference Series, Volume 2161, 1st International Conference on Artificial Intelligence, Computational Electronics and Communication System (AICECS 2021) 28-30 October 2021, Manipal, India. https://iopscience.iop.org/issue/1742-6596/2161/1

[9] GIRISH H , "Internet of Things Based Heart Beat Rate Monitoring System", © September 2022 | IJIRT | Volume 6 Issue 4 | ISSN: 2349-6002 IJIRT 156592 INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN TECHNOLOGY 227

[10] Dr Girish H, Dr. Mangala Gowri , Dr. Keshava Murthy , Chetan Naik J, Autonomous Car Using Deep Learning and Open CV, PAGE NO: 107-115, VOLUME 8 ISSUE 10 2022, Gradiva review journal, ISSN NO : 0363-8057, DOI:10.37897.GRJ.2022.V8I8.22.50332

[11] Girish H. "Intelligent Traffic Tracking System Using Wi-Fi ." International Journal for Scientific Research and Development 8.12 (2021): 86-90.

[12] Girish H, Shashikumar D R, "Emission monitoring system using IOT", Dogo Rangsang Research Journal, UGC Care Group I Journal, ISSN: 347-7180, Vol – 8 Issue-14, No.6, 2020.

[13] Girish H, Shashikumar D R, "Display and misson computer software loading", International journal of engineering research & Technology (IJERT), ISSN: 2278-0181, Vol – 10 Issue-08, No.13, August 2020.

[14] Girish H, Shashikumar D R, "A Novel Optimization Framework for Controlling Stabilization Issue in Design Principle of FinFET based SRAM", International Journal of Electrical and Computer Engineering (IJECE) Vol. 9, No. 5 October 2019, pp. 4027~4034. ISSN: 2088-8708, DOI: 10.11591/ijece.v9i5.pp.4027-4034

[15] Girish H, Shashikumar D R, "PAOD: a predictive approach for optimization of design in FinFET/SRAM", International Journal of Electrical and Computer Engineering (IJECE) Vol. 9, No. 2, April 2019, pp. 960~966. ISSN: 2088-8708, DOI: 10.11591/ijece.v9i2.pp.960-966

[16] Girish H, Shashikumar D R, "SOPA: Search Optimization Based Predictive Approach for Design Optimization in FinFET/SRAM", © Springer International Publishing AG, part of Springer Nature 2019 Silhavy (Ed.): CSOC 2018, AISC 764, pp. 21–29, 2019. https://doi.org/10.1007/978-3-319-91189-2_3.

[17] Girish H, Shashikumar D R, "Cost-Effective Computational Modelling of Fault Tolerant Optimization of FinFET-based SRAM Cells", © Springer International Publishing AG 2017 R. Silhavy et al. (eds.), Cybernetics and Mathematics Applications in Intelligent Systems, Advances in Intelligent Systems and Computing 574, DOI 10.1007/978-3-319-57264-2_1.

[18] Girish H, Shashikumar D R, "A Survey on the Performance Analysis of FinFET SRAM Cells for Different Technologies", International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-4 Issue-6, 2016.

[19] [17] Girish H, Shashikumar D R, "Insights of Performance Enhancement Techniques on FinFET-based SRAM Cells", Communications on Applied Electronics (CAE) – ISSN: 2394-4714, Foundation of Computer Science FCS, New York, USA .Volume 5 – No.6, July 2016 – www.caeaccess.org

[20] [18] Girish H, Shashikumar D R, "DESIGN OF FINFET", International Journal of Engineering Research ISSN: 2319-6890) (online), 2347-5013(print) Volume No.5 Issue: Special 5, pp: 992-1128, doi: 10.17950/ijer/v5i5/013 2016.

[21] Dr. MANGALA GOWRI S G, Dr. SHASHIDHAR T M, Dr. SUNITHA R, SHYLAJA V, Dr. GIRISH H, "DESIGN OF 3-BIT CMOS WALLACE MULTIPLIER" , Seybold report, ISSN 1533-9211, Volume 18, Page No: 1260-1271 DOI: 10.5281/zenodo.8300481

[22] Dr. Mangala Gowri S G, Dr. Girish H, Ramesh N, Dr. Nataraj Vijaypur, "IOT based plant monitoring system and smart irrigation using new features" ResMilitaris, vol.13, n°2, January Issue 2023

https://resmilitaris.net/menu-script/index.php/resmilitaris/article/view/3312/2608

[23] Girish H.2023, Smart Theft Securityvehicular System Using Iot. Int J Recent Sci Res. 14(02), pp. 2881-2884. DOI: http://dx.doi.org/10.24327/ijrsr.2023.1402.0591

[24] Dr. Mangala Gowri S G, Dr. Girish H, Dr. Santosh Dattatray Bhopale, Weed Detection based on Neural Networks, International Journal of All Research Education and Scientific Methods (IJARESM), ISSN: 2455-6211 Volume 11, Issue 3, March-2023, Impact Factor: 7.429, Available online at: www.ijaresm.com DOI: https://doi.org/11.56025/IJARESM.2023.11323351

[25] Shashidhara, K.S., Girish, H., Parameshwara, M.C., Rai, B.K., Dakulagi, V. (2023). A Novel Approach for Identification of Healthy and Unhealthy Leaves Using Scale Invariant Feature Transform and Shading Histogram-PCA Techniques. In: Shetty, N.R., Patnaik, L.M., Prasad, N.H. (eds) Emerging Research in Computing, Information, Communication and Applications. Lecture Notes in Electrical Engineering, vol 928. Springer, Singapore. https://doi.org/10.1007/978-981-19-5482-5_47

[26] R. Team, "60RPM 12V DC Motor with Gearbox [RKI- 1036] - Robokits India, Easy to use, Versatile Robotics & DIY kits", Robokits.co.in,2016. [Online].Available: http://robokits.co.in/motors/60rpB-12v-dc-motor- withgearbox.

[27] "Arduino MPU 6050 - Best IMU Sensor Tutorial – DIY Hacking", DIY Hacking, 2014. [Online]. Available: http://diyhacking.com/arduino-mpu-6050-imu-sensortutorial/."Magnetometer", Wikipedia, 2016. [Online]. Available: https://en.wikipedia.org/wiki/Magnetometer.