Self-Driving Car Based on Image Processing and Microcontroller

Roopesh A R, Assistant Professor Dept. of ECE, RIT, Hassan, Karnataka
Shubas S R, Assistant Professor Dept. of ECE, RIT, Hassan, Karnataka
Manoj S, Dept. of ECE, Rajeev Institute of Technology, Hassan, Karnataka
Likitha S R, Dept. of ECE, Rajeev Institute of Technology, Hassan, Karnataka
Nisarga H R, Dept. of ECE, Rajeev Institute of Technology, Hassan, Karnataka

Abstract- Road accidents near school or hospital zones are all too common in today's environment. We see a lot of people driving at high speeds under the influence of alcohol or for thrill causing road accidents. So, vehicle speed control must be taken into consideration to avoid casualties due to over speeding. The purpose of this system is to control the speed of the vehicle by recognizing speed sign boards on the roadside with the help of Rasberry Pi board and image processing. The camera is positioned at the front of the vehicle to capture the images of sign boards to process them. The captured images are then compared with the stored images using Python open CV to control the vehicle's speed in the speed restricted areas. The objective of this project is to reduce the accidents caused due to over speeding. The automatic sign detection and recognition has been converted to a real challenge for high performance of computer vision and machine learning techniques. Traffic sign analysis can be divided in three main problems: automatic location, detection and categorization of traffic signs. Basically, most of the approaches in locating and detecting of traffic signs are based on color information extraction.

I. INTRODUCTION

Most of the traffic accidents are the result of neglectfulness, ignorance of the traffic rules and disobeying traffic sign boards, by the drivers and also people in the society at large. Due to inflated vehicle density and over speed driving causes a lot of accidents. Image processing technology plays an important role in the speed limit sign board capturing. In this journal we have introduced a system that can help the driver, significantly increasing passenger’s safety. Road sign detection and recognition systems have also been implemented lately by many companies. In earlier days the road signs were detected manually by the drivers. But now the Automatic speed controlling of vehicles based on signboard detection using image processing can easily recognize the signs using the camera module [1].

Thanks to quantum leaps made in computing technologies in the past 30years cheap sensing, reliable object recognition, and real-time, portable, large-scale data analysis automated vehicles are becoming a reality. Inspired by ongoing research today from around the world, this MQP aims to imagine a divergent take on autonomous vehicle technology by challenging modern vision algorithms combined with affordable sensing technology [2].

Using vehicle to vehicle communication systems, these automatic speed control systems can also be used to reduce the speeding of the vehicle to avoid accidents. This can be accomplished with the help of sensors installed in the vehicles which provide relative distances and speeds between them [3].

In this paper, we introduce an automatic detection and recognition system of speed- limit sign board that can handle different conditions of lighting and blurriness in images. The system uses a detection method based on MSER detection and recognizes speed-limit signs with Histogram of Oriented Gradient features by a SVM classifier [5].

These accidents results in blockage of road which contributes to traffic jams. So we have to deploy sophisticated speed guns and cops to detect violators. This system addresses all these issues at on go, being major motivation behind the project [4].

Self-driving cars, also known as autonomous vehicles, rely on advanced technologies like image processing and microcontrollers to navigate without human intervention. Image processing plays a crucial role in enabling these vehicles to perceive their surroundings by analyzing visual data from cameras mounted on the car.

Integrating image processing with microcontrollers enhances the vehicle's decision-making capabilities, making self-driving cars more reliable and efficient.

II. BLOCK DIAGRAM

This development aims to build a monocular vision autonomous car prototype using Laptop with Arduino as a processing chip. An HD camera along with an ultrasonic sensor is used to provide necessary data from the real world to the car. The car is capable of reaching the given destination safely and intelligently thus avoiding the risk of human errors. Many existing algorithms like lane detection, obstacle detection are combined together to provide the necessary control to the car.

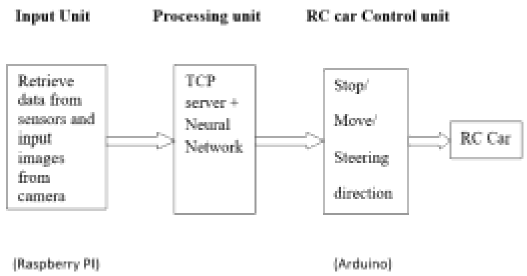

Fig.1: Block diagram of Block diagram of Self- Driving Car.

1. Input Unit: Retrieve Data from Sensors: Collects real-time data from various sensors (e.g., ultrasonic, infrared) to detect obstacles, speed, distance, etc. Input Images from Camera: Captures visual data using a camera for object recognition, lane detection, and other image-based processing.

2. Processing Unit: TCP Server: Handles communication between different modules, enabling data transfer between the Raspberry Pi and Arduino. Neural Network: Processes the sensor and camera data to make intelligent decisions (like identifying objects, lanes, and determining actions based on learned models).

3. RC (Car Control Unit): Stop/Move/Steering Direction: Controls the car's motion based on the processed commands. Stop: Engages braking mechanisms when an obstacle is detected. Move: Commands the car to move forward, backward, or maintain speed. Steering Direction: Adjusts the car's direction to avoid obstacles, follow lanes, or switch directions as needed. Executes the final movement commands, controlling the actual physical motion of the self-driving car based on the processed data.Proposed self-driving model:

The proposed model takes an image with the help of camera attached with Laptop with Arduino on the car. The Raspberry-Pi and the laptop is connected to the same network, the Laptop with Arduino sends the image captured which serves as the input image to the Neural Network. The image is grey- scaled before passing it to the Neural Network. Upon prediction the model gives one of the four output i.e. left, right, forward or stop. When the result is predicted corresponding signal is triggered which in turn helps the car to move in a particular direction with the helpof its controller.

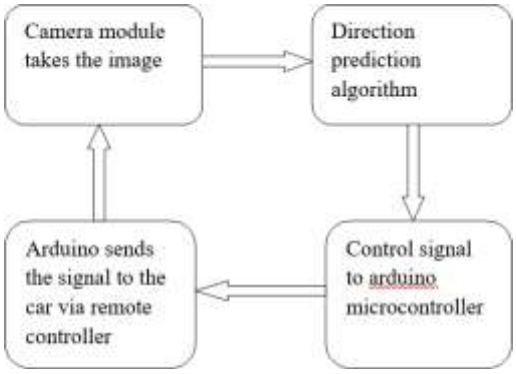

Fig.2: Block diagram of Proposed self- driving model

1. Capturing Phase: To detect motion, we first have to capture live images of the area to be monitored and kept under surveillance. This is done by using camera.
2. Comparing Phase: Comparing the current frames captured with previous frames to detect motion: for checking whether any motion is present in the live images, we compare the live images being provided by the web cam with each other so that we can detect changes in these frames and hence predict the occurrence of some motion.
3. Pre-processing: Pre – processing is heavily dependent on feature extraction method and input image type. Some common methods are: De- noising: applying a Gaussian or simple box filter for de-noising.
4. Contrast enhancement: if gray level image is too dark or bright. Down sampling to increase speed. Morphological operations for binary images. Scaling by some factor.

III. IMPLEMENTATION:

Flow Chart:

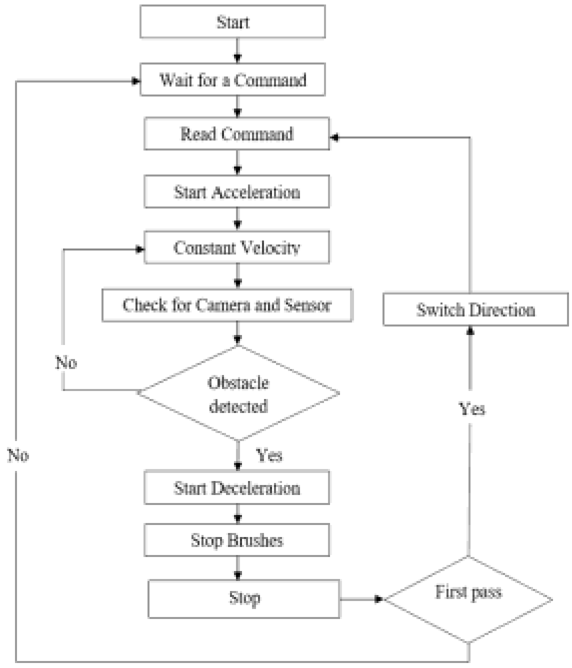

Fig. 3: Flow Chart of Smart

1. Start: The process begins here.
2. Wait for a Command: The system stays idle until a command is received.
3. Read Command: Once a command is detected, it is read for further processing.
4. Start Acceleration: The system initiates movement by accelerating.
5. Constant Velocity: After accelerating, the system maintains a constant speed.
6. Check for Obstacle and Sensor: The system continuously checks for any obstacles using sensors.
7. Process Sensor Data: Convert and interpret the raw sensor values.
8. Identify Hand Gesture: Match the processed data with pre-defined gestures.

If No Obstacle Detected: The system continues at constant velocity.

If Obstacle Detected: The system proceeds to start deceleration.

9. Start Deceleration: The system begins to slow down upon detecting an obstacle.
10. Stop Brakes: After decelerating, the braking mechanism is activated to stop the system.
11. Stop: The system comes to a complete halt.
12. Switch Direction (if applicable): There's an option to switch direction depending on specific conditions.
13. Yes: If switching is required, the system changes direction and resumes checking for obstacles.

14. No: Continues the process as is.

IV. RESULTS AND DISCUSSION

Fig 4: Image Sign processing

Sign Recognition: The last stage of the algorithm is the recognition of the traffic signs. The key principle is to match the detected signs to a database (library) of traffic sign templates. Several techniques can be used for this purpose. The target of the recognition procedure is to assign each region of interest to the class that it belongs. If the match is found sound notification is given to the driver, interpreting the meaning of the candidate sign. . If the ROI does not match with any of the templates in the database (library) of traffic sign templates, the Candidate image is discarded. But the false rate is generally less since the candidate images with no blobs have already discarded using the noise removal and also the shape classification. Traffic signs have two major features: Shape (square, circle, triangle, etc.) and Color (red, yellow, blue, green). Detecting shapes in an urban environment is very challenging and unreliable due to complications in image acquired. Also, environmental conditions play an important role in the detection procedure.

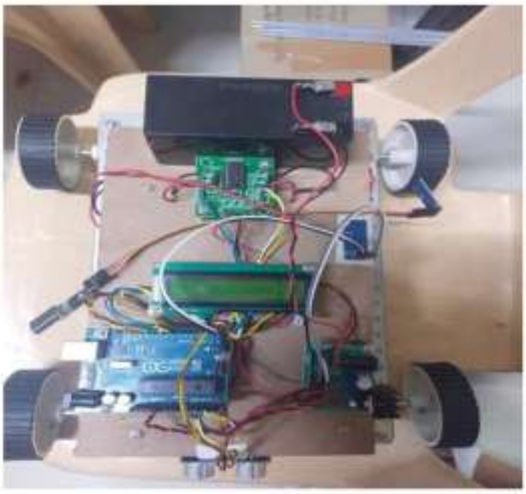

Fig 5: Module of self-driving Car

The methodology for developing a monocular vision autonomous car prototype involves integrating hardware and software components to enable intelligent navigation. A laptop, functioning as the primary processing unit, runs a neural network model that analyzes real-time input from an HD camera and an ultrasonic sensor. These sensors continuously capture environmental data, which is then processed to detect lanes, obstacles, and other critical driving elements. The neural network model interprets the visual input and generates steering commands, ensuring the car can navigate towards its destination while minimizing human intervention. The processed control signals are transmitted to an Arduino board, which serves as the interface between the laptop and the RC car's mechanical systems. This allows for precise movement adjustments based on the predictions made by the neural network. By combining multiple existing algorithms, such as lane detection and obstacle avoidance, the system enhances the vehicle's ability to operate safely and autonomously. The overall framework ensures real-time decision-making, contributing to a more efficient and intelligent self-driving prototype. This allows for precise movement adjustments based on the predictions made by the neural network. By combining multiple existing algorithms, such as lane detection and obstacle avoidance, the system enhances the vehicle's

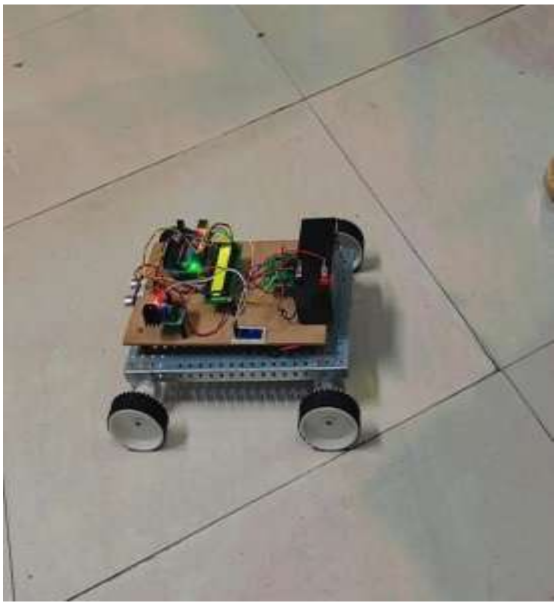

Fig 6: Result of self-driving Car

ability to operate safely and autonomously. The overall framework ensures real-time decision-making, contributing to a more efficient and intelligent self- driving prototype.

To further refine the autonomous car prototype, the methodology incorporates a structured approach to data processing, decision-making, and actuation. The HD camera provides a continuous video feed, which is pre-processed using computer vision techniques such as edge detection and color filtering to enhance lane recognition. Simultaneously, the ultrasonic sensor measures distances to nearby objects, ensuring effective obstacle detection.

V. CONCLUSION

Self-driving cars, powered by advanced image processing and microcontroller, represents a revolutionary steps in transportation technology, offering transformative benefits across multiple dimension. The self-driving car successfully demonstrated autonomous navigation capabilities, showcasing the integration of sensors, actuators, and programming logic. Key challenges included calibration of sensors, optimization of navigation algorithms, and handling real-world environmental variability. Potential improvements include adding GPS for path planning, machine learning for smarter decision-making, and advanced sensors like LiDAR for greater precision. The project demonstrates the potential for Arduino-based solutions in autonomous systems, relevant for industries like transportation and robotics. The implementation of a self-driving car using image processing and a microcontroller demonstrates the potential for autonomous vehicle technology at a smaller scale. By integrating a camera for real-time image acquisition, image processing techniques for object detection and lane tracking, and a microcontroller for decision-making and control, the system can successfully navigate its environment with minimal human intervention.

In conclusion, while this prototype demonstrates the viability of self-driving technology at a basic level, further advancements in hardware and algorithms are needed for real- world deployment.

REFERENCES

1. E. Hariharan, Mr. M. Arun "Automatic Speed Controlling of Vehicles Based on Signboard Detection using Image Processing" International Research Journal of Engineering and Technology (IRJET) Jan 2022.
2. Mrs. Susmitha B C, Karthik Sharma K "Automated Vehicle Driving using Image Processing" International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering June 2021.
3. Tejo Sumanth Pavuluri, T. Vamsi Krishna "Automatic Speed Control System in Speed Zone Areas Using OpenCV" International Conference on Electronic Circuits and Signaling Technologies 2022.
4. Chavan Avinash Ganpat, Wakchaure Ganesh Subhash "Vehicle Speed Control System Using Image Processing" www.ierjournal.org 2023.
5. Aditya Anand, Rushiraj Gawande "Intelligent Vehicle Speed Controlling and Pothole Detection System" E3S Web of Conferences 2020.
6. Aditya Kumar Jain, "working model of Self- driving car using Convolutional Neural Network, Laptop Processor and Arduino" 2018.