Design And Implementation Of Real-Time Autonomous Vehicle Using IoT BASED Image Processing

Group 7
121AD0013 – Madhav
121AD0024 – Vivek
121AD0005 – Ankit
121AD0009 – Satyam

DESIGN AND IMPLEMENTATION OF REAL-TIME AUTONOMOUS VEHICLE USING IOT BASED IMAGE PROCESSING

Narender Chinthamu[1], George Chellin Chandran J[2], A.Usha Ruby[3], C. Thangavel[4], Vjay Birchha[5], Himanshu Pant[6]

Article History: Received: 12.02.2023 **Revised:** 27.03.2023 **Accepted:** 12.05.2023

Abstract

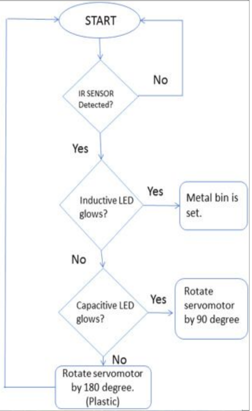
The design and implementation of a real-time autonomous vehicle using IoT-based image processing is presented in this research. The system incorporates a sensor camera, Raspberry Pi, and a Convolutional Neural Network (CNN) program to detect and classify various vehicles and obstacles in the surrounding environment. The captured images are transmitted to the Raspberry Pi, where a Python program executes the CNN algorithm for object identification. Upon detection, the system sends signals to the motor controller, reducing the motor speed accordingly. Additionally, voice commands are used to alert the user about the presence of objects. Ultrasonic and proximity sensors further enhance the system's capabilities by providing real-time feedback. The research results demonstrate the successful detection and classification of different vehicles, validating the effectiveness of the CNN program. The

Research Paper

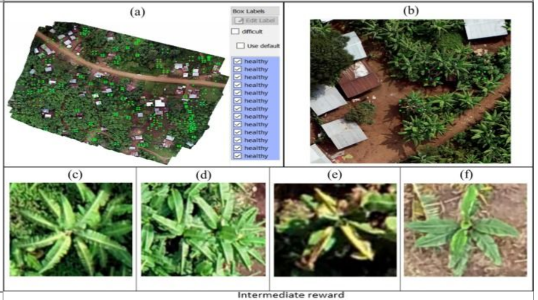
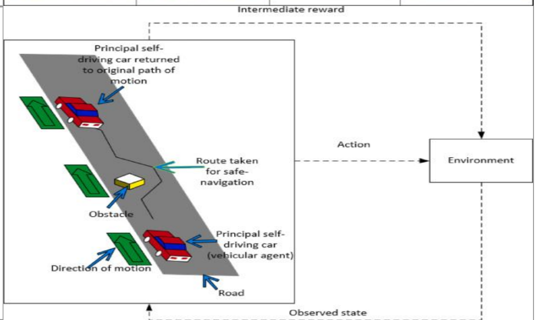
Abstract

- Detects and classifies vehicles and obstacles.
- Python program executes CNN algorithm for object identification on RPi.
- Motor speed controlled based on object detection.
- Voice commands to reduce the risk of accidents.
- Ultrasonic and proximity sensors provide real-time feedback.

Literature Survey

Paper Name	Author	Abstract
SmartBin system with waste tracking and sorting mechanism using IoT	Adi Suvarnamma	  Waste Segregation Diagram
Identification of Pathological Disease in Plants using Deep Neural Networks - Powered by Intel Distribution of OpenVINOTM Toolkit	Risab Biswas	

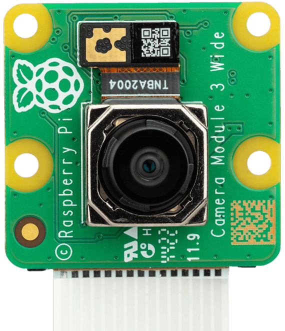
Literature Survey (Contd..)

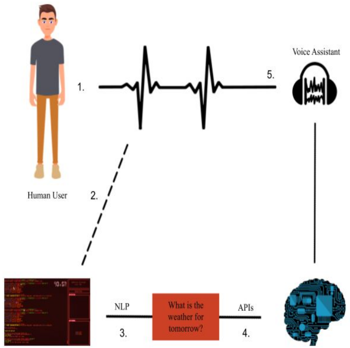
Paper Name	Author	Abstract
Detection of banana plants and their major diseases through aerial images and machine learning methods: A case study in DR Congo and Republic of Benin	Michael Gomez Selvaraj	
Deep learning for object detection and scene perception in self-driving cars: Survey, challenges, and open issues	Abhishek Gupta	

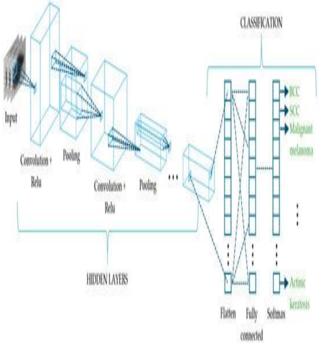
Proposed Methodology

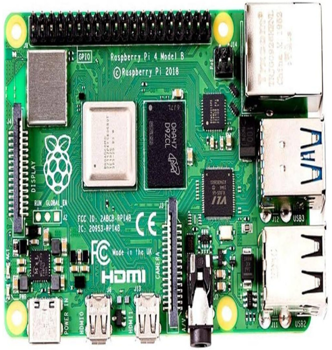
- Uses real time image processing using IoT based systems
- Uses CNN for image recognition aided by sensors
- Has voice identification which prevents the accidents
- Motor controls which reduce speed based on obstacle

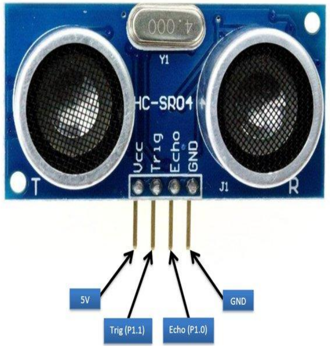
Components

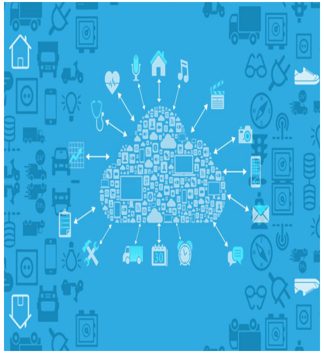

©Raspberry Pi
Camera Module 3 Wide
TNBA2004


Human User
1.
2.
5.
Voice Assistant
NLP
3.
What is the weather for tomorrow?
APIs
4.


CLASSIFICATION
Input
Convolution + Relu
Pooling
Convolution + Relu
Pooling
HIDDEN LAYERS
Flatten
Fully connected
Softmax
BCC
SCC
Malignant melanoma
Actinic keratosis


Raspberry Pi 4 Model B
©Raspberry Pi 2018
HDMI

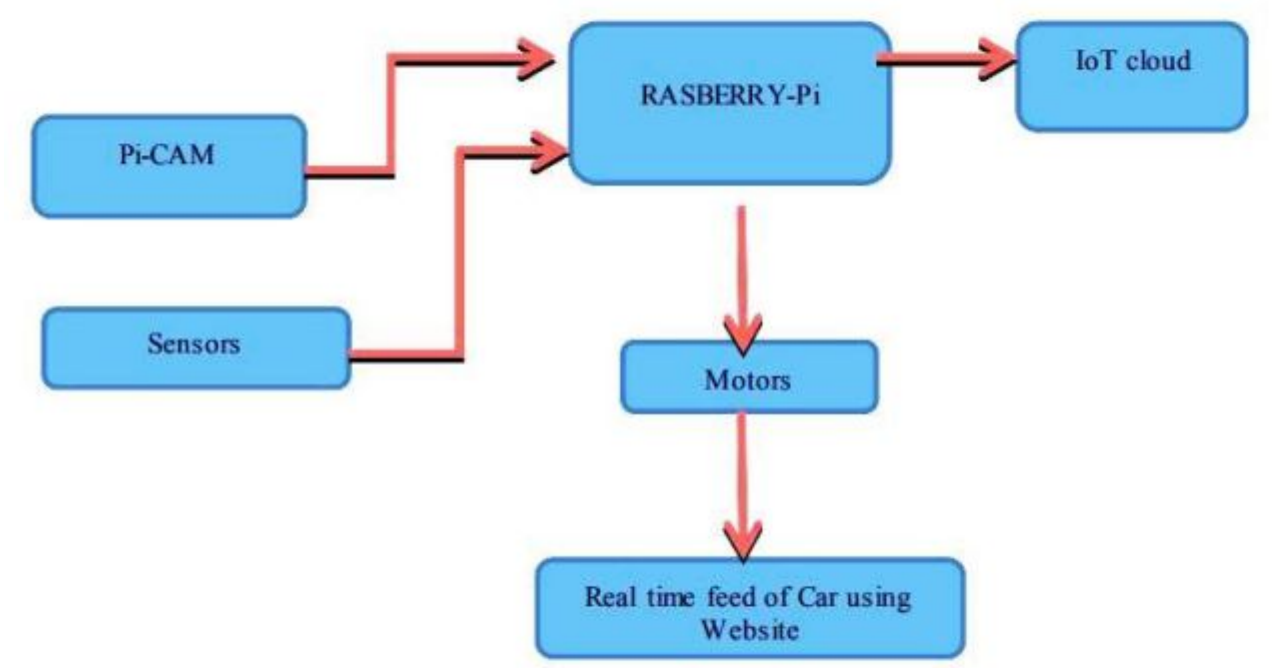

Y1
HC-SR04
Vcc
Trig
Echo
GND
T
J1
R
5V
Trig (P1.1)
Echo (P1.0)
GND



System Components Overview

- **Camera:** Captures images of the vehicle's surroundings.
- **Raspberry Pi:** Central computing platform for image processing.
- **Image Processing:** Python program on Raspberry Pi analyzes images.
- **Convolutional Neural Network (CNN):** Identifies objects within images.
- **Object Recognition Output:** Provides information about identified objects.
- **Current Controller:** Adjusts motor speed based on object detection.
- **User Interface:** Enables interaction with the system.
- **Voice Commands:** Provides real-time feedback to the user.
- **Ultrasonic Sensor:** Measures distance between vehicle and nearby objects.
- **Proximity Sensor:** Enhances obstacle detection capability.

Architecture



CNN Architecture

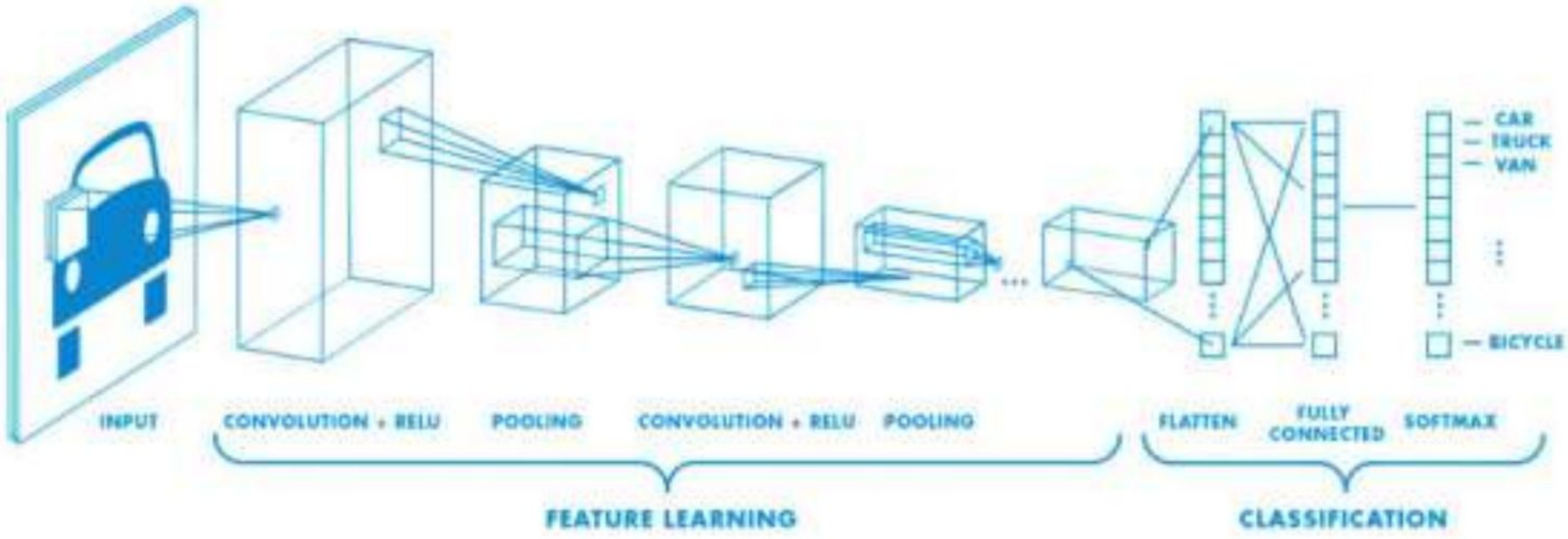


Fig. 2. CNN architecture

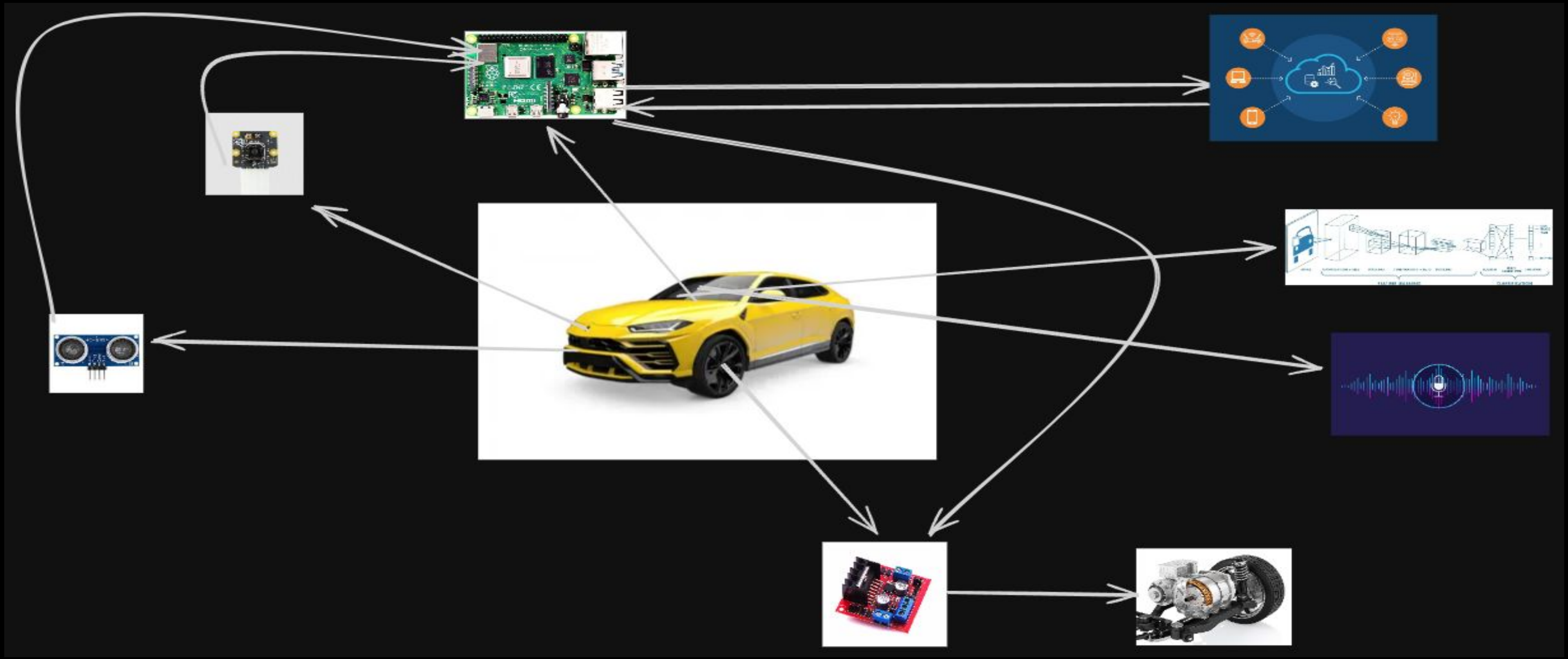
System Workflow

- Camera captures images of surroundings.
- Images are processed using Python program on Raspberry Pi.
- Convolutional Neural Network identifies objects in images.
- Object Recognition Output provides information about objects.
- Current Controller adjusts motor speed based on detected obstacles.
- User Interface interacts with the system, providing feedback to the user.
- Voice Commands alert the user about detected objects.
- Ultrasonic and Proximity Sensors contribute to obstacle detection.

System Output

- Control signals sent to the current controller adjust motor speed.
- Voice commands alert the user about detected obstacles.
- Combined output ensures real-time feedback and obstacle avoidance.
- Enhances safety and usability of the autonomous vehicle system.

Implementation



System Integration and Testing

- Integrated system into the front part of a car.
- Utilized a small DC motor setup powered by a battery for vehicle movement.
- Manually tuned acceleration and deceleration for smooth operation.
- Main objective: detect objects using sensor camera, communicate with Raspberry Pi, execute CNN program for object identification.
- Motor speed adjusted based on detection results for enhanced safety.

CNN Classification of Objects

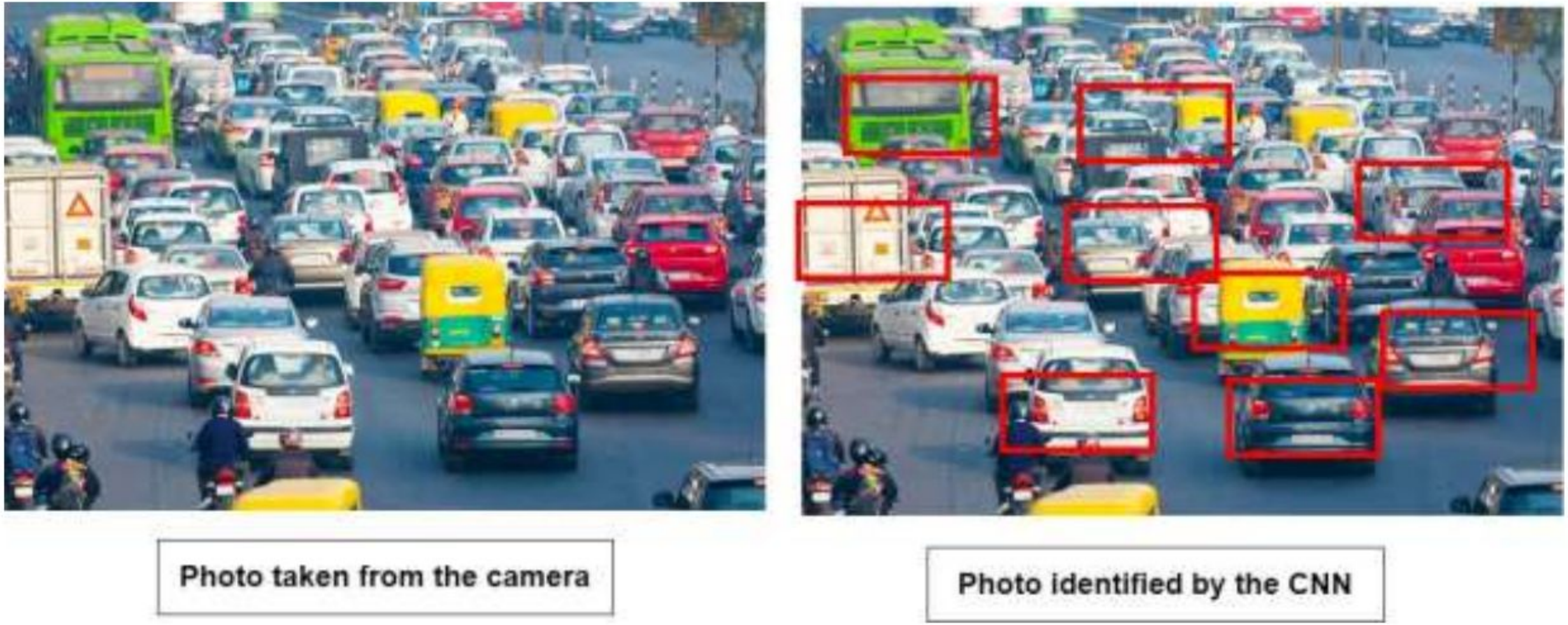


Fig. 3 Image obtained from camera and identified by the CNN

Experimental Results and Performance Evaluation

- Tested system performance in real-world scenarios.
- Sensor camera captured surroundings, including vehicles and obstacles.
- CNN program accurately recognized various vehicle types: cars, trucks, buses, bicycles, pedestrians.
- Motor speed adjusted based on object detection to prevent collisions.
- Ultrasonic and proximity sensors used to enhance user experience.
- Voice commands issued to alert users about detected objects.
- Successful detection of vehicles validates effectiveness of CNN program.
- Continuous monitoring and dynamic speed adjustments enhance safety and navigation efficiency.

Sensor Data and Motor Speed Adjustment

- Ultrasonic Distance: Distance measured by ultrasonic sensor.
- Proximity Sensor: Detection of objects by proximity sensor.
- Camera Detection: Objects identified by camera (e.g., cars, trucks, bicycles, pedestrians).
- Speed Reduction: Percentage reduction in motor speed based on detected objects.
- System dynamically adjusts motor speed to mitigate risks associated with detected objects.
- Continuous monitoring of environment ensures efficient navigation in presence of vehicles and obstacles.

Sensor Readings and Motor Response

Reading	Ultrasonic Distance (cm)	Proximity Sensor (Boolean)	Camera Detection	Speed Reduction (%)
1	50	1	Car, Truck	10
2	30	0	Bicycle	20
3	40	1	Pedestrian	15
4	60	1	None	0
5	35	0	Car	10
6	25	1	Truck	20
7	45	1	None	00
8	55	0	Bicycle	15
9	20	1	Pedestrian	25
10	50	1	Car	10

Future Works

- The paper proposes a LiDAR sensor , the main advantage over normal camera is that normal camera will give us two dimensional view and has less perception of distance
- But by including LiDAR the image obtained is 3 Dimensional
- The latest technology implements this , the paper was written exactly an year before so author is unaware

Limitations

- The CNN model runs on RPi causing slow execution speed of model we can make model run on cloud and communicate using API’s
- The car does not inculcate potholes, the CNN is not trained on potholes and hence user gives alert to car about that , this may cause jerks
- The car does not have a method to sense depth of any object in other words we have 2-D camera sensor
- The image data is very heavy that is why it's not recommended anymore to use camera

Conclusion

- The paper integrated camera, CNN and voice based command feature in one workflow
- The paper has Motor responses like humans which makes it more attractive
- The paper has CNN model running on RPi
- The paper proposes LiDAR sensor as future work

Thank You

Any questions ?

GROUP 7

- Satyam Prashant, 121ad0009@iiitk.ac.in
- Ankit Kumar, 121ad0005@iiitk.ac.in
- Vivek Kumar, 121ad0024@iiitk.ac.in
- Madhav Sharma, 121ad0013@iiitk.ac.in