Project Title: Voice Based with UGV Obstacles Detection Model

Name: Ajay Krishna DM
Guide: Srikanth Reddy
Batch: COMETFWC037
Date: 22 October 2025

Voice Based UGV Control with Obstacle Detection

Abstract

In this paper, we demonstrate how to guide an unmanned ground vehicle (UGV) using a gamepad interface as well as voice commands. This is done through an android application for sensing voice or touch on the phone and relaying the control data to the UGV via Bluetooth. An ultrasonic sensor is used to detect and avoid obstacles automatically. This platform offers a low-cost alternative for implementing artificial intelligence (AI) algorithms on embedded hardware.

1 Introduction

Autonomous navigation has been a major area of research in robotics, with pioneering projects such as [?], [?, ?], demonstrating autonomous navigation in complex environments. End-to-end learning approaches, such as NVIDIA's system for self-driving cars [?], have further simplified navigation pipelines by mapping sensor inputs directly to control outputs. Speech-based human-robot interaction has enabled intuitive control of robots, including intelligent wheelchairs and mobile robots [?, ?, ?], while datasets such as Google's Speech Commands [?] have accelerated the development of voice-controlled systems.

Inspired by the above, this work presents a prototype using an ESP32 microcontroller, L293D motor driver, and an ultrasonic sensor to build a voice-enabled toy car that can detect obstacles while being controlled through voice and Bluetooth.

2 Hardware Setup

The components used in this project and their description are listed in Table 1.

Component	Description
ESP32	Microcontroller with Bluetooth and Wi-Fi
L293D	Motor Driver IC
Ultrasonic Sensor (HC-SR04)	Measures distance to detect obstacles
DC Motors	Drive the toy car wheels
Chassis	Car body to hold components
Power Bank	Power source for ESP32
Jumper Wires	For connections

Table 1: List of Components

Assembly Steps

1. Assemble the chassis, fix the motors, and mount the wheels.
2. Fix the breadboard on the base of the car.
3. Plug the **L293D** motor driver IC on the breadboard.
4. Connect L293D output pins to the motors (M_1, M_2).
5. Connect ESP32 to L293D as per motor control pins.
6. Connect the ultrasonic sensor to the ESP32 as shown in Table 2.

Ultrasonic Pin	ESP32 Pin
VCC	5V
GND	GND
TRIG	GPIO 5
ECHO	GPIO 18

Table 2: Ultrasonic Sensor and ESP32 Connections

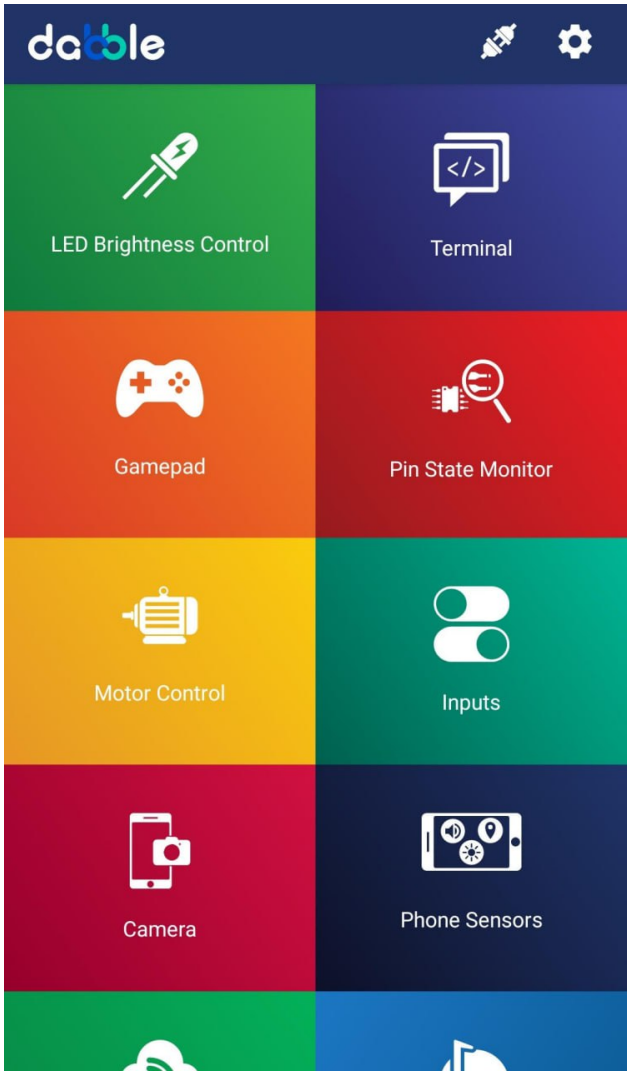

Figure 1: Dabble App Main Interface

Dabble App Interface

L293D Motor Driver IC

ESP32 Microcontroller Board

3 Software Setup

3.1 Dabble App Control

1. Install the *Dabble* app from the Google Play Store.

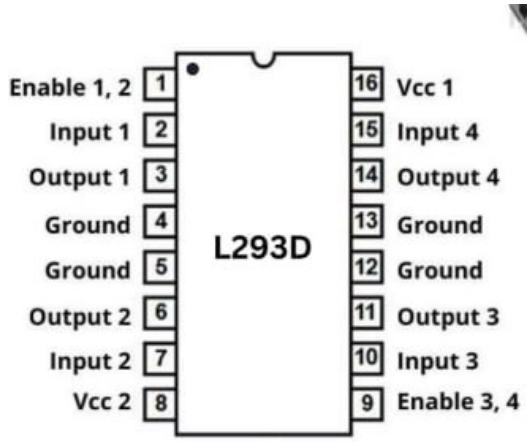

Figure 2: Pin Diagram of L293D Motor Driver IC

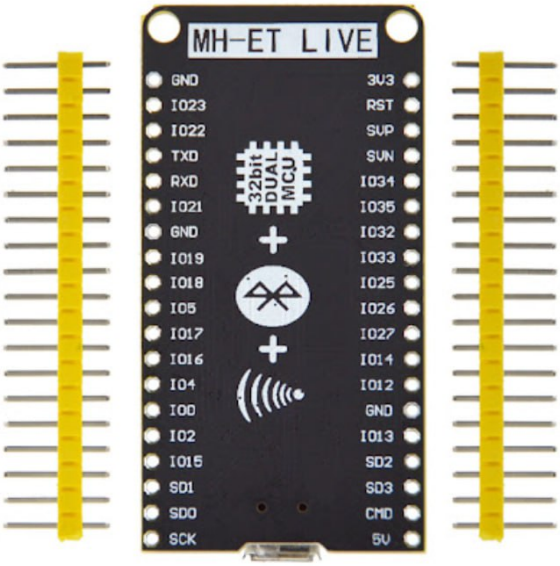

Figure 3: ESP32 MH-ET LIVE Board

2. Upload code to ESP32 using PlatformIO from the following link: `https://github.com/gad dabble_gamepad.cpp`

3. Replace contents of `src/main.cpp` in PlatformIO with the above code and upload.

4. Connect the ESP32 to the phone via Bluetooth.

5. Use the Gamepad in the Dabble app to control directions — Forward,

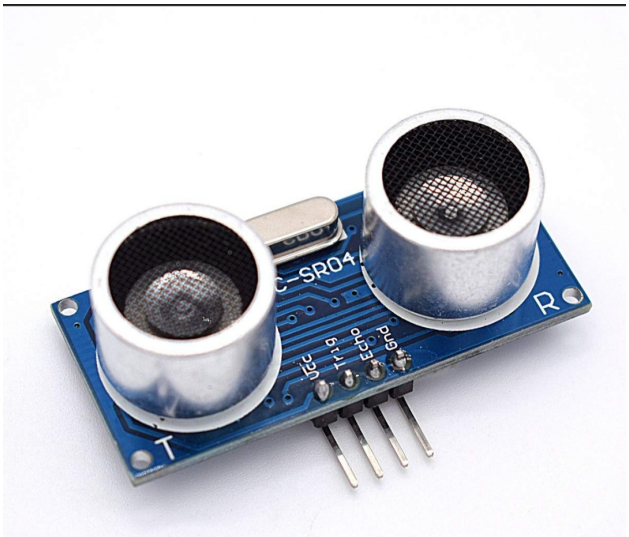

Figure 4: HC-SR04 Ultrasonic Sensor

Back, Left, Right.

3.2 Voice Control using Arduino Bluetooth Controller

1. Install the *Arduino Bluetooth Controller* app.
2. Upload the voice control code to ESP32 using PlatformIO: `https://github.com/gadepall voice.cpp`
3. Connect via Bluetooth and select the "Voice Control" mode.
4. Use commands: *Left, Right, Forward, Back, Stop.*

3.3 Obstacle Detection with Ultrasonic Sensor

The ultrasonic sensor detects obstacles by emitting sound waves and measuring the time it takes for the echo to return. If the distance is below a threshold (e.g., 20 cm), the ESP32 automatically stops or diverts the car to avoid collision.

Listing 1: Sample Ultrasonic Code Snippet

```
#define TRIG 5
#define ECHO 18

void loop() {
```

```
  long duration;
  int distance;
  digitalWrite(TRIG, LOW);
  delayMicroseconds(2);
  digitalWrite(TRIG, HIGH);
  delayMicroseconds(10);
  digitalWrite(TRIG, LOW);
  duration = pulseIn(ECHO, HIGH);
  distance = duration * 0.034 / 2;

  if (distance < 20) {
    stopCar(); // Stop motors
  } else {
    moveForward();
  }
}
```

4 Conclusion and Future Work

We have successfully built a voice-controlled UGV equipped with an ultrasonic sensor for obstacle detection. This system demonstrates an efficient low-cost model for AI-based control and interaction. Future improvements include integrating onboard speech recognition and path-planning algorithms directly on the ESP32 to eliminate the need for external servers.