Voice Based UGV control

Gajjarapu Satyanarayana and G.V.V. Sharma
Department of Electrical Engineering,
Indian Institute of Technology Hyderabad,
Kandi, India 502284
gadepall@ee.iith.ac.in

***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. In the process, we show how this platform offers a low-cost alternative for porting artifical intelligence (AI) algorithms on hardware.**

I. Introduction

Autonomous navigation has been a major area of research in robotics, with pioneering projects such as Stanley, which won the DARPA Grand Challenge [1], and Boss and Junior, which competed in the DARPA Urban Challenge [2], [3], demonstrating autonomous navigation in complex environments. End-to-end learning approaches, such as NVIDIA's system for self-driving cars [4], have further simplified navigation pipelines by mapping sensor inputs directly to control outputs. Surveys on intelligent vehicles highlight a wide variety of autonomous driving applications [5], and research on fully autonomous systems explores both the hardware and software required for robust navigation [6]. In parallel, speech-based human-robot interaction has enabled intuitive control of robots in constrained environments, including intelligent wheelchairs and mobile robots [7], [8], [10], and robust speech recognition datasets such as Google's Speech Commands [9] have accelerated development of voice-controlled systems. Inspired by these high-level projects, this work presents a scaled-down prototype using an ESP32 microcontroller and an L293D motor driver IC to build a voice-enabled toy car, integrating simple navigation with bluetooth control and speech commands for user interaction.

II. List of Components

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

Item	Qty.	Description
UGV kit	1	For assembling the toycar chassis.
ESP32	1	Microcontroller used for control and wireless communication.
L293D Motor Driver IC	1	For driving and controlling the DC motors.
Power Bank	1	Provides portable power supply to the system.
DC Motors	2	Used for propulsion of the toy car.
Breadboard	1	For making circuit connections.
Jumper Wires	11	For making electrical connections between components.
Micro-USB cable	1	Connection between the ESP32 and the power bank.

TABLE II.1
List of Components

III. Hardware Setup

III.1 Assemble the chassis, fix the motors and mount the wheels to build the toycar.

III.2 Fix the breadboard on the base of the toycar.

III.3 Plug the **L293D** motor driver IC in Fig. III.3 on the breadboard.

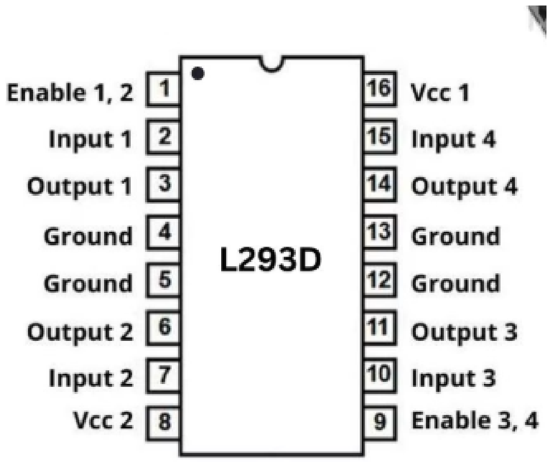

Fig. III.3. L293D Motor Driver IC

III.4 The connections between the L293D output pins and the motors (M_1, M_2) are according to Table III.4

L293D IC	3	6	11	14
Motors	M_1 (+)	M_1 (-)	M_2 (+)	M_2 (-)

TABLE III.4
L293D & Motors connections

III.5 Connect any 4 GPIO pins (**Ex**: 25, 26, 33 & 32) of **ESP32** in Fig. III.5 to L293D inputs

III.6 The connections between the ESP32 and the L293D input pins are according to Table III.6

ESP32	32	33	25	26
L293D IC	3	6	11	14

TABLE III.6
L293D & ESP32 connections

III.7 Connect the ground pins of the L293D IC and the ESP32 to a common ground on the breadboard.

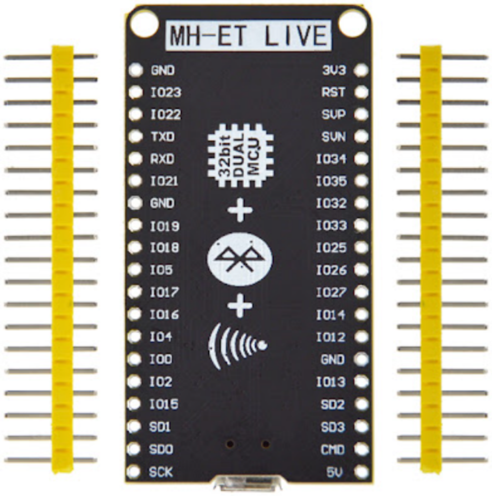

Fig. III.5. ESP 32

III.8 Connect the 5V pin of the ESP32 to the VCC 1 pin of the L293D IC.

IV. Implementation

A. *Dabble*

IV.1 Install **Dabble** app using Google Playstore in an Android mobile.

IV.2 Upload the following code to the ESP32 using any IDE.

```
wget https://github.com/Satyanarayana-123456/UGV_toycar/blob/main/codes/dabble_gamepad.cpp
```

IV.3 After uploading the above code, plug the ESP32 to a power bank via a micro-USB cable.

IV.4 Open the Dabble app and connect to the ESP32 via bluetooth. The app interface looks like Fig. IV.4

IV.5 Now use the **Gamepad** of the app in Fig. IV.5 to control the toycar.

IV.6 Operate the left-side control buttons labeled *Forward, Back, Left & Right* to give the respective commands.

B. *Arduino Bluetooth Controller*

IV.7 Install **Arduino Bluetooth Controller** app using Google Playstore in an Android mobile.

IV.8 Upload the following code to the ESP32 using any IDE.

```
wget https://github.com/Satyanarayana-123456/UGV_toycar/blob/main/codes/ABC_voice.cpp
```

IV.9 After uploading the above code, plug the ESP32 to a power bank via a micro-USB cable.

IV.10 Open the Arduino Bluetooth Controller app and connect to the ESP32 via bluetooth. The app interface looks like Fig. IV.10

IV.11 Now use the **Voice Control** section of the app to control the toycar.

IV.12 The commands which the voice control takes are *Left, Right, Forward, Back & Stop*.

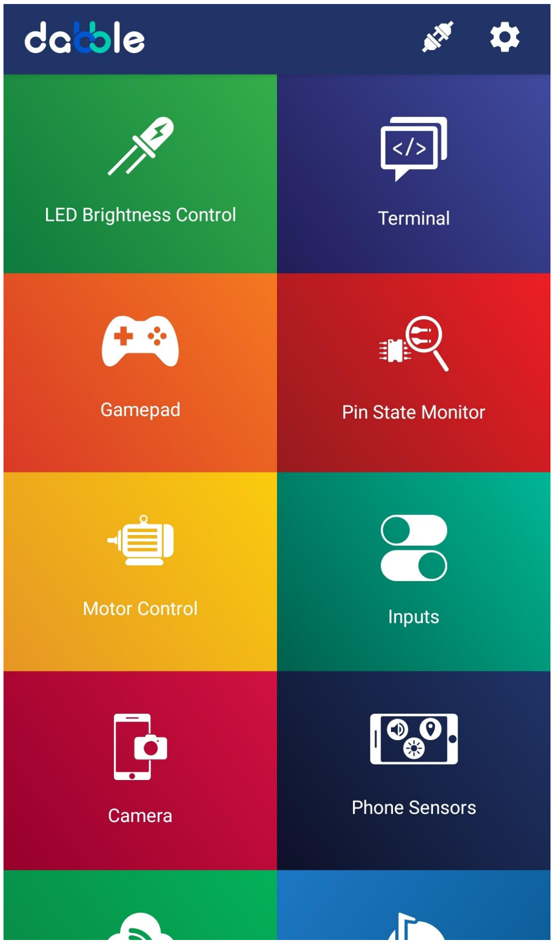

Fig. IV.4. Dabble Interface

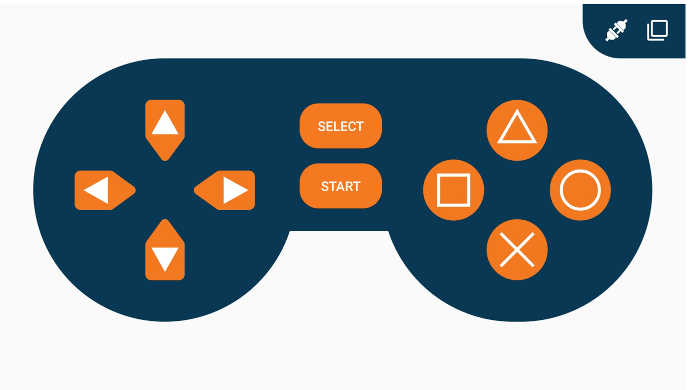

Fig. IV.5. Gamepad in Dabble App

References

[1] S. Thrun et al., “Stanley: The robot that won the DARPA Grand Challenge,” *Journal of Field Robotics*, vol. 23, no. 9, pp. 661–692, 2006.

[2] C. Urmson et al., “Autonomous driving in urban environments: Boss and the Urban Challenge,” *Journal of Field Robotics*, vol. 25, no. 8, pp. 425–466, 2008.

[3] M. Montemerlo et al., “Junior: The Stanford entry in the Urban Challenge,” *Journal of Field Robotics*, vol. 25, no. 9, pp. 569–597, 2008.

[4] M. Bojarski et al., “End to end learning for self-driving cars,” *arXiv*

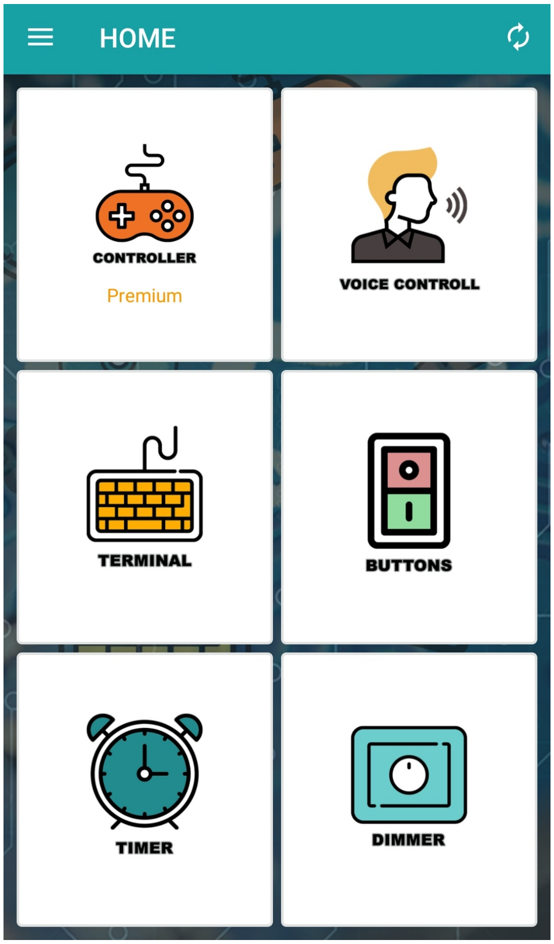

Fig. IV.10. Arduino Bueetoth Controller Interface

preprint arXiv:1604.07316, 2016. [Online]. Available: https://arxiv.org/abs/1604.07316

[5] R. Bishop, "A survey of intelligent vehicle applications worldwide," *IEEE Intelligent Vehicles Symposium*, vol. 15, no. 1, pp. 113–122, 2000.

[6] J. Levinson, J. Askeland, J. Becker et al., "Towards fully autonomous driving: Systems and algorithms," in *Proc. IEEE Intelligent Vehicles Symposium*, 2011, pp. 163–168.

[7] G. Li, S. Mei, H. Yang et al., "Speech interaction with robots: A review," *International Journal of Advanced Robotic Systems*, vol. 14, no. 6, pp. 1–15, 2017.

[8] P. R. K. Prasad, V. G. Kumar, and R. R. Reddy, "Voice-controlled robotic vehicle," *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, vol. 2, no. 6, pp. 2723–2728, 2013.

[9] P. Warden, "Speech commands: A dataset for limited-vocabulary speech recognition," *arXiv preprint arXiv:1804.03209*, 2018.

[10] A. Vasudevan and R. Siegwart, "Speech-based human–robot interaction for assistive robots," *Robotics and Autonomous Systems*, vol. 58, no. 7, pp. 881–888, 2010.