Ball following Robot using ESP32-cam & Arduino UNO

Suujay Dhhoka
Aaron Chandekar
Abhirup Bhattacharjee

Electronics and Telecommunication dept,
Vishwakarma Institute of Information Technology,
Pune, India

Vijaya Aher ma'am

E&Tc project mentor
Vishwakarma Institute of Information Technology,
Pune, India

---***---

Abstract- **This article discusses a robot that chases a ball (object). It offers a detailed summary of a few of the most recent research projects in this field. There are still many questions that need to be investigated in this busy area of study. Robotic arms come in a variety of various varieties nowadays. Some of them are extremely repeatable and accurate. In this paper, we look at the numerous characteristics and applications of an arm. Future research studies may benefit from the information and suggestions provided by this survey. The report concludes with some work that is proposed and research gaps. Object following robots are used in a variety of places, including homes, offices, hospitals, and even in the military.**

Keywords- ***arduino UNO, L293D driver, ESP32-cam, IR sensors, Li-ion batteries, DC motor driver, chassis, arduino IDE (code)***

INTRODUCTION

An autonomous robot that can monitor, detect, and follow a ball's movement in real time is called a ball following robot. The robot is made to behave like a human player in a ball game, like basketball or soccer. This robot's primary function is to provide entertainment and enjoyment, but it may also be an important research tool for examining how humans and robots interact as well as robotics control. Design: Both hardware and software elements go into the creation of a ball-following robot. The robot's movement is controlled by motors, which drive the robot, and a microcontroller, sometimes known as a single-board computer, which processes the camera data and manages the motors. The software component involves algorithms for ball detection and tracking, motor control, and decision making.

Ball Detection: A crucial part of the ball-following robot is the ball detection algorithm. It entails analysing the camera's video stream to determine the position and dimensions of the ball. The algorithm needs to be strong enough to find the ball even in poor illumination and busy backgrounds.

Ball tracking: The tracking algorithm is in charge of monitoring the ball's motion after its location has been determined. The robot's movement must be coordinated with the tracking algorithm's ability to forecast the trajectory of the ball. It must also manage circumstances in which the ball is travelling quickly or abruptly changing direction.

Motor Control: The motor control algorithm regulates the robot's motion in accordance with the position and trajectory of the ball. To maintain the ball in its range of vision, the robot's speed, direction, and turning radius must all be able to be altered.

Decision-making: Based on the position and trajectory of the ball, the decision-making algorithm must choose the optimal course of action. It must make decisions on how quickly the robot should move, stop, and turn.

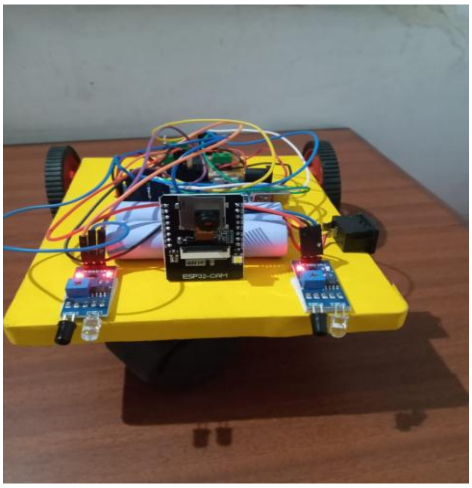

Figure 1. A ball following robot

A. *Hardware*

Arduino UNO

An open-source platform called Arduino is utilised to create electrical projects. Integrated Development Environment (IDE), which runs on the system, is used to develop and upload system code to the physical board. Arduino is made up of both a physical programmable circuit board (often referred to as a microcontroller) and this software.

It is simpler to learn to programme thanks to the Arduino IDE's use of a simplified version of C++. The micro-controller's functionalities are separated into a more usable container by using a standard form factor that Arduino offers. An excellent option for novices is the Uno, one of the Arduino family's more well-liked boards. [9]

A microcontroller is included on a pre-built Arduino board, and the Arduino programming language is used to programme it, with the Arduino development setting. This platform essentially offers a means of creating and programming electronic parts. Arduino's "sketches," which make use of fundamental programming constructs like variables and functions, are the basis for its simplified version of the C/C++ programming language. Following that, these are turned into a C++ programme.[10]

One of Arduino's standard boards is the UNO. The Italian word UNO here is for "one." To identify the initial release of the Arduino Software, it was given the moniker UNO. It was also the first USB board that Arduino had ever released. It is regarded as a strong board that is employed in many projects. The Arduino UNO board was created by Arduino.cc.

The ATmega328P microprocessor is the foundation of the Arduino UNO. Compared to other boards, like the Arduino Mega board, etc., it is simple to use. The board is made up of shields, various circuits, and digital and analogue Input/Output (I/O) pins. The Arduino UNO has 14 digital pins, a USB port, a power jack, and an ICSP (In-Circuit Serial Programming) header in addition to 6 analogue pin inputs. It is programmed based on IDE, which stands for Integrated Development Environment. It can run on both online and offline platforms.

All of the Arduino boards that are available share the same IDE.

L293D Motor Driver IC

Figure 3. L293D Motor Driver

The L293D is a 16-pin motor driver IC that has the ability to simultaneously operate two DC motors in either direction. The L293D can deliver bidirectional drive currents up to 600 mA (per channel) at voltages ranging from 4.5 V to 36 V (at pin 8!). It can be used to control toy motors, which are miniature dc motors. It may occasionally be swelteringly hot.

Even the most basic robot needs a motor to turn a wheel or carry out a certain task. In order to drive a motor further, you need some kind of switch that can absorb a tiny current, amplify it, and provide a greater current because motors demand more current than the usual microcontroller pin can produce. A "motor driver" is the one who completes the full operation. That work is made simple by the L293D Motor Driver IC, which has assisted in a number of applications with reasonable ease.

The most popular driver for applications involving bidirectional motor driving is the L293D H-bridge driver. DC motors may be driven in either direction thanks to the L293D IC. A pair of two DC motors can be controlled simultaneously in any direction using the 16-pin IC L293D. It means that a single L293D IC may operate two DC motors. since it contains two H-Bridge Circuits. The L293D is also capable of driving large, quiet motors. An H-bridge motor control circuit can be created in a number of ways, including by employing relays, transistors, and L293D/L298. Before getting into the specifics, let's first define an H-Bridge circuit.

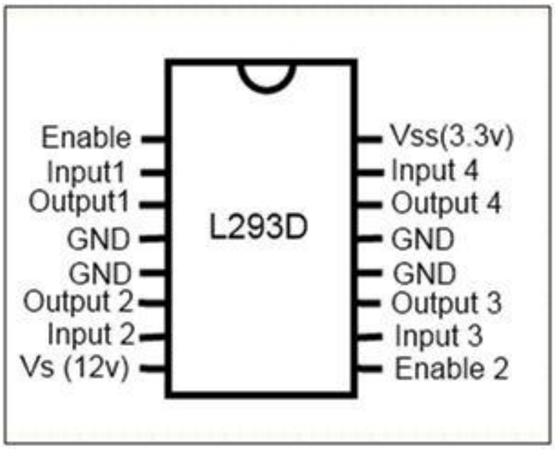

Figure 4. pin diagram of L293D Motor Driver

ESP32-Cam

The ESP32-CAM is a full-featured microcontroller that also has an integrated video camera and microSD card socket. It's inexpensive and easy to use, and is perfect for IoT devices requiring a camera with advanced functions like image tracking and recognition.

Figure 5. ESP32-Cam

The ESP32-CAM is based upon the ESP32-S module, so it shares the same specifications. It has the following features:

- 802.11b/g/n Wi-Fi
- Bluetooth 4.2 with BLE
- UART, SPI, I2C and PWM interfaces
- Clock speed up to 160 MHz
- Computing power up to 600 DMIPS
- 520 KB SRAM plus 4 MB PSRAM
- Supports WiFi Image Upload
- Multiple Sleep modes
- Firmware Over the Air (FOTA) upgrades possible
- 9 GPIO ports
- Built-in Flash LED

With one significant exception, utilising the ESP32-CAM is very similar to using the ESP32 modules we previously examined. You cannot just connect the ESP32-CAM board to your computer and begin loading programmes because it lacks a USB connector.

You must instead include an external FTDI converter. Since you would use this adapter to programme an Arduino Pro Mini, chances are good that you already own one if you've dealt with the Pro Mini.

IR Sensor

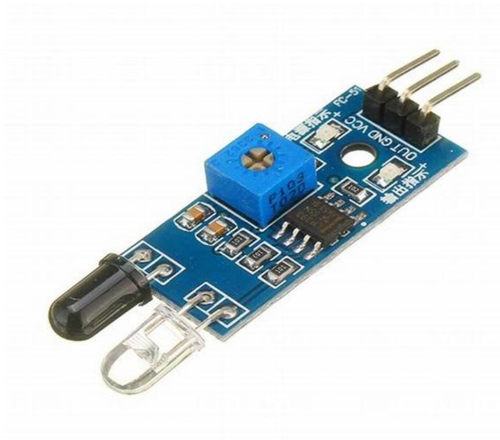

Figure 6. Infrared sensor

An electronic gadget known as an IR sensor emits light in order to detect nearby objects. An IR sensor can monitor an object's heat while also spotting movement. Typically, all items emit some kind of thermal radiation in the infrared range. Although these kinds of radiations are invisible to the human eye, infrared sensors can pick them up.

The detector is merely an IR photodiode, while the emitter is merely an IR LED (Light Emitting Diode). The same wavelength of infrared light that an IR LED emits may be detected by a photodiode. The resistances and output voltages when IR light strikes the photodiode will vary proportionally to the intensity of the IR light received.

An infrared source, a transmission medium, an optical component, infrared detectors or receivers, and signal processing are the five fundamental components of a conventional infrared detection system. Infrared sources include infrared lasers and LEDs of a certain wavelength.

Chassis, Wheels & DC Motors, Batteries

A robotics project's primary foundation is a robot chassis. We offer a wide variety of chassis with tank tracks or four wheel drive (4WD).

The simple and quick configuration of the robot chassis makes it possible to quickly construct a mobile robotics platform. When you don't have the time or resources to build your own robotics chassis, they are the ideal alternative.

The majority of robot chassis come with numerous pre-drilled holes and slots that make it easy to attach additional components, such as sensors or LiDar, quickly.

We have attached two wheels on either side of the robot. They are attached with the DC motors which are powered by the Li-ion batteries in the robot.

There are more and more battery power alternatives available to robot designers. The largest energy densities are available from Li-ion, whereas LiFePO4 offers advantages in terms of environmental toughness. There are more options now, such as recycled Li-ions, which could offer less expensive replacements for brand-new batteries.

Additionally, the robot has a caster wheel at the front that enables for 360-degree movement. Robots can now move freely and take turns with ease.

B. Software

Arduino IDE (code)

The advent of the Arduino software (IDE) and arduino boards (hardware) has made the development of electronics simpler. With the aid of additional parts, this set facilitates the construction of digital and interactive gadgets. Previously, we discussed Arduino boards. In this tutorial, we'll define Arduino software (IDE) and explain how to utilise it.

The Arduino software (IDE), developed by arduino.cc, is an integrated development environment that is open source and used to code the Arduino boards. Permit programming and uploading to Arduino boards. It also included a number of libraries and a collection of sample mini-projects.

The C/C++ programming languages are supported by the Arduino software (IDE), which is compatible with multiple operating systems (Windows, Linux, and Mac OS X).

Beginners and more experienced users can both easily utilise the Arduino software. It is used to create interactive prototypes and get started with robotics and electronics programming.

Arduino software is therefore a tool for creating new things. by Anyone (children, hobbyists, engineers, programmers, etc.) and develop new electronic creations.

Here is a code for ball following robot using a pi-cam:

```
#include <SPI.h>
#include <Pixy.h>
Pixy pixy;

/////////////////////////////////////////////////////
////

// ENA IN1 IN2 IN3 IN4 ENB
int myPins[6] = {5, 6, 7, 8, 9, 10};
float deadZone = 0.15;
//int baseSpeed = 130;

/////////////////////////////////////////////////////
////

int cont = 0;
int signature, x, y, width, height;
float cx, cy, area;

void setup() {
 Serial.begin(9600);
 Serial.print("Starting...n");
 pixy.init();
 for (int i = 0; i < 6; i++) {
  pinMode(myPins[i], OUTPUT);
 }
}

void loop() {
 float turn = pixyCheck();
 if (turn > -deadZone && turn < deadZone) {
  turn = 0;
 }
 if (turn < 0) {
  moveRobot(-80, 170);
 }
 else if (turn > 0) {
  moveRobot(170, -80);
 }
 else {
  moveRobot(70, 70);
 }
 delay(1);
}


float pixyCheck() {
 static int i = 0;
 int j;
 uint16_t blocks;
 char buf[32];
 blocks = pixy.getBlocks();

 if (blocks)
 {
  signature = pixy.blocks[0].signature;
  height = pixy.blocks[0].height;
  width = pixy.blocks[0].width;
  x = pixy.blocks[0].x;
  y = pixy.blocks[0].y;
  cx = (x + (width / 2));
  cy = (y + (height / 2));
  cx = mapfloat(cx, 0, 320, -1, 1);
  cy = mapfloat(cy, 0, 200, 1, -1);
  area = width * height;

 }
 else {
  cont += 1;
```

```
  if (cont == 100) {
   cont = 0;
   cx = 0;
  }
 }
 return cx;
}

float mapfloat(long x, long in_min, long in_max, long out_min, long out_max)
{
 return (float)(x - in_min) * (out_max - out_min) / (float)(in_max - in_min) + out_min;
}


void moveRobot(int leftSpeed, int rightSpeed)
{
 if (leftSpeed >= 0) {
  digitalWrite(myPins[1], 0);
  digitalWrite(myPins[2], 1);
 }
 else {
  digitalWrite(myPins[1], 1);
  digitalWrite(myPins[2], 0);
 }

 if (rightSpeed >= 0) {
  digitalWrite(myPins[3], 0);
  digitalWrite(myPins[4], 1);
 }
 else {
  digitalWrite(myPins[3], 1);
  digitalWrite(myPins[4], 0);
 }

 analogWrite(myPins[0], abs(leftSpeed));
 analogWrite(myPins[5], abs(rightSpeed));
}
```

SIMPLIFICATIONS

Some of the simplifications we make throughout the text are not mentioned. The first is that we only looked at left-to-right motions and not forward-backward ones. Even though the ball and robot are separated by a constant distance L, we still consider the movement to be linear because the ball must move with the rotating robot around the circle. We can assume that at a certain distance L, the difference is negligibly different. Next, the robot model assumes a pure integrator that responds instantly to signals. This isn't true since the robot's inertia mass m causes the replies to be delayed.

We can ignore the robot's relatively slow motions because it weighs only 500 grammes when batteries and a camera are included. Furthermore, unlike what we silently believed, a genuine robot's movement is not linearly dependent on the input signal to the motor. Furthermore, because the motor power is limited, we are unable to exceed specific boundaries. Once more, for relatively tiny variations from the stable position close to zero, we can ignore this issue. The camera processing model is the primary distinction between the model and the actual robot.

In actuality, the qualities of the acquired image are different, and there are many variables that affect how it is analysed. When there are more items in the image, it is not difficult to imagine that the ball recognition process will take significantly longer. Each time an image is processed, the amount of time required varies depending on the time delay TD model. By just considering the highest value of TD, we can get around this issue. The oscillations will not be seen in the real robot, but they all have slightly varied periods. By employing as homogeneous of environmental conditions as possible, this can be prevented.

CONCLUSION

We have examined 15 years' worth of articles and spoken with numerous robot parameters. Even though artificial intelligence techniques are said to be "more clever" than conventional methods, in our example, the conventional way provides more information than just a description of the system. Additionally, we are able to determine the exact values of the crucial system parameters and anticipate the behaviour of the system. A ball-following robot can be used for many different things and can be of many various sorts or upgrades. We then talked about research gaps and obstacles, how it might act as a guide for future study, and how we might work to improve a ball-following robot by creating effective algorithms and simulations.

REFERENCES

[1] Lúcny, A.: Building Control System of Mobile Robots with Agent-Space Architecture. In: CLAWAR/EURON Workshop, Vienna, 2004.

[2] Lúcny, A.: Real-time Cognition. In: Cognition and Artificial Life'09 (Kelemen, J. - Kvasnicka, V. - Rybar, J. eds.), SLU Opava, 2009. In Slovak.

[3] Padé approximant. Wikipedia, The Free Encyclopedia. Retrieved: 10 July 2009, 14:14 UTC. Available on-line: PadC3 - Wikipedia

[4] Kumar, A., Patra, R., Manjunatha, M., Mukhopadhyay, J., & Majumdar, A. K. (2011, January). An electronic travel aid for navigation of visually impaired persons. In Communication systems and networks (COMSNETS), 2011 third international conference on (pp. 1-5). IEEE.

[5] Kumar, M. N., & Usha, K. (2013). Voice Based Guidance and Location Indication System for the Blind Using GSM, GPS and Optical Device Indicator. International Journal of Engineering Trends and Technology (IJETT), 4(7), 3083-3085.

[6] Bahadir, S. K., Koncar, V., & Kalaoglu, F. (2012). Wearable obstacle detection system fully integrated to textile structures for visually impaired people. Sensors and Actuators A: Physical, 179, 297-311.

[7] Tanveer, M. S. R., Hashem, M. M. A., & Hossain, M. K. (2015, December). Android assistant EyeMate for blind and blind tracker. In Computer and Information Technology (ICCIT), 2015 18th International Conference on (pp. 266-271). IEEE.

[8] Sakhardande, J., Pattanayak, P., & Bhowmick, M. (2012). Smart cane assisted mobility for the visually impaired. World Academy of Science, Engineering and Technology, 70.

[9] Arduino- https://learn.sparkfun.com/tutorials/what-is-an-arduino

[10] Arduino- https://www.techopedia.com/definition/27874/arduino

[11] Jayantilal, S. H. Interfacing of AT Command based HC-05 Serial Bluetooth Module with Minicom in Linux.

[12] Arduino-https://arduino-info.wikispaces.com/BlueTooth-HC05- HC06-Modules-How-To.html

[13] Gaikwad, P. V., & Kalshetty, Y. R. (2015). Bluetooth Based Smart Automation System Using Android. International Journal of Science and Research.

[14] Bluetooth-https://www.rhydolabz.com/wireless-bluetooth-ble-c-130_132/hc05-bluetooth-module-masterslave-p-1169.html

[15] Ayushi Y Vadwala, Krina A Suthar, Yesha A Karmakar and Nirali Pandya. Survey paper on Different Speech Recognition Algorithm: Challenges and Techniques. International Journal of Computer Applications 175(1):31-36, October 2017.

[16] Ms.Ayushi Y. Vadwala, Ms.Krina A. Suthar, Ms.Yesha A. Karmakar, P. N. P. (2017). Intelligent Android Voice Assistant - A Future Requisite. International Journal of Engineering Development and Research, 5(Issue 3), 337-339.