AR BASED TARGET SYSTEM

Project report for ESD ICE 4002

Abstract

To create an augmented reality based real time target identification and shooter system.

SARANSH AGRAWAL-170921060

[Email address]

Project Report

Aim: Create a bot and a wearable device to detect the target and launch a projectile to it.

Part 1: Wearable device

Apparatus: Transparent single colour OLED screen, SSD1309 OLED driver (I2C), Raspberry Pi 3B+, 8MP Sony IMX219 camera, HC-05, MPU-6050.

Fig1: Testing apparatus

Theory: First the camera is used to load the image in the R-Pi using open cv2 library for python, and the frame to it is set to 128X56 ratio. It is done so as to fit the OLED screen dimensions, the target should be within this frame ratio. The target is detected using a colour based identification algorithm, where a pre-set value determines the target.

Fig2:

This target is marked by a single pixel on the screen that moves relative to the target position in real time (seeFig2) the rest of the screen is transparent making it possible for naked eye to see the target through it. The screen is mounted 12cms away from the eye, the current dimensions of the screen makes the frame of object detection low, but there are other methods of improving it, like using a bigger screen or eye projectors. For demonstration a smaller screen is used.

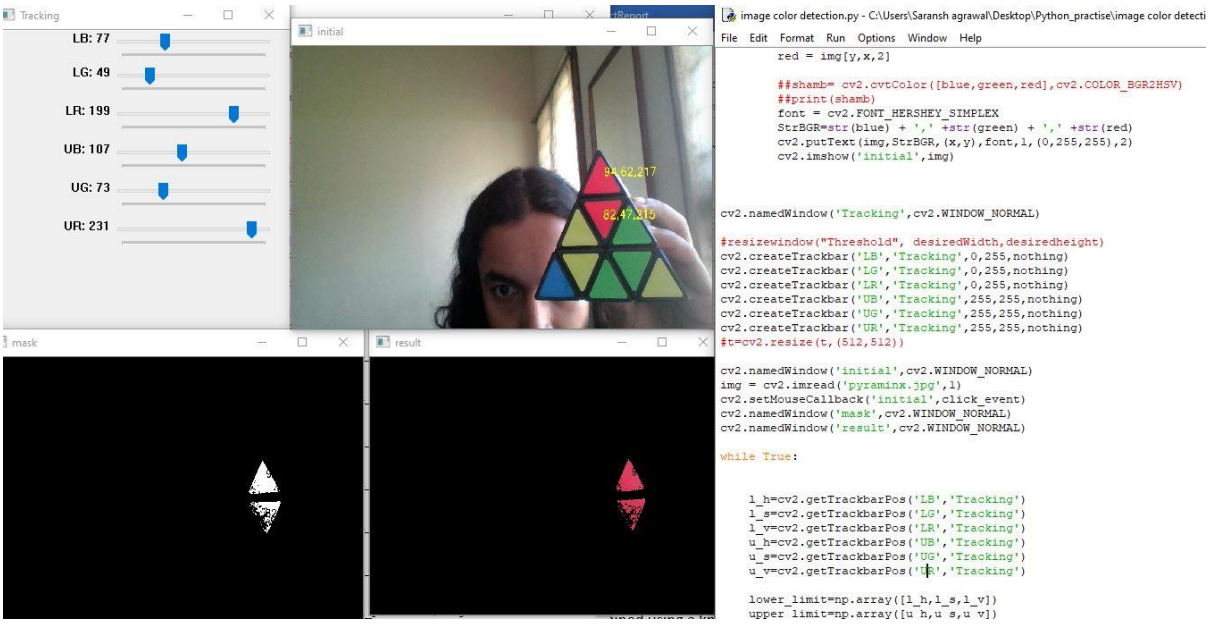

Fig3

Breakdown of fig3: In the “initial” window the image is loaded and right click function marks the “RGB” values for the red colour. Then in the “tracking” window the lower and

upper values are selected to show the visible red regions in the image. In the mask window the threshold selected region is shown and in the result window a bitwise multiplication of the two windows is used to get the red which is specified as target.

The next part of the algorithm is to calculate the distance to the target based triangle similarity:

First focal length of the camera is determined using a known Width (W) object placed at known distance (D) from the camera, clicking a picture of it gives us its width (P) in pixels. Using formula we have focal length.:

F = (P x D) / W

Result of this is 3.04 mm for the camera used in experiment.

Using triangle similarity: object distance in real time (OD) is calculated as follows, with known F:

OD = (W x F) / P

This distance is transferred along with the data from MPU6050 gyroscope through the Bluetooth module to the on ground bot.

The MPU6050 is placed on both the bots and data from them is used to determine the relative position of the Bot and the wearable device from each other using a starting point reference.

Part 2: On ground Bot:

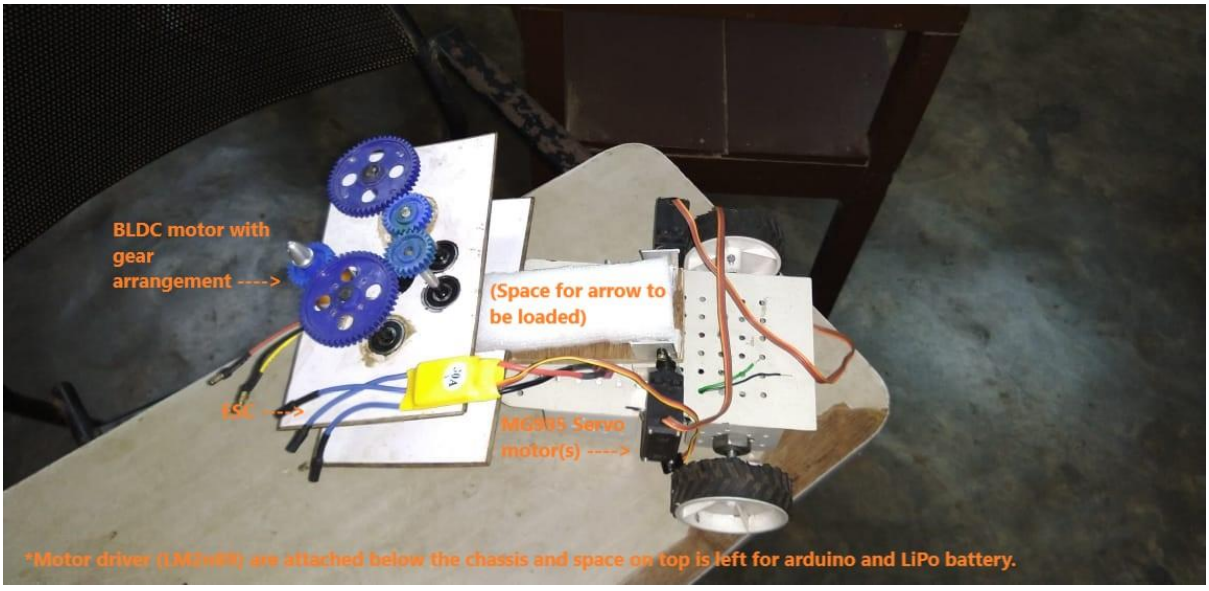

Fig4:

Apparatus: custom made chassis, 2 Tower pro mg995 servo motors, 2 200 rpm geared dc motors with LM298N motor drivers, castor wheel, 5mm shafts, gears, ID 5mm bearings, wheels, 1000 KV BLDC motor with 30A ESC, MPU6050, Arduino UNO, HC-05.

Theory: This bot uses the HC-05 Bluetooth module to receive data from R-Pi. The received data consists of distance of target from eyewear and the distance of Eyewear from Bot (this data is in 3 axis, which also gives us the height of the target) , this data is used to form a triangle and two distances are calculated, first the horizontal distance that the bot has to travel to be in the minimum approachable distance of the target, and the second is the angle that the projectile has to be launched to reach the target.

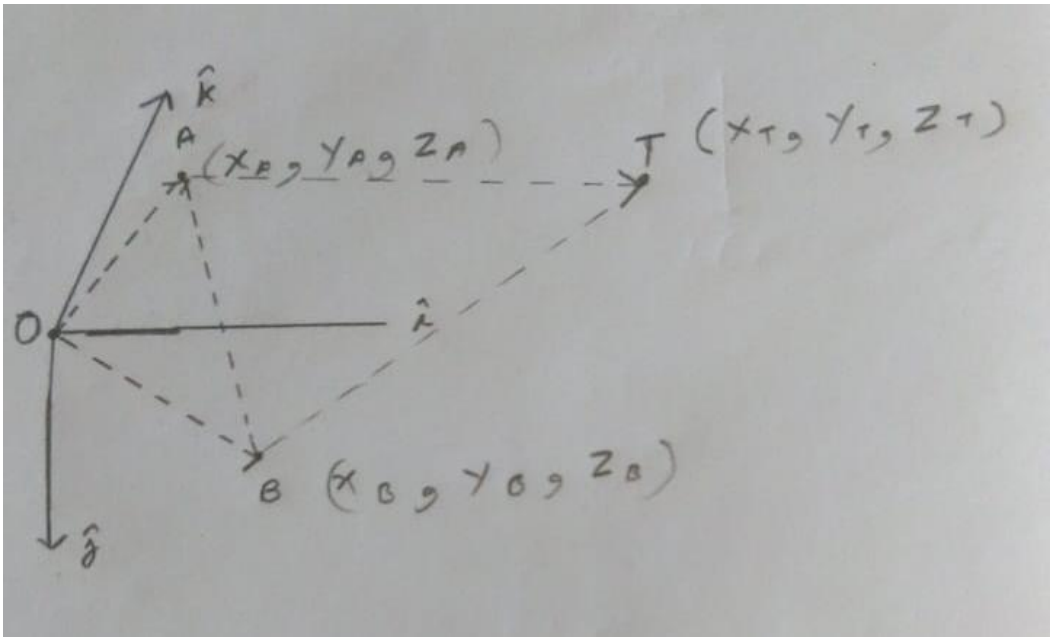

Fig5

In the above figure **A** marks the Eyewear, **B** marks the on ground bot and **T** is the target. Since gyroscope gives us angle and acceleration in x,y,z coordinates, we can calculate the vectors **OA, OB** where **O** is the origin synchronised point. We also know **AT** from the Rpi data. Hence using co-ordinate system **BT** can be found. Hence the distance to target and its elevation is known and the bot can approach the minimum firing distance.