PRESIDENCY UNIVERSITY

Itgalpura, Rajanukunte, Bengaluru - 560064

School of Engineering

A Project Report on

IOT- Based Color Sorting Machine

Submitted in partial fulfillment of the requirement for the course
Innovative Project – Raspberry-Pi using Python (**CSE 1003**)

Submitted - by
Group: IPR-452

Student Name	Roll No
Ronit Pathak	20211CAI0010
Sohith N R	20211CAI0006
Elluri Bhavya Sree	20211CAI00035

Under the supervision of

Guide name: Mrs.Annapurna
Designation: Assistant Professor
Department: ECE-Dept

Dec-2022

Abstract:

- ➔ There is a wide usage of many products in our day-to-day life, and manufacturing of these products are done in many large and small-scale industries. Arranging makes quality consistency an issue.
- ➔ Nowadays the main difficulty that is faced after the production is sorting. Arranging items in an industry is a dull modern process, which is by and large done physically.
- ➔ Consistent manuals the need of this type of machine in the industries will help in sorting the machine according to their weight, size, color, shape, etc.
- ➔ IOT based colored products sorting machine widely used in the candy industry, food industry (grain, fruit) and mining industry.
- ➔ Using this in the candy industry can differentiate the candies according to their color, whereas in the grain industry using this can differentiate the grains based on their color.
- ➔ In the Diamond and mining industry, segregates the precious stones according to their color.
- ➔ This machine arranges the items in particular order as required, so physical work is not needed, i.e., Improves automation and decreases the man work.
- ➔ We are implementing this machine in a much more effective way, using a color sensor (TCS 3200), Raspberry Pi, Servo motors and LCD display. Here we are arranging the setup in a pro-type manner where we can arrange the Servo motors to rotate based on the presented-color.

Hardware/Software Tools used:

Hardware:

- Servo Motor – 2 motors
- Raspberry Pi Module – 4
- Pi camera

Servo motors:

Servo motors or “servos”, as they are known, are electronic devices and rotary or linear actuators that rotate and push parts of a machine with precision. Servos are mainly used for angular or linear position and for specific velocity, and acceleration.

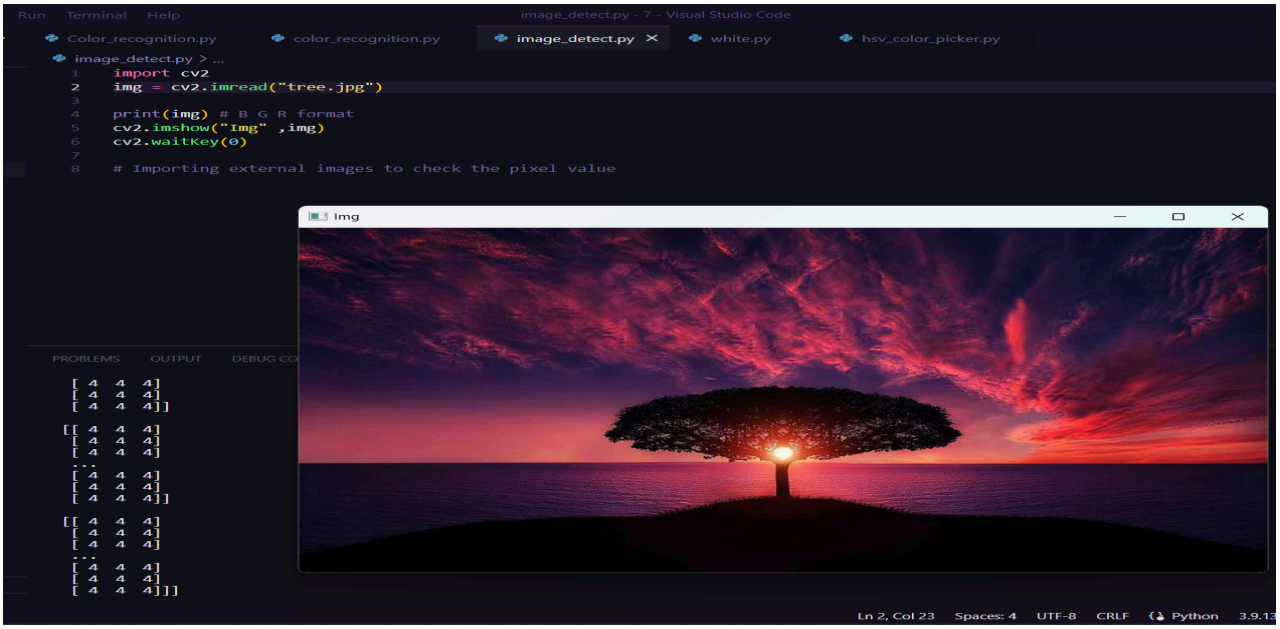
- Input 5 volt is connected to Pin 1.
- Ground is Connected to Pin 3.
- Pin 11 is GPIO connected to the 3rd terminal.

Proto-Type Code for Servo Motor:

```python
from gpiozero import Servo
from time import sleep

myGPIO=17

servo = Servo(myGPIO)
print("Rassberry Pi Servo");
while True:
    servo.min()
    print("min")
    sleep(0.5)
    servo.mid()
    print("mid")
    sleep(1)
    servo.max()
    print("max")

    sleep(1)
```

```python
from gpiozero import Servo
from time import sleep

myGPIO=17

myCorrection=0
maxPW=(2.0+myCorrection)/1000
minPW=(1.0-myCorrection)/1000

servo = Servo(myGPIO,min_pulse_width=minPW,max_pulse_width=maxPW)

while True:

  print("Set value range -1.0 to +1.0")
  for value in range(0,21):
    value2=(float(value)-10)/10
    servo.value=value2
    print(value2)
    sleep(0.5)

  print("Set value range +1.0 to -1.0")
  for value in range(20,-1,-1):
    value2=(float(value)-10)/10
    servo.value=value2
    print(value2)
    sleep(0.5)
```

RGB Detection:



Color -Recognition code:

color_recognition.py

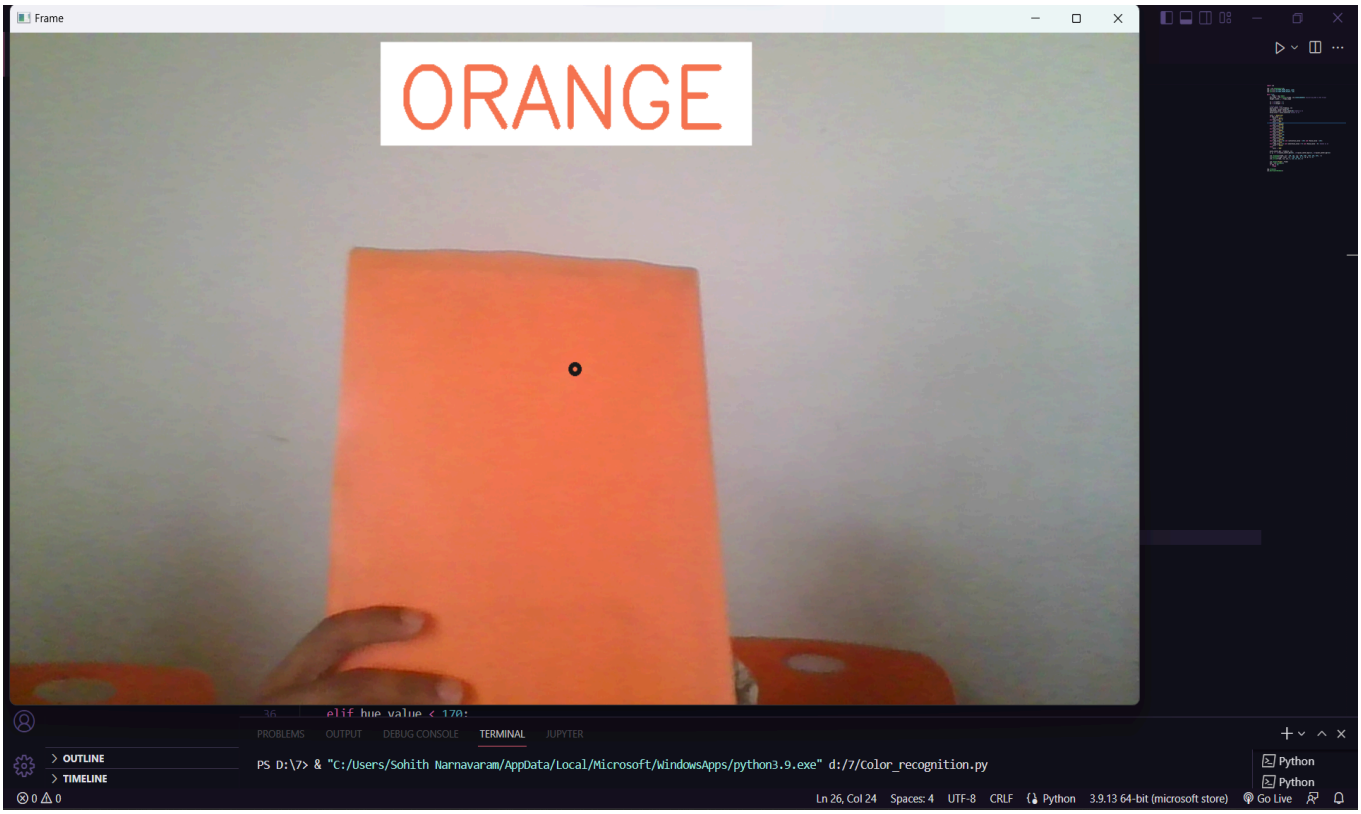
: > Users > Sohith Narnavaram > OneDrive > Desktop > 2nd Year > RasberiPI > YT_codes > simple-color-recognition > simple color rec

```python
import cv2

cap = cv2.VideoCapture(0)
cap.set(cv2.CAP_PROP_FRAME_WIDTH, 1280)
cap.set(cv2.CAP_PROP_FRAME_HEIGHT, 720)

while True:
    _, frame = cap.read()
    hsv_frame = cv2.cvtColor(frame, cv2.COLOR_BGR2HSV) #converting BGR to HSV format
    height, width, _ = frame.shape

    cx = int(width / 2)
    cy = int(height / 2)

    # Pick pixel value
    pixel_center = hsv_frame[cy, cx]
    hue_value = pixel_center[0]

    color = "Undefined"
    if hue_value < 5:
        color = "RED"
    elif hue_value < 22:
        color = "ORANGE"
    elif hue_value < 33:
        color = "YELLOW"
    elif hue_value < 78:
        color = "GREEN"
    elif hue_value < 131:
        color = "BLUE"
    elif hue_value < 170:
        color = "VIOLET"
    else:
        color = "RED"

    pixel_center_bgr = frame[cy, cx]
    b, g, r = int(pixel_center_bgr[0]), int(pixel_center_bgr[1]), int(pixel_center_bgr[2])
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

```
PS D:\7> & "C:/Users/Sohith Narnavaram/AppData/Local/Microsoft/WindowsApps/python3.9.exe" d:/7/Color_recognit
```

Ln 1, Col 1 Spaces: 4



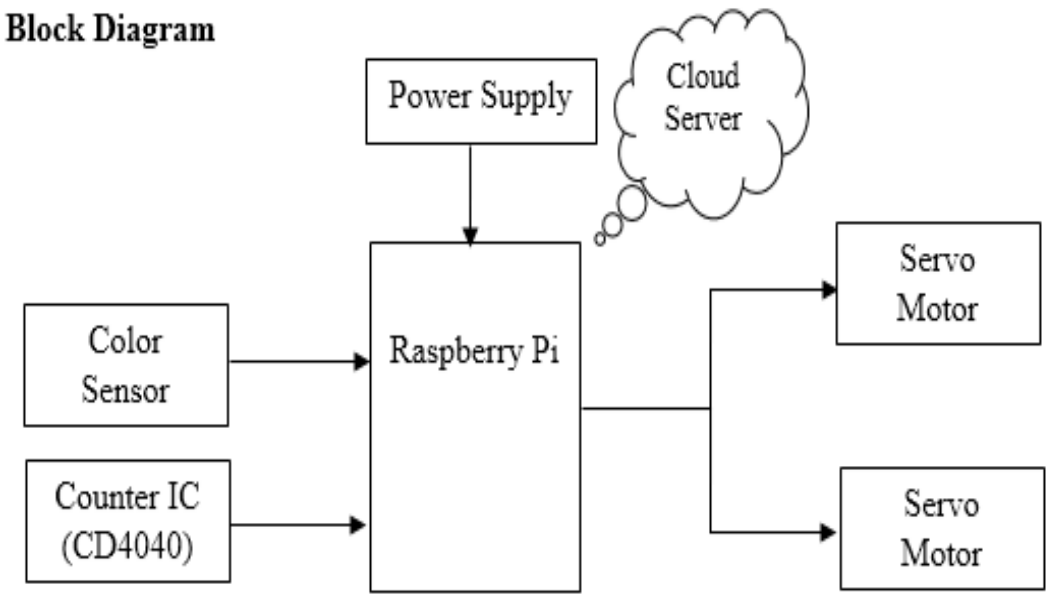
Colorama:

Colorama is a Python library for rendering colored terminal text.

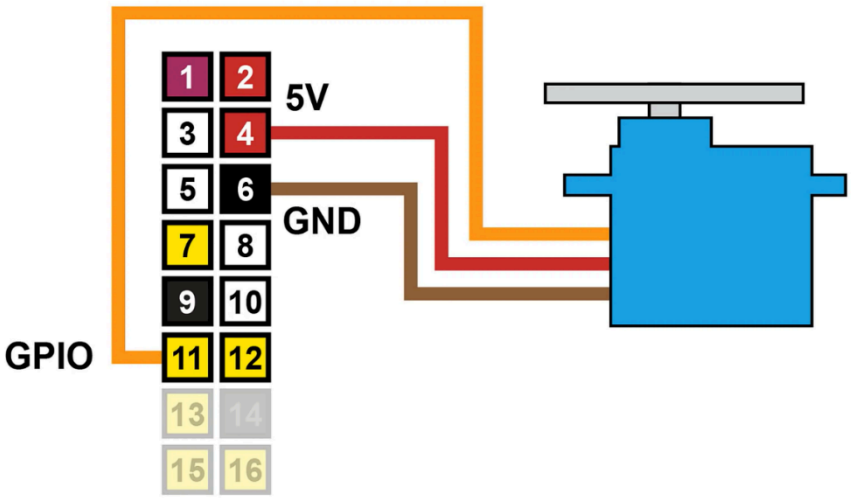
Object's Color Recognition with OpenCV:

- Library used is OpenCV.
- Recognition of the color using a camera in Real Time.
- Defining color using H S V color picker
- Hue - Color itself
- Saturation – Intensity of the color
- Value – Brightness

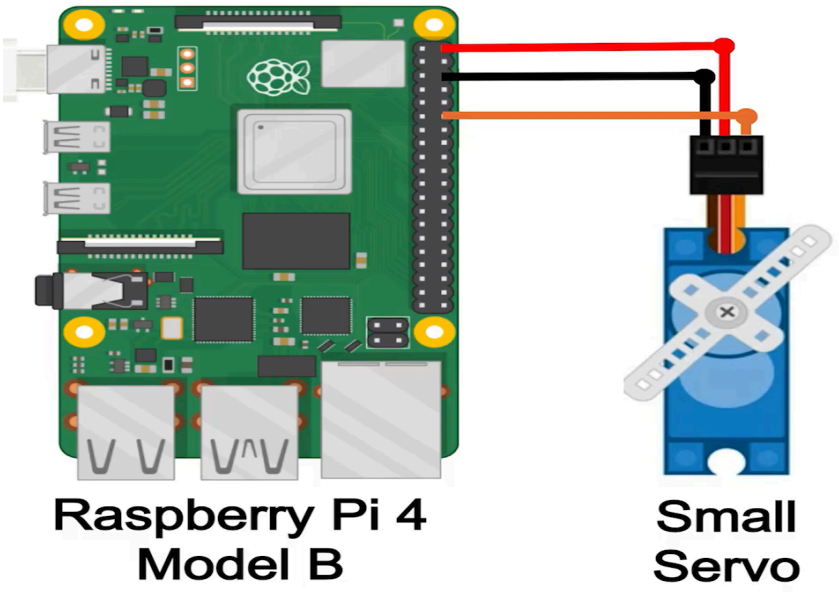
Block diagram:



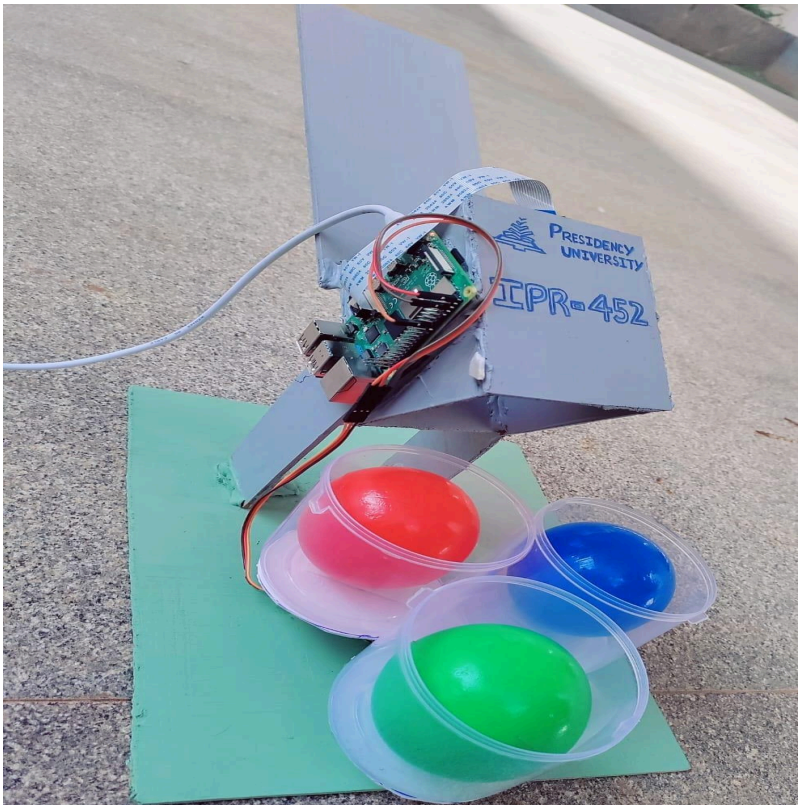
Circuit Diagram of Servo Motor:



Connections with Raspberry - Pi 4 :



Results:



The objects are sorted with respect to their color and dropped into the respective box.

Its advantages are that it is-

- ·Accurate
- ·Good repeatability
- ·Reduce labor cost
- ·Requires Less human interference.

They are mainly used -

- · In fruits and vegetable farming areas (rural areas) where installation of expensive sorters is very difficult.
- · In the food industry to identify rotted fruits and vegetables, in minor scale and big scale productions, to categorize the products established on several factors.
- · In production units to scan and identify the defects in raw materials.

Challenges faced:

- Challenges faced in gaining more information about the hardware, and Software with limited libraries.
- Challenge faced in choosing the appropriate tools and equipment for the project, be it the sensors or the Pi-cam.
- Learning and exploring the Raspberry-Pi as it is a new concept for everyone.
- Finalizing the working model's structure which had to be perfect in order to make it run.
- Finding the exact measurements/angles for the servo motors to rotate and function.

Conclusion:

- The suggested framework will be a demo rendition which gives expense effectiveness, taking less time and technically the easiest way for differentiating objects.
- This framework utilizes Raspberry Pi-4 which makes this model simple to utilize which is more additional effective.
- The main failure will be caused if the sensing of objects according to color is not done. Therefore, it is very important to have proper and checked Apparatus.
- Further, making desirable changes can be used in small-scale and large-scale industries as well.

- When any color from red, green or blue is kept for detection in front of the Pi-cam then the desired color as shown is detected on the screen and the output of the sensing of color is seen.