SEM 1 25/26

SECTION 1

COURSE CODE: MCTA 3202

LECTURER NAME:
ZULKIFLI BIN ZAINAL ABIDIN

WEEK 9: COLOR DETECTION AND ANALYSIS

PREPARED BY:
GROUP 1

NAME	MATRIC NUMBER
AMIR MUIZZUDDIN BIN NORAIDIL RAZMAN	2129579
MUHAMMAD HAIKAL HANIF BIN ABDUL RAZAK	2213297
AHMAD ILHAM BIN ABDUL AZIZ	2112109

INTRODUCTION

Color detection is an important technique in mechatronics and automation systems, widely used in applications such as object sorting, quality control, robotics, and machine vision. By detecting and identifying colors accurately, a system can make intelligent decisions without human intervention. Traditional color detection systems rely on color sensors that measure reflected light intensity, while modern systems employ artificial intelligence (AI) based vision sensors for improved accuracy and robustness.

In this experiment, a color detection system was developed using two approaches: a conventional color sensor and the Gravity HuskyLens AI Vision Camera interfaced with an Arduino Uno. The system detects different colored objects and outputs their corresponding color IDs through serial communication. The performance of the system was observed under different lighting conditions to evaluate accuracy, response time, and reliability. This experiment provides hands-on experience in hardware interfacing, programming, and basic image/video processing concepts in mechatronics systems

MATERIALS

The following hardware and software components were used to design and evaluate a color detection system using both a conventional color sensor and an AI vision camera:

Hardware:

- Arduino Uno microcontroller board
- Color sensor (TCS3200 or equivalent)
- Gravity HuskyLens AI Vision Camera
- Breadboard
- Jumper wires
- RGB LED (optional, for visual indication)
- USB cable for Arduino connection

Software:

- Arduino IDE
- Python (with pyserial library installed)
- DFRobot HuskyLens Arduino Library

EXPERIMENT SETUP

Hardware Connections

- HuskyLens communicates with the Arduino using SoftwareSerial:
    - RX → Arduino pin 4
    - TX → Arduino pin 5
- RGB LED connections:
    - Red pin → Arduino pin 9
    - Green pin → Arduino pin 10
    - Blue pin → Arduino pin 11
- All RGB pins are connected to PWM-enabled pins to allow brightness control.
- Common ground is shared between the Arduino, HuskyLens, and RGB LED.

METHODOLOGY

System Initialization

- Serial communication is initialized at a 9600 **baud rate** for debugging.
- SoftwareSerial communication is initialized for HuskyLens.
- RGB LED pins are configured as **output pins**.
- The RGB LED is initially turned **OFF**.
- The system continuously checks for HuskyLens connection before proceeding.

Color Detection Process

1. The Arduino sends a request to HuskyLens to obtain detection data.
2. If the request fails, the system retries after a short delay.
3. When an object is detected, HuskyLens returns the **trained color ID**.
4. The detected ID is displayed on the Serial Monitor for verification.

RGB LED Control Logic

- Based on the detected color ID:
    - **ID 1** → RGB LED lights up **Red**
    - **ID 2** → RGB LED lights up **Green**
    - **ID 3** → RGB LED lights up **Blue**
- If an unrecognized ID is detected, the RGB LED is turned **OFF**.
- PWM signals using analogWrite() control the LED intensity

<u>CONCLUSION</u>

This experiment successfully demonstrated two different approaches to color detection: a traditional color sensor–based method and an AI vision–based method using the Gravity HuskyLens camera. The color sensor approach allowed direct measurement of RGB values and provided a fundamental understanding of color sensing principles. However, its performance was highly dependent on proper calibration and was sensitive to changes in ambient lighting conditions.

In contrast, the HuskyLens AI Vision Camera exhibited more consistent and robust color detection performance. Once trained, it was able to recognize predefined colors with higher accuracy and faster response time, even under moderate variations in lighting. The built-in machine vision processing reduced the need for complex external data interpretation and simplified system integration.

Overall, the HuskyLens-based approach proved to be more reliable and user-friendly for real-time color detection tasks, while the color sensor method remains valuable for educational purposes and low-cost implementations. Future improvements could include improved lighting control, enhanced calibration procedures, and integration of both sensing methods to further increase detection accuracy and system robustness.