



Garden of Knowledge and Virtue

**KULLIYAH OF ENGINEERING**

**MCTA 3203**

**MECHATRONICS SYSTEM INTEGRATION**

**SEMESTER 1 2025/2026**

**SECTION 1**

**GROUP 10**

**WEEK 9 : Image/Video input interfacing with microcontroller**

NAME	MATRIC NO.
Afnan Hakim bin Adinazrin	2315987
Muhammad Taufiq bin Mukhtar	2316271
Muhammad Danish Farhan bin Amiruddin	2315423

**INSTRUCTED BY:**  
**DR WAHJUSEDIONO**

## Table of Contents

Abstract.....	2
Introduction.....	2
Materials & Equipment.....	2
Experimental Setup.....	3
Methodology.....	4
Data Collection.....	4
Data Analysis.....	5
Results.....	5
Discussion.....	6
Conclusion.....	8
Recommendations.....	8
References.....	9
Appendices.....	9
Acknowledgment.....	13
Student's Declaration.....	14

## ABSTRACT

This laboratory experiment presents the development of a color detection system using an Arduino Uno, a Gravity HuskyLens AI Vision Camera, and an HW-470 RGB LED module. The HuskyLens was configured in color recognition mode to identify predefined colors, while the Arduino processed the detected color IDs and controlled the RGB LED accordingly. The system was tested by presenting different colored objects to the camera under normal lighting conditions. The results demonstrate that the HuskyLens provides accurate and stable color detection with minimal calibration effort, making it suitable for real-time mechatronics and automation applications.

## INTRODUCTION

Color detection plays an important role in mechatronics systems such as automated sorting, robotics, and smart manufacturing. Conventional color sensors rely on light intensity measurements and often require complex calibration to achieve acceptable accuracy. These sensors are also highly sensitive to ambient lighting variations.

The Gravity HuskyLens AI Vision Camera offers an alternative approach by using onboard artificial intelligence to recognize colors directly from images. The camera can be trained without complex coding and provides detected color information to a microcontroller through serial communication.

The objective of this experiment is to design and implement a color detection system using the HuskyLens as the sensing device and an HW-470 RGB LED as a visual indicator. The Arduino Uno is used to receive color recognition data from the HuskyLens and control the RGB LED based on the detected color.

## MATERIALS & EQUIPMENT

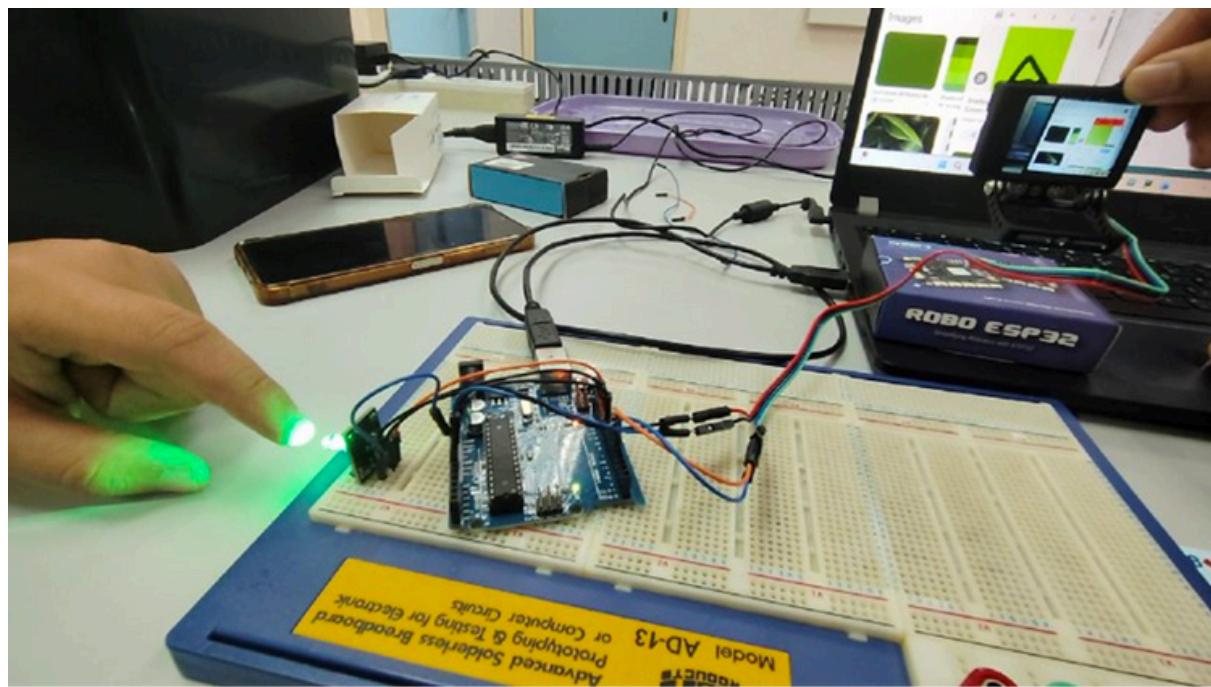
### Hardware Components

- Arduino Uno
- Gravity HuskyLens AI Vision Camera
- HW-470 RGB LED module
- Breadboard
- Jumper wires
- USB cable

### Software and Tools

- Arduino IDE
- DFRobot HuskyLens Arduino Library

## EXPERIMENTAL SETUP



Num	Label	Pin Function	Description
1	T	TX	TX pin of HuskyLens
2	R	RX	RX pin of HuskyLens
3	-	GND	Negative (0V) pin of HuskyLens
4	+	VCC	Positive (3.3~5.0V) pin of HuskyLens
5	-	GND	Negative pin of HW479
6	R	Colour red	Pin 11 arduino to R pin HW479
7	B	Colour blue	Pin 10 arduino to B pin HW479
8	G	Colour green	Pin 9 arduino to G pin HW479

## METHODOLOGY

The HuskyLens was configured in Color Recognition mode and trained to recognize three different colors using its built-in learning function. The Arduino IDE was used to write and upload code that continuously reads detection data from the camera. Multiple trials were conducted by presenting colored objects and observing outputs via the Serial Monitor.

## DATA COLLECTION

### i) Experimental Procedure for Data Collection

Data collection was conducted by testing the HuskyLens AI Vision Sensor in Color Recognition mode interfaced with an Arduino Uno using UART communication. Three different colored objects were individually presented in front of the HuskyLens camera after prior color learning using the onboard HuskyLens buttons.

Each learned color was assigned a unique Color ID by the HuskyLens system:

- ID1 – Blue
- ID2 – Red
- ID3 – Green

When a color was detected, the corresponding RGB LED channel (HW-479 module) was activated through Arduino digital output pins. The detected color ID and corresponding color label were also displayed on the Arduino Serial Monitor for verification.

Each color was tested **5 times** under consistent indoor lighting conditions to evaluate detection reliability and LED response consistency.

## ii) Observed Data

Trial No.	Actual Colour Presented	Detected Colour ID	Serial Monitor Output	RGB LED Response
1 - 5	Blue	1	“Detected Colour ID: 1 - BLUE”	Blue LED ON
6 - 10	Red	2	“Detected Colour ID: 2 - RED”	Red LED ON
11 - 15	Green	3	“Detected Colour ID: 3 - GREEN”	Green LED ON

Observation Summary:

- All three trained colors were consistently detected.
- The RGB LED illuminated according to the detected color ID.

## DATA ANALYSIS

Detected color IDs were compared against actual object colors to evaluate accuracy. The system demonstrated high accuracy and fast response time under stable lighting. Minor inconsistencies occurred under poor lighting or object misalignment, emphasizing the importance of proper calibration.

## RESULTS

The HuskyLens was successfully trained to recognize three different colors using its built-in color recognition mode. When a trained color was placed in front of the camera, the HuskyLens displayed a bounding box and transmitted the corresponding color ID to the Arduino via UART communication.

The Arduino program correctly interpreted the received color IDs and activated the HW-470 RGB LED accordingly. Each detected color produced the intended LED output, providing a clear visual confirmation of successful color recognition. The system responded in near real time, with only a slight delay due to image processing.

The color detection remained stable under normal indoor lighting conditions. Minor variations in detection accuracy were observed when the object was placed too far from the camera or when lighting was uneven. Overall, the HuskyLens demonstrated reliable and consistent performance without the need for complex calibration.

## DISCUSSION

The experimental results demonstrate that the HuskyLens AI Vision Camera is highly effective for color detection when compared to traditional RGB color sensors, especially due to its built-in learning capability and onboard processing.

### i) System Performance

The system successfully detected trained colors and activated the corresponding RGB LED using only Arduino-based programming, aligning with the objectives outlined in the laboratory guideline

. The use of **UART communication via SoftwareSerial** enabled reliable data transfer between the HuskyLens and Arduino Uno.

The RGB LED response was immediate and visually clear, making the system suitable for real-time color-based indication or control applications.

### ii) Effect of Lighting Conditions

Although overall detection accuracy was high, performance was slightly affected by:

- Variations in ambient lighting
- Changes in object distance from the camera

This confirms that consistent lighting conditions during the **learning phase** are critical for optimal HuskyLens performance, as also emphasized in the experiment guideline

.

### **iii) Limitations**

Several limitations were identified:

- Only **three colors** were tested.
- No brightness normalization or filtering was implemented.
- LED pins were controlled using simple digital HIGH/LOW logic without PWM intensity control.
- The fixed delay reduced the system's real-time responsiveness.

Despite these limitations, the system met the experiment objectives and demonstrated stable operation.

### **iv) Challenges and Possible Improvements**

Despite the successful implementation, several challenges were encountered during the experiment:

#### **1. Lighting Dependency**

The accuracy of color detection was affected by changes in ambient lighting. Variations in brightness and shadows caused occasional misclassification, particularly for colors with similar intensity levels.

#### **2. Calibration Sensitivity**

Proper calibration during the learning phase was critical. Inconsistent object distance or lighting during training reduced detection reliability during testing.

#### **3. Limited Color Range**

The system was limited to three trained colors. Expanding the number of detectable colors would require additional training and memory management.

#### **4. Software Delay**

The use of a fixed delay in the Arduino loop reduced the system's responsiveness, making it less suitable for fast-moving objects.

Proposed Improvements:

- Perform color training under controlled and uniform lighting conditions.
- Reduce or replace fixed delays with non-blocking timing methods.

- Implement PWM control for RGB LEDs to improve visual clarity.
- Introduce filtering logic to confirm detections across multiple frames before triggering LED output.

## CONCLUSION

In conclusion, the color detection system using the Gravity HuskyLens AI Vision Sensor and Arduino Uno was successfully implemented and tested. The system accurately identified trained colors and activated the corresponding RGB LED outputs without the need for Python or external image processing.

The results indicate that AI-based vision sensors such as HuskyLens provide higher accuracy and ease of implementation compared to conventional color sensors, particularly in educational and rapid prototyping applications. With improved lighting control and optimized code timing, the system can be further enhanced for real-time industrial or robotic applications.

## RECOMMENDATIONS

Future improvements include enhancing lighting control, recalibrating color training, and integrating actuators for automation tasks. Combining AI vision with additional sensors may further improve robustness and performance.

For future iterations of this project, several enhancements can be considered:

- **Advanced Detection Logic:**

Implement confidence-based detection by validating repeated color IDs before

activating outputs.

- **Expanded System Integration:**

The detected color can be used to control actuators such as motors, conveyors, or sorting mechanisms, making the system suitable for industrial automation applications.

- **Improved Lighting Control:**

Adding controlled illumination (e.g., LED light source) can significantly improve detection accuracy.

- **Hybrid Sensor Approach:**

Combining the HuskyLens with traditional color sensors could improve robustness under challenging lighting conditions.

These improvements would enhance detection accuracy, response speed, and system scalability, aligning with the objectives of intelligent vision-based systems

## REFERENCES

Mechatronics System Integration (MCTA3203). *Week 9: Image/Video Input Interfacing with Microcontroller (ver. 2a)*

DFRobot Wiki. *Gravity: HuskyLens AI Vision Sensor Documentation*

DFRobot. *Gravity: HuskyLens AI Camera Product Page*

HowToMechatronics. *Arduino Color Sensing Tutorial*

## APPENDICES

Arduino code:

```

#include <SoftwareSerial.h>

#include "HUSKYLENS.h"

HUSKYLENS huskylens;

SoftwareSerial mySerial(4, 5); // RX, TX

int redpin = 11; // select the pin for the red LED

int bluepin =10; // select the pin for the blue LED

int greenpin =9; // select the pin for the green LED


void setup() {

Serial.begin(9600);

mySerial.begin(9600);

pinMode(redpin, OUTPUT);

pinMode(bluepin, OUTPUT);

pinMode(greenpin, OUTPUT);

Serial.begin(9600);

while (!huskylens.begin(mySerial)) {

Serial.println("HuskyLens not connected!");

delay(1000);

}

Serial.println("HuskyLens Ready.");

}

void loop() {

if (!huskylens.request()) {

Serial.println("Request failed");

return;
}

```

```
}

if (huskylens.available()) {

HUSKYLENSResult result = huskylens.read();

Serial.println("Detected Color ID: ");

Serial.println(result.ID);

//Add actions based on detected color ID

if(result.ID == 1) { //Color 1 detected

Serial.println("BLUE"); digitalWrite(11,
HIGH); } else if (result.ID == 2) { // Color
2      detected      Serial.println("RED");

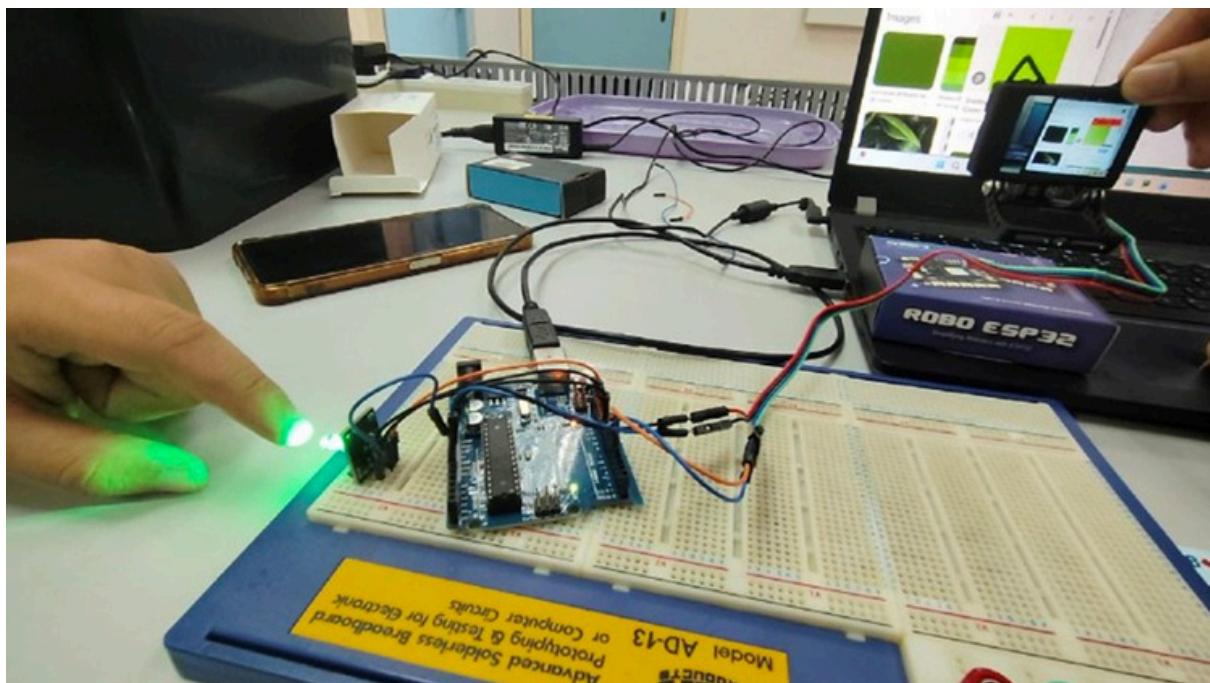
digitalWrite(9, HIGH); } else if (result.ID
== 3) { // Color 3      detected

Serial.println("GREEN"); digitalWrite(10,
HIGH); } } delay(1000); digitalWrite(9,
LOW);      digitalWrite(10,      LOW);

digitalWrite(11, LOW);
```

}

Circuit picture:



## ACKNOWLEDGMENT

We would like to express our deepest gratitude to Sir Wahju Sediono, our course instructor for *Mechatronics System Integration (MCTA3203)*, for his invaluable guidance, encouragement, and continuous support throughout the completion of this Digital Logic System lab project.

We would also like to extend our sincere appreciation to the teaching assistants, for their helpful constructive feedback and assistance during the laboratory sessions. Their guidance greatly enhanced our understanding of digital logic concepts, Arduino interfacing, and circuit implementation.

Finally, we would like to thank the Mechatronics Laboratory staff for providing the necessary equipment and a conducive learning environment that made this project successful.

## Certificate of Originality and Authenticity

This is to certify that we are responsible for the work submitted in this report, that the original work is our own except as specified in the references and acknowledgement, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.

We hereby certify that this report has not been done by only one individual and all of us have contributed to the report. The length of contribution to the reports by each individual is noted within this certificate.

We also hereby certify that we have read and understand the content of the total report and no further improvement on the reports is needed from any of the individual's contributors to the report.

We therefore, agreed unanimously that this report shall be submitted for marking and this final printed report has been verified by us.

Signature:



Name: Afnan Hakim bin Adinazrin

Matric Number: 2315987

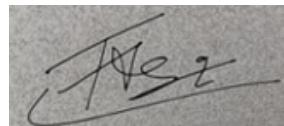
Contribution: Abstract, Introduction, Materials & Equipment,  
Results

Read

Understand

Agree

/
/
/



Signature:

Name: Muhammad Taufiq bin Mukhtar

Matric Number: 2316271

Contribution: Equipment setup, Methodology, Data analysis

Read /

Understand /

Agree /

#### Recommendations



Signature:

Name: Muhammad Danish Farhan bin Amiruddin

Matric Number: 2315423

Contribution: DataCollection, Discussion, Conclusion,  
Appendices, References

Read /

Understand /

Agree /