ELBOW ANGLE BASED LED CONTROL SYSTEM

A PROJECT REPORT

Submitted by

MUHAMMED FARSAN K S

[Register No:2122J0934]

Under the guidance of

Mr. MICHAEL RAJ S, MCA., M.Phil., (Ph.D.,)

[PG Coordinator & Asst.Professor , Department of Computer Applications]

In partial fulfillment for the award of the degree

Of

BACHELOR OF COMPUTER APPLICATIONS

In

DEPARTMENT OF COMPUTER APPLICATIONS

NILGIRI COLLEGE OF ARTS AND SCIENCE

(AUTONOMOUS)

MARCH 2024

NILGIRI COLLEGE OF ARTS AND SCIENCE

(AUTONOMOUS)

DEPARTMENT OF COMPUTER APPLICATIONS

PROJECT WORK

MARCH - 2024

This is to certify that the project entitled

ELBOW ANGLE BASED LED CONTROL SYSTEM

Is a bonafide record of project work done by

MUHAMMED FARSAN K S

[Register No:2122J0934]

Bachelor of Computer Applications during the year 2021-2024

Project Guide | **Head of the Department**

Submitted for the project viva-voce examination held on.............................

Internal Examiner | External Examiner

DECLARATION

I hereby declare that the project work, " **ELBOW ANGLE BASED LED CONTROL SYSTEM**" submitted to Nilgiri College of Arts & Science (Autonomous), in partial fulfillment of the requirements for the award of degree **Bachelor of Computer Applications**, is a record of original project work done by me ,during the period of December 2023 to March 2024 under the guidance of **Mr. MICHAEL RAJ S, MCA., M.Phil., (Ph.D.,) PG Coordinator & Asst.Professor , Department of Computer Applications,** Nilgiri College of Arts and Science(Autonomous), Thaloor.

MUHAMMED FARSAN K S

Signature of the Student

PLACE :

DATE :

ACKNOWLEDGEMENT

I am grateful to Almighty God for giving me the strength, knowledge and understanding to complete this project. His love has been more than sufficient to keep and sustain me.

I hereby express my sincere gratitude to the people, whose cooperation has helped me for the successful completion of my project work. I would like to thank them deep from my heart, for the valuable assistance they rendered for me.

I express my deep sense of gratitude to our honorable Principal In charge **Dr. SENTHIL KUMAR G, M.Sc., M.Phil., (Ph.D.,) Principal, Nilgiri College of Arts and Science (Autonomous),** for the outstanding facilities provided to carry out my project work.

I am extremely grateful display indebted to our **Mr. P MUTHUKUMAR, MCA., M.Phil., B.Ed., (Ph. D.,) Head, Department of Computer Applications,** Nilgiri College of Arts and Science (Autonomous)Thaloor, for the valuable facilities and help provided to me.

I express my deep sense of esteem to my guide **Mr. MICHAEL RAJ S, MCA., M.Phil., (Ph.D.,) PG Coordinator & Asst.Professor , Department of Computer Applications,** Nilgiri College of Arts and Science (Autonomous) Thaloor**,** for all his encouragement, valuable advice and timely instructions at various stages of my project work.

I am thankfully recollecting the helping mentality and kind cooperation rendered by my intimate friends, family and all my dear and near ones for the successful completion of my project work.

MUHAMMED FARSAN K S

LIST OF FIGURES

LIST OF ABBREVIATIONS

- **RGB LED:** Red, Green, Blue Light-Emitting Diode
- **LED:** Light-Emitting Diode
- **Arduino Nano:** A microcontroller board from Arduino used for various electronic projects
- **USB:** Universal Serial Bus (used for power and communication with the Arduino)
- **GND:** Ground (common reference point in a circuit)
- **GPIO:** General-Purpose Input/Output
- **API:** Application Programming Interface
- **CV:** Computer Vision

ABSTRACT

This project presents the design and development of an interactive system that utilizes computer Vision techniques to control the color of LEDs based on the user's elbow angle. The system leverages an Arduino Nano microcontroller as the central processing unit, interfacing with a webcam to capture real-time video frames. Employing computer vision libraries like OpenCV, the system identifies the user's elbow within the image frame. Subsequently, through geometric calculations, the elbow joint angle is determined.

Predefined threshold ranges categorize the calculated angle into "low," "medium," or "high." Based on this categorization, the system activates the corresponding green, blue, or red LED, respectively, providing visual feedback on the detected elbow angle.

TABLE OF CONTENTS

PROBLEM DEFINITION

1. PROBLEM DEFINITION

1.1.1 OVERVIEW

This project endeavors to create a real-time system capable of determining the user's elbow angle and adjusting the color of LEDs accordingly. The system comprises various hardware components, including an Arduino Nano for data processing and LED control, a webcam for capturing elbow video frames, and LEDs with resistors for visual feedback. Additionally, software components involve Arduino IDE for programming and OpenCV or similar libraries for image processing and computer vision tasks. The operational flow involves continuous video frame capture, image preprocessing, elbow detection using color-based segmentation, feature detection, or machine learning models, angle calculation via trigonometry, LED control based on angle thresholds, and calibration options for user-specific settings. Tuning parameters ensure accurate detection and robust performance across different body dimensions and lighting conditions.

1.1.2 PROBLEM STATEMENT

This project aims to revolutionize the monitoring of joint movements by proposing a low-cost, interactive system that utilizes common components and computer vision techniques. Unlike traditional methods reliant on manual observation or expensive physical therapy equipment, this system offers objective and continuous monitoring of the user's elbow angle in real-time. Through immediate visual feedback via LED color change, it provides intuitive and engaging interaction. Moreover, its versatility extends beyond its primary focus, offering potential applications in physical therapy, gaming, and human-computer interaction. By overcoming the limitations of conventional approaches, this project presents a novel and accessible solution for monitoring and interacting with joint movements, thus opening doors for further exploration and development in this field.

INTRODUCTION

2. INTRODUCTION

This project introduces an interactive system that revolutionizes the monitoring and analysis of human joint movements by leveraging computer vision techniques and accessible components. By utilizing an Arduino Nano microcontroller and a webcam, the system accurately captures and analyzes elbow joint angles in real-time. Through the integration of OpenCV and predefined threshold values, the system categorizes angles and provides immediate visual feedback via LED color changes.

This low-cost approach democratizes access to joint angle monitoring, with applications spanning physical therapy, gaming, and human-computer interaction. Its real-time feedback capabilities empower users across various domains, marking a significant advancement in interactive systems and computer vision technology.

2.1 SYSTEM SPECIFICATION

2.1.1 Hardware Requirement:

Processor	:	Corei5 Processor
RAM	:	16GB
ROM	:	512 GB

2.1.2 Software Requirement:

Front End	:	Python , CPP
Operating system	:	windows 11

SYSTEM STUDY

3. SYSTEM STUDY

System study is the first stage of a system development life cycle. This gives a clear picture of what the physical system actually is. The system study is done in two phases. In the first phase, the preliminary survey of the system is done which helps in identifying the scope of the system. The second phase of the system study is a more detailed and in-depth study in which the identification of user’s requirements and the limitations and problems of the present system are studied. After completing the system study, a system proposal is prepared by the user.

3.1 EXISTING SYSTEM

The existing system for elbow angle-based LED control relies on manual input or fixed thresholds to determine LED activation corresponding to different elbow angles. This system lacks adaptability and real-time feedback, as it does not utilize advanced techniques such as computer vision for accurate angle detection. Users must manually set thresholds or adjust controls, which can lead to inaccuracies and inefficiencies in LED activation. Moreover, the system does not provide real-time feedback to users, hindering their ability to make immediate adjustments to their movements. Overall, the existing system's reliance on manual control and fixed thresholds limits its accuracy, flexibility, and usability for users requiring dynamic and precise LED control based on elbow angle.

3.1.1 Drawbacks

- Manual Control
- Lack of Accuracy
- Limited Flexibility
- No Real-time Feedback
- Complexity

3.2 PROPOSED SYSTEM

The proposed system introduces an innovative approach to elbow angle-based LED control by integrating advanced computer vision techniques for real-time angle estimation. Unlike the existing manual control system, the proposed system dynamically activates LEDs in different colors green for low angles, blue for medium angles, and red for high angles based on the detected elbow angle. This adaptive LED control system offers users accurate and immediate feedback on their movements, enhancing usability and effectiveness. Additionally, the system provides calibration options to customize angle detection according to individual users' range of motion, ensuring precise LED activation. With its user-friendly interface and potential for integration with other devices or applications, the proposed system offers a comprehensive solution for users seeking dynamic and intuitive LED control based on elbow angle.

3.2.1 Features

- Computer Vision Integration
- Adaptive LED Control
- Real-time Feedback
- Caliberation Options
- User-Friendly Interface
- Customization
- Integration Potential

LITERATURE SURVEY

4. LITERATURE SURVEY

Literature Survey: Elbow Angle Based LED Control System

This section will explore existing research and developments related to Elbow angle controlled systems, focusing on their application in controlling RGB LEDs.

1. Technological Advances in Computer Vision for Human Motion Analysis:

It covers topics such as pose estimation algorithms, deep learning approaches, and real-time performance optimization, providing insights into the state-of-the-art methods applicable to elbow angle detection in LED control systems by using computer vision techniques.

2. Integration of Wearable Sensors in Interactive Systems:

It discusses sensor fusion techniques, data processing algorithms, and integration challenges, offering guidance on selecting and deploying sensors for accurate elbow angle estimation in LED control systems.

3. User-Centric Design Principles for Interactive LED Systems:

It discusses user preferences, usability testing methodologies, and design guidelines for creatingintuitive and engaging LED control interfaces tailored to the needs of users interacting with elbow angle-based control systems.

4. Applications of Gesture Recognition in Gaming and Entertainment:

It discusses game design concepts, user engagement strategies, and implementation considerations relevant to incorporating elbow angle-based LED control in gaming and entertainment contexts.

5. Practical Considerations for Deploying LED Control Systems in Real-World Settings:

This survey addresses practical considerations for deploying LED control systems in real-world environments, including hardware selection, installation methods, and maintenance requirements.

SYSTEM DESIGN AND DEVELOPMENT

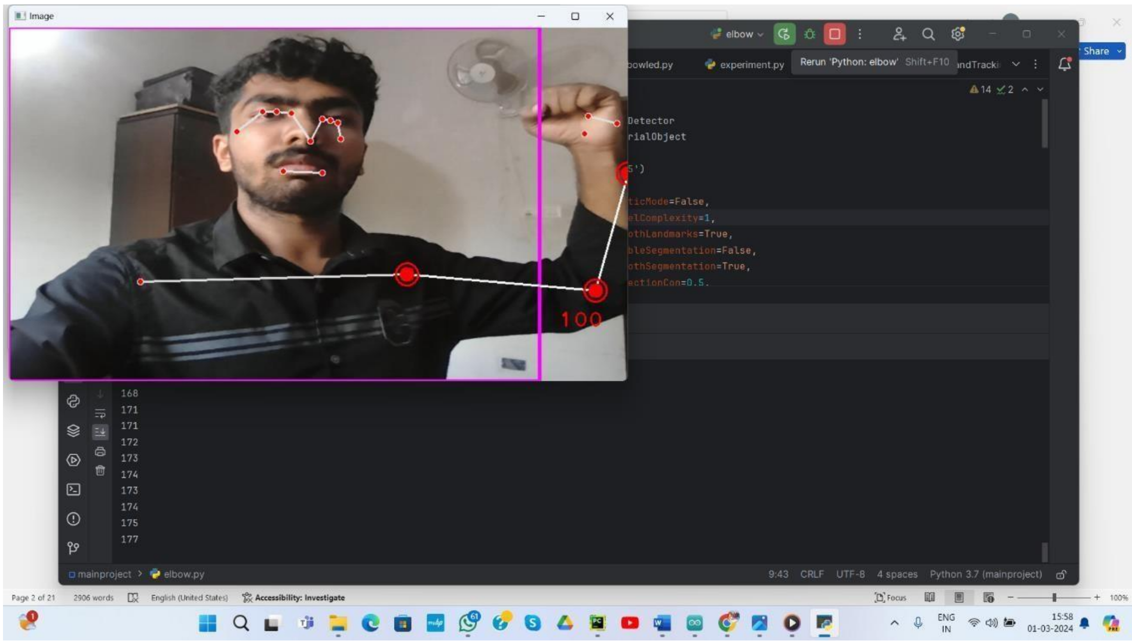
5. SYSTEM DESIGN AND DEVELOPMENT

System design transforms a logical representation of what the system is required to do into the physical specification. The specifications are converted into a physical reality during the development. Design forms a blueprint of the system and adds how the components relate to each other. The design phase proceeds accordingly to an ordinary sequence of steps, beginning with review and assignment of task and ending with package design. Design phase is the life cycle phase in which the detailed design of the system selected in the study phase is accomplished. A smooth transition from the study phase to design is necessary because the design phase continues the activities in the earlier phase. The first step in the design phase is to design the database and then input and output within predefined guidelines.

5.1 INPUT DESIGN

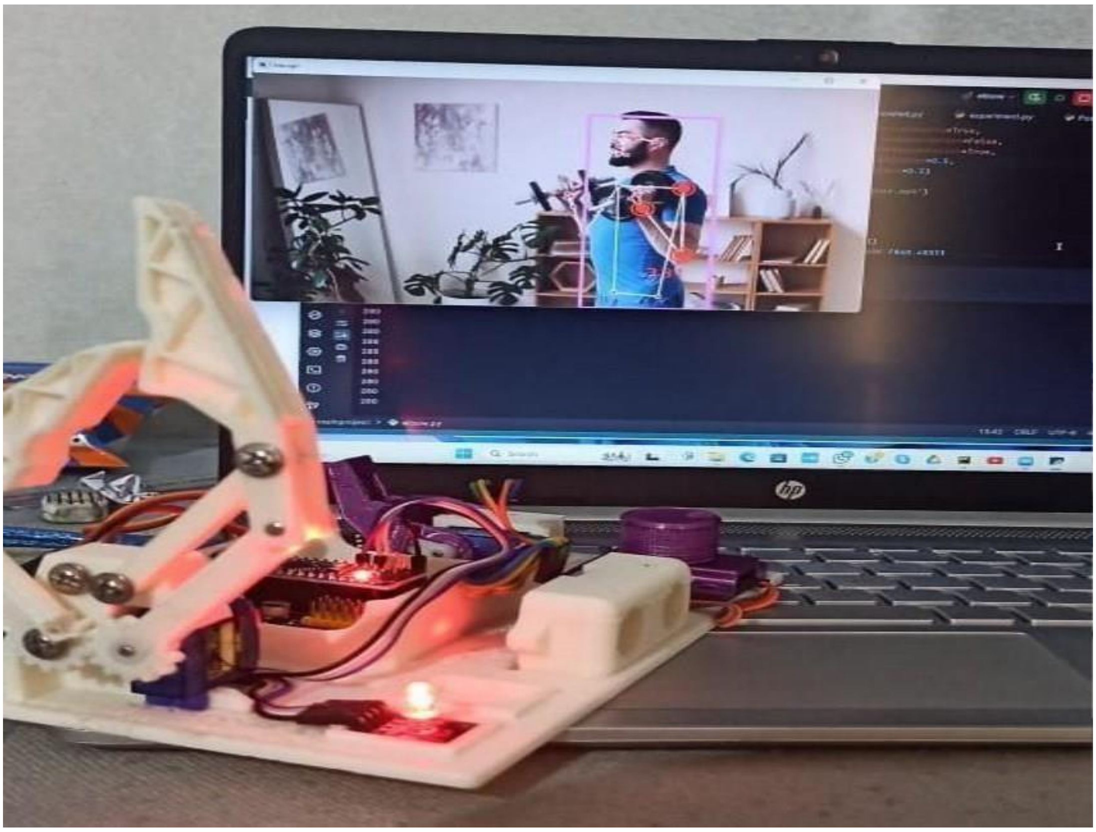
Input design deals with data that should be given as input, how the data should be arranged or code, the dialogue to guide the operating personnel in providing input, methods for preparing input validations and steps to follow when errors occur. Input design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system. It is by creating a user-friendly screen for the data entry to handle large volumes of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data can be performed. It also provides record viewing facilities.

When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user will not be in maize instant. Thus the objective of input design is to create an input layout that is easy to follow. In this project, the input design consists of a login screen, for the admin to get into that for the updation purpose and the users are not allowed to enter into that section of part in the web application.



5.2 OUTPUT DESIGN

A quality output is one, which meets the requirements of the end user and presents the information clearly. The objective of output design is to convey information about the products, current rates of the products mentioned or warnings, trigger an action, confirm an action etc. Efficient, intelligible output design should improve the system's relationships with the user and help in decision making. In output design the emphasis is on displaying the output on a CRT screen in a predefined format. The primary consideration in design of output is the information requirement and objectives of the end users. The major information of the output is to convey the information and so its layout and design need careful consideration. There is an output display screen for showing the compressed/decompressed file or folder details (Original file size, Compressed/Decompressed file size, distinct characters).



5.3 DESCRIPTION OF MODULES

1.Angle Detection Module:

This module is responsible for capturing live video input from the webcam. It utilizes OpenCV algorithms to process the video feed and detect hand gestures in real-time.

2.NumPy:

Provides support for mathematical operations and arrays, which may be useful for processing and manipulating data obtained from the camera.

3.Serial Communication Module:

This module facilitates communication between the computer running the gesture recognition software and the Arduino board controlling the LEDs.

4.Arduino Control Module:

On the Arduino board, this module receives gesture commands via serial communication from the computer.

5.User Interface Module:

This module provides a graphical user interface (GUI) for visualizing gesture recognition feedback and providing user instructions.

6.Pose Detection:

This module provides functionality for detecting human poses in images or video streams. It utilizes advanced algorithms and machine learning models to accurately identify points corresponding to different parts, including elbows and angle values in project.

TESTING AND IMPLEMENTATION

6. TESTING AND IMPLEMENTATION

Testing is a set of activities that can be planned in advance and conducted**.** Systematically, this is aimed at ensuring that the system works accurately and efficiently before live operations commences.

- Testing Is the process of correcting a program with the intent of finding an error.
- A good test case is one that has a high probability of finding a yet undiscovered error.
- A successful test is one that uncovers a yet undiscovered error.

Testing objectives

There are several rules that can serve as testing objectives

- Testing is a process of executing a program with the intent of finding an error.
- A good test case is one that has a high probability of finding an undiscovered error.
- A successful test is one that uncovers an undiscovered error.

Testing is vital to the success of the system. System testing makes a logical assumption that all parts of the system are subject to a variety of tests on-line response, volume, stress, recovery and security and usability tests. A series of tests are performed before the system is ready for user acceptance testing.

Testing Strategies

- Manual Testing
  - Usability Testing
  - Acceptance Testing

Manual Testing

This testing is performed without taking help of automated testing tools. The software tester prepares test cases for different sections and levels of the code, executes the tests and reports the result to the manager. Manual testing is time and resource consuming. The tester needs to confirm whether or not the right test cases are used. Major portion of testing involves manual testing.

Usability Testing

Usability testing refers to evaluating a product or service by testing it with representative users. Typically, during a test, participants will try to complete typical tasks while observers watch, listen and take notes. The goal is to identify any usability problems, collect qualitative and quantitative data and determine the participant's satisfaction with the product.Usability testing lets the design and development teams identify problems before they are coded. The earlier issues are identified and fixed, the less expensive the fixes will be in terms of both staff time and possible impact to the schedule. Thus the proposed system under consideration has been tested by usability testing & found to be working successfully. Also it satisfy the following conditions :

- To know the participants are able to complete specific tasks successfully.
- Identify how long it takes to complete specific tasks.
- Find out how satisfied participants are with your Web site or other product.
- Identify changes required to improve user performance and satisfaction.
- And analyze the performance to see if it meets your usability objectives.

Acceptance Testing

When the software is ready to hand over to the customer it has to go through the last phase of testing where it is tested for user-interaction and response. This is important because even if the software matches all user requirements and if the user does not like the way it appears or works, it may be rejected.

Alpha testing - The team of developers themselves perform alpha testing by using the system as if it is being used in a work environment. They try to find out how users would react to some action in software and how the system should respond to inputs.

Beta testing - After the software is tested internally, it is handed over to the users to use it under their production environment only for testing purpose. This is not yet the delivered product. Developers expect that users at this stage will bring minute problems, which were skipped to attend

Libraries:

- **Open CV:**

  OpenCV (Open Source Computer Vision Library) is a popular open-source library for computer vision and image processing tasks.

- **Cvzone:**

  Cvzone is a Python library built on top of OpenCV, designed to simplify computer vision tasks and streamline the development process.

- **Mediapipe:**

  Mediapipe is a Google-developed library for building machine learning pipelines for various tasks, including hand tracking, pose estimation, and facial recognition.

- **Pyfirmata:**

  Pyfirmata is a Python library for communicating with Arduino microcontroller boards using the Firmata protocol.

- **Raspberry Pi GPIO Library:**

  If Raspberry Pi is used as the controller, libraries such as RPi.GPIO will be essential for interfacing with GPIO pins and controlling hardware peripherals.

- **Pyserial:**

  PySerial is a library for serial communication in Python, which can be used to communicate with microcontrollers or hardware devices such as LEDs for controlling their behavior.

IMPLEMENTATION

Hardware Setup:

Gather necessary hardware components, including an Arduino board, LEDs, a webcam, and any required peripherals.

Software Installation:

Install the Arduino IDE on your computer and set up the Arduino board for programming. Install OpenCV library and any dependencies on your development environment for image processing and gesture recognition.

Elbow Recognition Algorithm:

Develop or utilize pre-existing hand gesture recognition algorithms using OpenCV. Implement algorithms for hand detection, tracking, and angle recognition.

Arduino Programming:

Arduino code to receive gesture commands from the computer via serial communication. Implement functions to control the LEDs based on the received commands.

Computer Vision Integration:

Set up the webcam to capture live video input Process the video feed using OpenCV to detect and recognize Elbow angle in real time.Map detected gestures to corresponding commands and transmit them to the Arduino board for LED control.

Integration and Deployment:

Integrate all components of the system, including hardware and software, into a cohesive unit.

User Training :

Provide user training on how to interact with the system using hand gestures.

CONCLUSION

7. CONCLUSION

The development of the elbow angle-based LED control system signifies a significant stride towards intuitive and interactive human-computer interfaces. Through the amalgamation of computer vision, machine learning, and real-time feedback mechanisms, this project has culminated in a sophisticated system capable of accurately detecting and responding to elbow movements in real-time. By leveraging libraries such as OpenCV for precise pose estimation and PySerial for seamless hardware interfacing, the system demonstrates its robustness and versatility in diverse applications ranging from gaming to rehabilitation. Furthermore, the emphasis on user-centric design principles ensures that the system is not only technologically advanced but also accessible and customizable to meet the unique needs of individual users.

The successful implementation of the elbow angle-based LED control system underscores the potential of interdisciplinary collaboration in pushing the boundaries of human-computer interaction. Beyond its immediate applications, the project lays the groundwork for future advancements in gesture-based interfaces and interactive systems. Through continued research and development, this technology holds promise in domains such as assistive technology, virtual reality, and beyond. Moreover, the project's open-source nature and modular design encourage collaboration and innovation within the broader community, fostering a culture of creativity and exploration in thefield of human-computer interaction. As we look ahead, the elbow angle-based LED control system stands as a testament to the transformative power of technology in enhancing our interaction with digital environments and enriching the human experience.

SCOPE OF FUTURE ENHANCEMENT

8.SCOPE OF FUTURE ENHANCEMENT

The scope of the elbow angle-based LED control system encompasses a wide range of applications across various domains, including gaming, rehabilitation, assistive technology, and interactive art installations. In gaming, the system can enhance user immersion by translating real-world movements into in-game actions, offering a more immersive and engaging experience. In rehabilitation settings, it can serve as a valuable tool for monitoring and guiding physical exercises, providing real-time feedback to users and clinicians. Additionally, the system holds potential in assistive technology applications, enabling individuals with mobility impairments to control digital interfaces and devices using natural gestures. Moreover, in interactive art installations, the system can be employed to create captivating visual displays that respond to human movements, fostering creativity and interaction among participants.

Looking towards future enhancements, there are several avenues for further development and refinement of the system. Firstly, improvements in pose estimation algorithms and machine learning models could enhance the accuracy and robustness of elbow angle detection, enabling more precise control of LEDs. Additionally, the integration of additional sensors, such as inertial measurement units (IMUs) or depth sensors, could provide complementary information to improve the system's performance in diverse environments and under varying lighting conditions. Furthermore, the development of a user-friendly calibration interface and intuitive gesture recognition system could streamline the setup process and enhance the overall user experience. Lastly, exploring opportunities for wireless communication and cloud-based data processing could enable greater flexibility and scalability, allowing the system to be deployed in a wider range of scenarios and applications.

REFERENCES

9.REFERENCES

1. Bradski, G. Opencv. Retrieved from OpenCV: https://opencv.org/

2. Hassan, M. cvzone. Retrieved from https://www.computervision.zone/

3. OpenAI. (2022). Github. Retrieved from https://github.com/openai/gpt-3.5-5.turboOpenAI.

4. OpenAI. Retrieved from https://openai.com

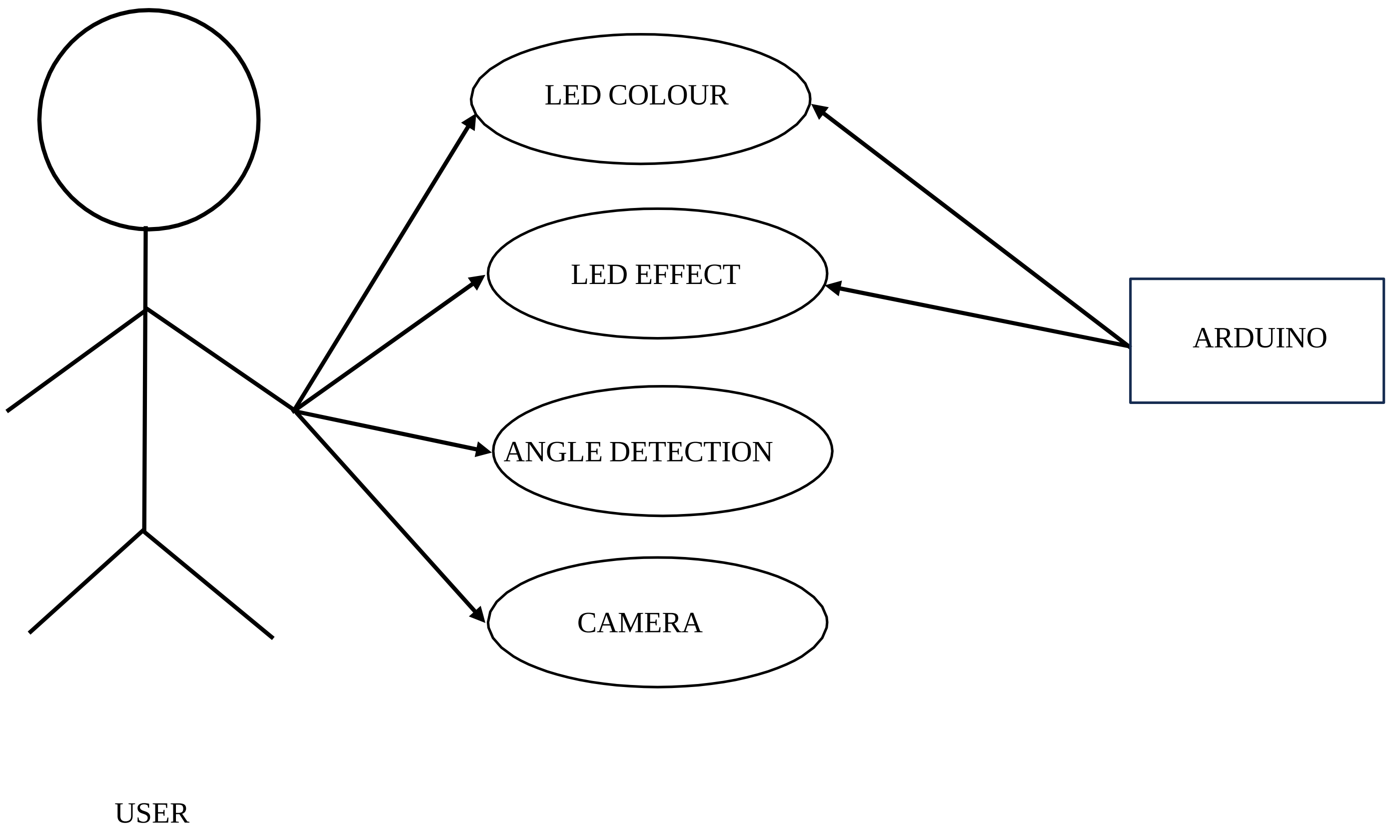
5. Rosebrock, CV ZONE. Retrieved from Computer Vision Zone: https://www.computervision.zone

6. Thompson, R. P. Mediapipe. Retrieved from Mediapipe: https://pypi.org/project/mediapipe/

APPENDICES

10.APPENDICES

A. USE CASE DIAGRAM



B. SAMPLE CODING

Elbowled.py

```
import cv2
import  numpy as np
from PoseModule import PoseDetector
from SerialModule import SerialObject
arduino = SerialObject('COM5')
detector = PoseDetector(staticMode=False,
                        modelComplexity=1,
                        smoothLandmarks=True,
                        enableSegmentation=False,
                        smoothSegmentation=True,
                        detectionCon=0.5,
                        trackCon=0.5)
cap = cv2.VideoCapture('trainer.mp4')
while True:
    success, img = cap.read()
    img = cv2.resize(img, (1280, 720))
    # Human pose
    img = detector.findPose(img)
    lmlist, bboxInfo = detector.findPosition(img, draw=True,
    bboxWithHands=False)
    # Calculate angle
    angle, img = detector.findAngle(lmlist[11]
    [0:2], lmlist[13][0:2],
    lmlist[15][0:2],
    img=img, color=(0, 0, 255), scale=10)
    per = int(np.interp(angle, (40, 175), (100, 280)))
    print(angle)
    # Control LED based on angle
    arduino.sendData([angle])
    cv2.imshow("Image", img)
```

```
    cv2.waitKey(1)
```

Arduino Code

```
#include <zebjus.h>
int redLED = 11;  // Pin connected to the red LED
int greenLED = 12;  // Pin connected to the green LED
int blueLED = 13;  // Pin connected to the blue LED
SerialData serialData(1, 3);  // (num0fValsRec, digitalsPerValRec)
int valsRec[1];
void setup() {
  pinMode(redLED, OUTPUT);
  pinMode(greenLED, OUTPUT);
  pinMode(blueLED, OUTPUT);
  Serial.begin(9600);  // Start serial communication
}
void loop() {
  if (Serial.available() > 0) {
    int angle = Serial.parseInt();  // Read angle value from serial
    Serial.println(angle);
    if (angle >= 320) {
      // High angle, turn on red LED
      digitalWrite(redLED, HIGH);
      digitalWrite(greenLED, LOW);
      digitalWrite(blueLED, LOW);
    } else if (angle >= 275 && angle < 320) {
      // Medium angle, turn on green LED
      digitalWrite(redLED, LOW);
      digitalWrite(greenLED, HIGH);
      digitalWrite(blueLED, LOW);
    } else {
      // Low angle, turn on blue LED
      digitalWrite(redLED, LOW);
```

```
        digitalWrite(greenLED, LOW);
        digitalWrite(blueLED, HIGH);
      }
    }
}
```

Serialmodule.py

```
import serial
import logging
import serial.tools.list_ports
class SerialObject:
    """
    Allow to transmit data to a Serial Device like Arduino.
    Example send $255255000
    """
    def __init__(self, portNo=None, baudRate=9600, digits=1, max_retries=5):
        """
        Initialize the serial object.
        :param portNo: Port Number
        :param baudRate: Baud Rate
        :param digits: Number of digits per value to send
        :param max_retries: Maximum number of retries to connect
        self.portNo = portNo
        self.baudRate = baudRate
        self.digits = digits
        self.max_retries = max_retries

        if self.portNo is None:
            for retry_count in range(1, self.max_retries + 1):
                print(f"Attempt {retry_count} of {self.max_retries} to connect...")
                connected = False
                ports = list(serial.tools.list_ports.comports())
                for p in ports:
                    if "Arduino" in p.description:
```

```
                print(f'{p.description} Connected')
                self.ser = serial.Serial(p.device)
                self.ser.baudrate = baudRate
                connected = True
                break
        if connected:
            break
        else:
            print(f"Attempt {retry_count} failed. Retrying...")
        if not connected:
        logging.warning("Arduino Not Found. Max retries reached. Please enter COM Port Number
instead.")
    else:
      for retry_count in range(1, self.max_retries + 1):
        print(f"Attempt {retry_count} of {self.max_retries} to connect...")
        try:
            self.ser = serial.Serial(self.portNo, self.baudRate)
            print("Serial Device Connected")
            break
        except:
            print(f"Attempt {retry_count} failed. Retrying...")
            if retry_count >= self.max_retries:
                logging.warning("Serial Device Not Connected. Max retries reached.")
```

C.SAMPLE INPUT

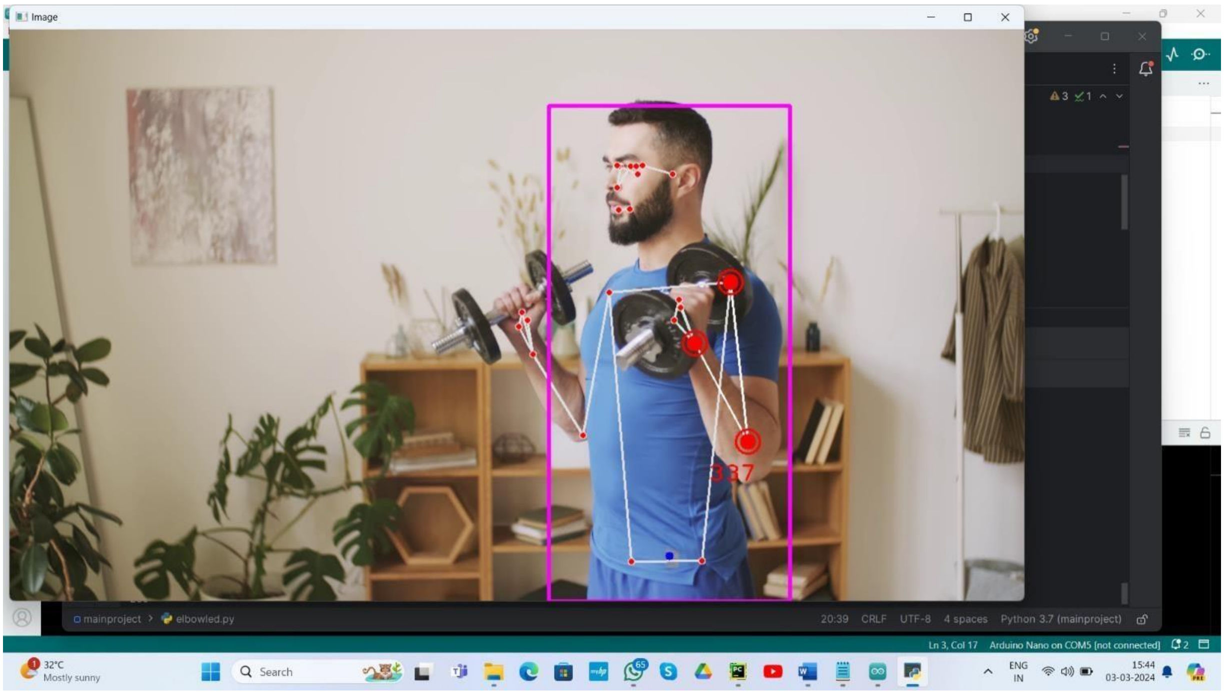


Fig 1: Angle detection using open cv and Mediapipe

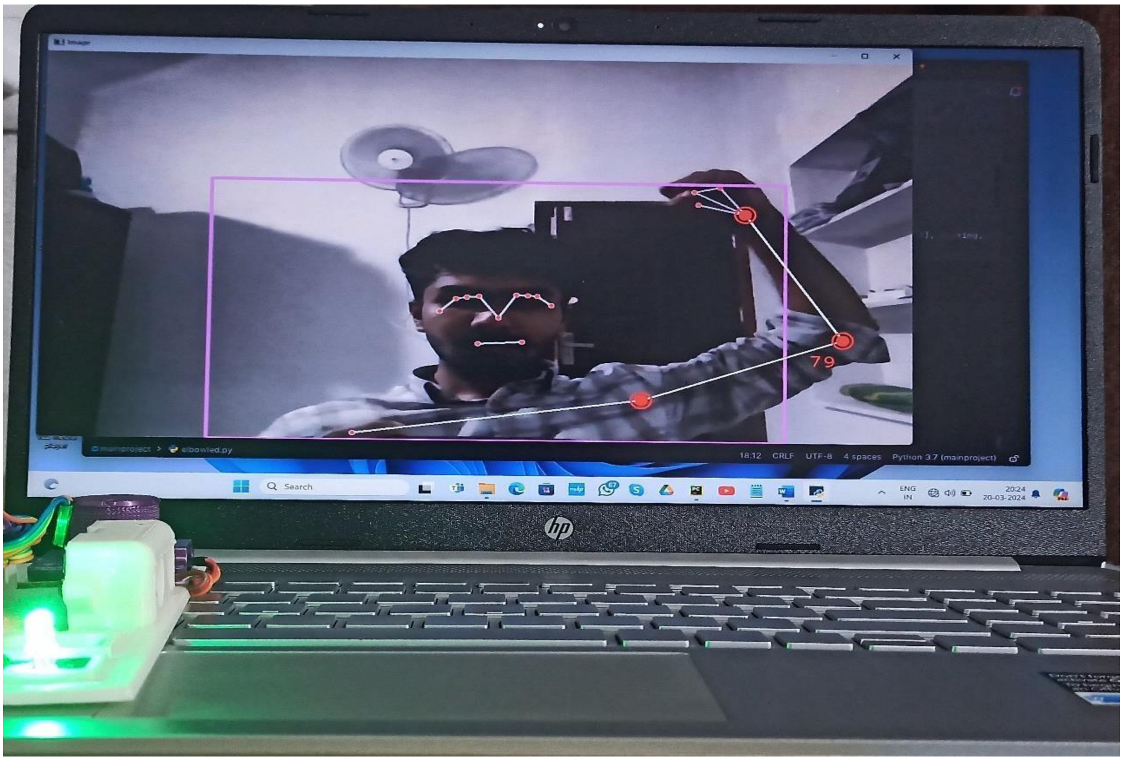


Fig 2: Angle detection with led blinking by using webcam

C. SAMPLE OUTPUT

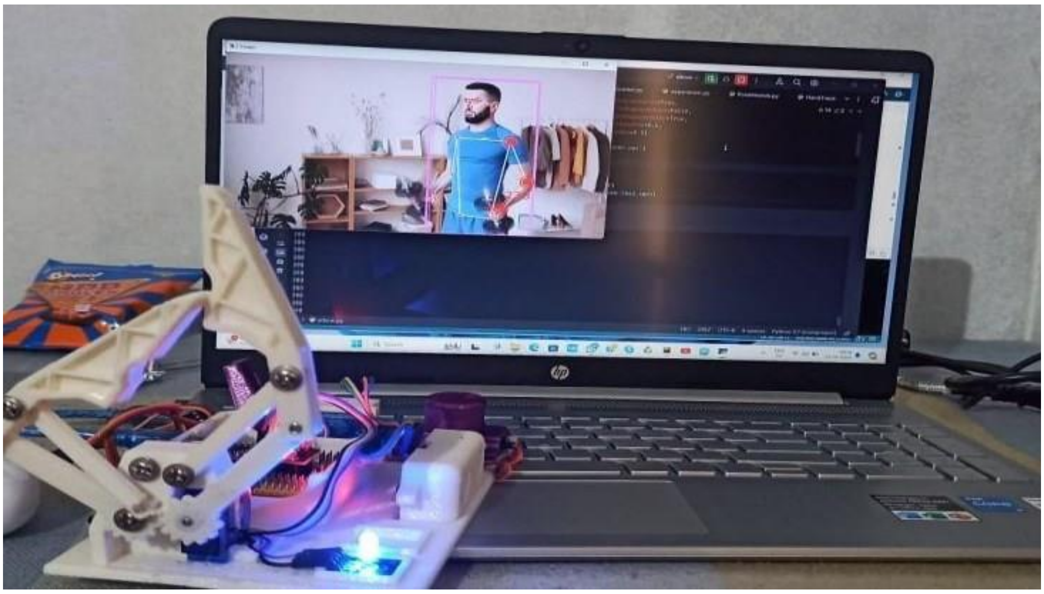


Fig 3: Turning on Blue LED when the angle is medium

Fig 4: Turning on Blue LED when the angle is low

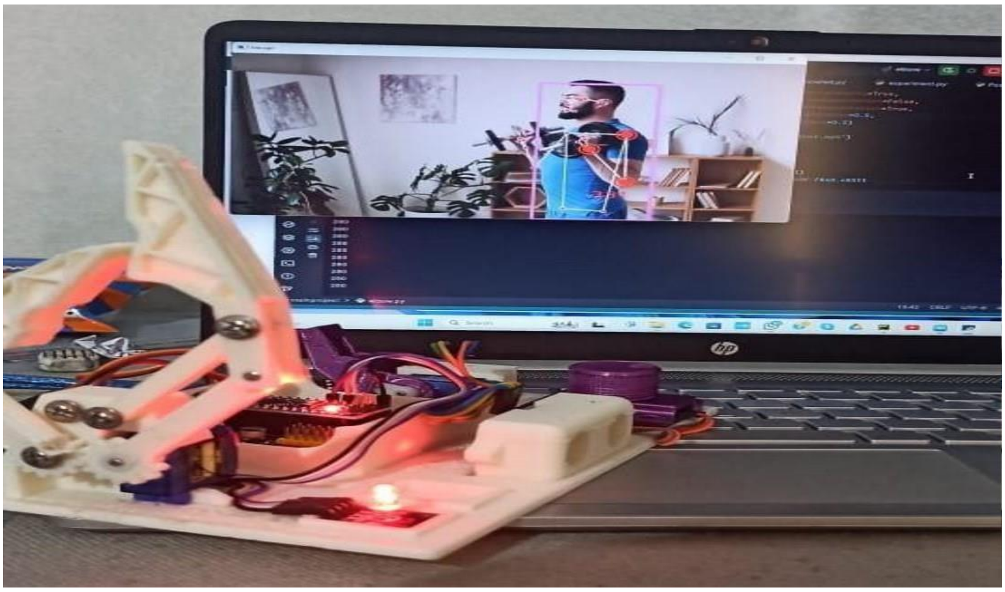


Fig 5: Turning on Blue LED when the angle is high

Fig 6:Arduino Nano with USB Cable