Internet of Things based Smart Switching and Controlling Module with Computer Vision

A Project report is submitted in partial fulfillment of the requirements for the award of Degree of Bachelor of Science in Electrical and Electronic Engineering.

Submitted by

Name: Md. Hasemi Rafsan Jani Shohan

ID: 191-33-849

Supervised by

Ms. Tanjum Rahi Akanto

Lecturer

Department of Electrical and Electronic Engineering



Department of Electrical and Electronic Engineering

Faculty of Engineering

DAFFODIL INTERNATIONAL UNIVERSITY

February, 2023

Declaration

I hereby declare that this project “Internet of Things based Smart Switching and Controlling Module with Computer Vision.” represents my own work which has been done in the laboratories of the Department of Electrical and Electronic Engineering under the Faculty of Engineering of Daffodil International University in partial fulfillment of the requirements for the degree of Bachelor of Science in Electrical and Electronic Engineering, and has not been previously included in a thesis or dissertation submitted to this or any other institution for a degree, diploma or other qualifications. I have attempted to identify all the risks related to this research that may arise in conducting this research, obtained the relevant ethical and/or safety approval (where applicable), and acknowledged my obligations and the rights of the participants.

Signature of the candidates

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Name: …………………..

ID: ……………………..

Approval

The project entitled “Internet of Things and Manual Remote Control based Smart Switching and Controlling Module with Computer Vision.” submitted by Md. Hasemi Rafsan Jani Shohan (ID 191-33-849) has been done under my supervision and accepted as satisfactory in partial fulfillment of the requirements for the degree of Bachelor of Science in Electrical and Electronic Engineering in February, 2023.

Signed

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Ms. Tanjum Rahi Akanto

Lecturer

Department of Electrical and Electronic Engineering

Faculty of Engineering

Daffodil International University

**ACKNOWLEDGEMENT**

First express my heartiest thanks and gratefulness to the Almighty ALLAH for his divine blessing to make us possible to complete this project successfully.

It is my great opportunity to convey the deepest and veneration to my honorable thesis supervisor, Ms. Tanjum Rahi Akanto, Lecturer, Department of Electrical and Electronic Engineering, Daffodil International University (DIU),

For engaging me in such an important research. She was always there to share his absolute expertise and valuable time.

It was his constant guidance, helpful suggestions, constructive criticism and endless patience throughout development of this project .

We are also grateful to different online resources from which we have got much information.

For his kind help to finish This project and also to other faculty members and the staff of EEE department of Daffodil International University.

I want to be more dedicated to various digital platforms such as thattube google, stack flow, Arduino cc, adafruit io, IFTTT that have greatly helped me in This project through information sharing.

Table of Contents

|  |  |
| --- | --- |
| **List of Figures** | **vii** |
| **List of Tables** | **ix** |
| **List of Abbreviations** | **x** |
| **List of Symbols** | **xi** |
| **Acknowledgment** |
| **Abstract** |

[**CHAPTER 1: INTRODUCTION 12**](#_Toc118501734)

[1.1 Background / Motivation / Introduction](#_Toc118501736) 13

[1.2 Problem Statement and /or Proposed solution(s)](#_Toc118501737) 14

[1.3 Aims/ Objectives](#_Toc118501738) 15

[1.4 Brief Methodology / Technologies/ Procedures](#_Toc118501739) 15

[1.5 Implementation Schedule / Gantt Chart](#_Toc118501740) 16

[1.6 Organization / Structure of the Report](#_Toc118501741) 17

[**CHAPTER 2:**](#_Toc118501742)[**LITERATURE REVIEW 18**](#_Toc118501743)

[2.1 Introduction 19](#_Toc118501744)

[2.2 Related Research/ Works 20](#_Toc118501745)

[2.3 Compare and Contrast 22](#_Toc118501746)

[2.4 Summary 23](#_Toc118501747)

[**CHAPTER 3**](#_Toc118501748)**:** [**MATERIALS AND METHODS [OR PROJECT/SYSTEM DESIGN] [OR DESIGN PROCEDURE] 24**](#_Toc118501749)

[3.1 Introduction 25](#_Toc118501750)

[3.2 Methods and Materials or System Design and Components 29](#_Toc118501751)

[3.3 Design Specifications. Standards and Constraints 33](#_Toc118501752)

[3.4 System Analysis or Design Analysis 36](#_Toc118501753)

[3.5 Simulation/Experimental Setup [or Implementation] 39](#_Toc118501754)

[3.6 Summary 39](#_Toc118501755)

[**CHAPTER 4**](#_Toc118501756)**:** [**RESULTS AND DISCUSSIONS 40**](#_Toc118501757)

[4.1 Results/ Observations/ 41-44](#_Toc118501758)

[4.2 Discussions 45](#_Toc118501759)

[**CHAPTER 5**](#_Toc118501760)**:** [**PROJECT MANAGEMENT 46**](#_Toc118501761)

[5.1 Task, Schedule and Milestones 46](#_Toc118501762)

[5.2 Resources and Cost Management 47](#_Toc118501763)

[5.3 Lesson Learned 47](#_Toc118501764)

[**CHAPTER 6**](#_Toc118501765)**:** [**IMPACT ASSESSMENT OF THE PROJECT 48**](#_Toc118501766)

[6.1 Economical, Societal and Global Impact 48](#_Toc118501767)

[6.2 Environmental and Ethical Issues 48](#_Toc118501768)

[6.3 Utilization of Existing Standards or Codes 48](#_Toc118501769)

[6.4 Other Concerns 49](#_Toc118501770)

[**CHAPTER 7**](#_Toc118501771)**:**[**CONCLUSIONS AND RECOMMENDATIONS 51**](#_Toc118501772)

[7.1 Conclusions 51](#_Toc118501773)

[7.2 New Skills and Experiences Learned 51](#_Toc118501774)

[7.3 Future Recommendations 51](#_Toc118501775)

[**REFERENCES 52**](#_Toc118501776)

[**APPENDIX A**](#_Toc118501777)**:** [**TURNITIN REPORT**](#_Toc118501778)

[**APPENDIX B:**](#_Toc118501779)[**COMPLEX ENGINEERING PROBLEM SOLVING AND ENGINEERING ACTIVITIES**](#_Toc118501780)

[**APPENDIX C:**](#_Toc118501781)[**PROGRAM CODE 55-64**](#_Toc118501782)

[**APPENDIX D:**](#_Toc118501783)[**DATASHEET OF COMPONENTS 66**](#_Toc118501784)

List of Figures

|  |  |  |
| --- | --- | --- |
| Fig 1.1 | Logic Control Block Diagram of this project | 12 |
| Fig 1.2 | Control Block of this project | 13 |
| Fig 2.1 | Control Steps Block[27] | 18 |
| Fig 2.2 | Switching Control Block | 19 |
| Fig 2.3 | Voltage Dimmer Control Block | 19 |
| Fig 2.4 | Gas Sensor Value Showing Block | 20 |
| Fig 2.5 | Temperature and Humidity Showing Block | 20 |
| Fig 2.6 | Manual remote Control Based Load Switching | 21 |
| Fig 2.7 | Computer Vision Object Movement Showing frame | 21 |
| Fig 2.8 | Computer Vision Object Movement Showing frame2 | 22 |
| Fig 2.9 | Self Developed Computer Vision Software Object Movement initial frame | 22 |
| Fig 2.11 | After clicking “open camera” button Computer Vision Movement frame. | 24 |
| Fig 2.12 | After clicking “capture” button saved captured image . | 24 |
| Fig 2.13 | Gesture Control phase Voltage Triggering point view. | 25 |
| Fig 2.14 | Gesture Control High Level gap between index and thumb finger . | 26 |
| Fig 2.15 | Gesture Control Medium Level gap between index and thumb finger. | 27 |
| Fig 2.16 | Gesture Control Low Level gap between index and thumb finger | 27 |
| Fig 2.17 | Gesture Control Zero Level gap between index and thumb finger . | 27 |
| Fig 2.18 | After Gesture Control Zero Level to High Level Voltage amplitude and  brightness fluctuate Result. | 28 |
| Fig3.1-2 | IOT and Manual Control structure diagram of this project, Component visual Circuit diagram of this project | 29-31 |
| Fig 3.3 | Node MCU esp8266 | 25 |
| Fig 3.4 | IR Receiver | 26 |
| Fig 3.5 | IR transmitter Star universal brand remote. | 26 |
| Fig 3.6 | Arduino uno r3 | 27 |
| Fig 3.7 | Bulb as an Load | 27 |
| Fig 3.8 | DTH11 Sensor | 28 |
| Fig 3.9 | The MQ6 Gas Sensor | 28 |
| Fig 3.10 | 5v four channel Relay Module | 29 |
| Fig 3.11 | Quard OR gate IC 7432 | 29 |
| Fig 3.12 | Scametric arrangement for Node mcu esp8266 | 30 |
| Fig 3.13 | Scametric arrangement of arduino uno for manual control | 31 |
| Fig 3.14 | Block diagram of Computer vision processing. | 32 |
| Fig 3.15 | Developing Block step For Computer vision Software | 32 |
| Fig 3.16 | IOT platform Creating block step on adfruit server . | 41 |
| Fig 3.17 | IOT platform Creating step 1 | 42 |
| Fig 3.18 | IOT platform Creating step 2 | 42 |
| Fig 3.19 | IOT platform Creating step 3 | 43 |
| Fig 3.20 | IOT platform Creating step 3 | 43 |
| Fig 3.21 | IOT platform Creating step 4 | 44 |
| Fig 3.22 | IOT platform Creating step 5 | 44 |
| Fig 3.23 | IOT platform Creating step 6 | 45 |
| Fig 3.24 | Hands Landmarks point | 47 |
| Fig 3.25 | 25 Hands Landmarks point Specification [35] | 47 |
| Fig 3.26 | Gesture Control Steps 1 | 48 |
| Fig 3.27 | Gesture Control Steps 2 | 48 |
| Fig 3.28 | Block steps of Gesture Control Development | 48 |
| Fig 4.0 | Flow Diagram | 50 |
| Fig 4.1 | IFTTT Service and Appleates1 | 50 |
| Fig 4.2 | IFTTT Service and Appleates2 | 51 |
| Fig 4.3 | IFTTT Service and Appleates3 | 52 |
| Fig 4.4 | Self-developed apps for use in projects | 53 |
| Fig 4.5 | Computer vision or video visualization protocols | 53 |
| Fig 4.6 | IP link http://192.168.0.103:8080/ based computer vision or video visualization through Internet. | 54 |
| Fig 4.7 | IP link http://192.168.0.103:8080/ and contours area selecting based computer vision or video visualization software through Internet. | 54 |
| Fig 4.8 | Index and thumb Finger Gesture Movement based phase angle control part . | 54 |
| Fig A1 | Code for IOT Control | 69 |
| Fig A2 | Code for Manual Remote Control | 71 |
| Fig A3 | Code For Computer vision[22] [9] | 72 |
| Fig A4-5 | Code For Computer vision Software Code For Gesture Control Voltage Dimmer | 77-78 |

Fig A6 Microcontroller Code for Voltage Dimmer 81

List of Tables

|  |  |  |
| --- | --- | --- |
| *Table No* | *Fig. Name* | *Page No.* |
| Table 1.1 | Figure Table | vii |
| Table 2.1 | Project Cost Management | 33 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

List of Abbreviations

|  |  |
| --- | --- |
| **IFTTT** | **IF This Then That** |
| **IOT** | **Internet of Things** |
| **CV2** | **Computer Vision** |
| **IR** | **Infrared Receiver** |
| **MQ6** | **LPG Gas Sensor** |
| **GPIO** | **General Input output Purpose** |
|  |  |

List of Symbols

|  |  |
| --- | --- |
| ***Symbol*** | ***Name of the symbol*** |
| **D1** | **GPIO Pin 1** |
| **D2** | **GPIO Pin 2** |
| **D3** | **GPIO Pin 3** |
| **D5** | **GPIO Pin 4** |
| **A0** | **Analog Input** |
|  |  |
|  |  |

Abstract

The fourth industrial revolution is rapidly approaching. This is my ongoing endeavor in that vein. Several of the elements that are crucial to the Fourth Industrial Revolution are included in This project . IOT-based Smart Switching and Controlling Module with Computer Vision is the focus of This project .

This changes the analog switching mechanism and transforms it into IOT. It makes switching possible from any location in the world. Additionally, a voltage dimmer, a temperature sensor, and a gas sensor were attached. With its assistance, anyone in the globe can find out the temperature, humidity, and amount of methane gas existing in a certain area. If this is implemented correctly, I believe it will make my sentiment duties much simpler.

In the present world's updated electrical engineering technology, I'm hoping This project will function extremely well. In order to make it simple to complete the based duties later, I have once again planned to save all of the project's data on the server at the same time.

I'm hoping that my suggestions from this project may help with the creation of an updated electrical project soon. Above all, I want to thank my managers and all of my coworkers for their cooperation and hard work, which made it easier for me to finish This project .

Computer vision, which permits remote visual inspection of controlled areas, and Gesture Control Phase angle Converter are crucial parts of this project.

For this project, I created computer vision software. Based on an IP link, it allows for video visualization. It makes it possible to detect the movement of things at a given place around the globe. This, in my opinion, will considerably enhance robotic vision.

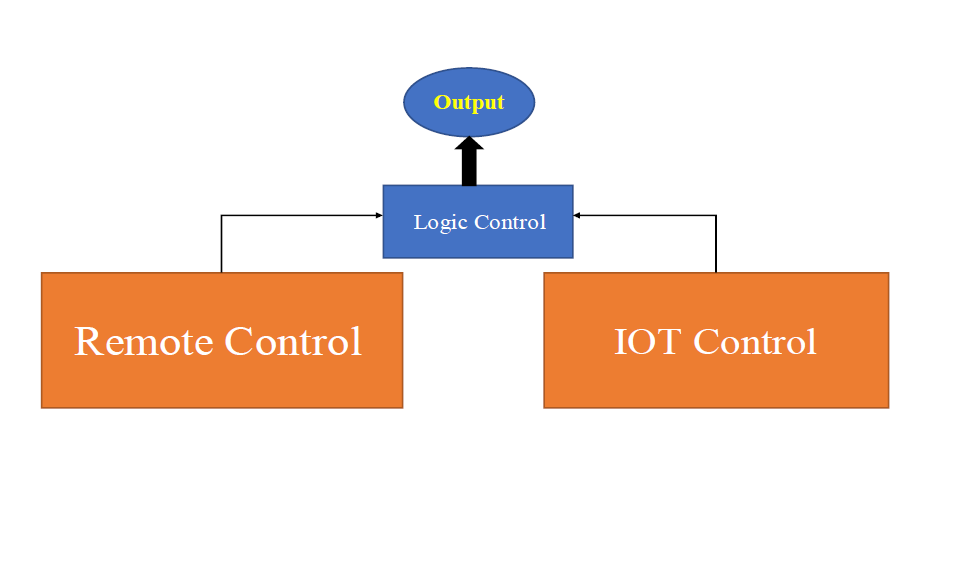
The default settings for Control Area and IP Link. Plans for the project's image capture

It is essentially feasible to visualize the controlled location with the aid of this technique, or to see it in the form of video or CCTV footage and Phase angle Control by Gesture.

The project's whole controlling protocols have been developed in an effort to create an Android IOT control app.

Chapter 1

* 1. Introduction



**Fig 1.1 Logic Controll Block Diagram of this project**

Is the world heading towards the fourth industrial revolution. This is This project in continuation of that. I have include in This project several of the factors that are playing an important role in the Fourth Industrial Revolution. This project is IOT based Smart Switching and controlling Module

By this the analog switching method is changed and converted to IOT. It enables switching from anywhere in the world. And also connected voltage divorce key temperature sensor and gas sensor. With the help, it is possible to determine the temperature and humidity and the amount of methane gas present in a specific place from anywhere in the world. I think if this is implemented properly, my work or work process steps tasks will become much easier.

I hope that This project will work very well in the updated technologies of electrical engineering in the modern world. I have again arranged to save all the data of the project on the server on the same date so that it is convenient to do the based tasks later.

And I have also set up a mail system to send warning messages in different situations. The biggest advantage of This project is that it can be controlled with a system without changing the code to control it.

An important component of this project is computer vision, which enables visual inspection of controlled locations from remote locations and Phase voltage (angle) control by Finger Gesture Movement.

I have developed a computer vision software for this project. It enables video visualization based on IP link. It enables movement detection of objects at a specific location over the world .

I think robotic vision can be greatly improved with this.

It has the default set to IP Link and Control Area. Image Capture arrangements are in project

I hope that my ideas from this project will contribute to the development of updated electrical project in the near future. Above all I would like to thank my supervisors and all my associates for their efforts and their cooperation which facilitated the completion of This project .

* 1. **Problem Statement and /or Proposed solution(s)**

**Computer Video Visualization and Gesture Control**

**World wide Switching and Control ability**

The increasing use of Internet of Things (IoT) devices in homes and businesses has led to a need for a more efficient and secure way to manage these devices. Currently, users must individually access and control each device through separate apps or web interfaces, which can be time-consuming and confusing.

The proposed solution is to develop a central management platform for IoT devices. This platform will allow users to easily access and control all of their connected devices through a single interface. The platform will also incorporate advanced security features to protect against hacking and unauthorized access. Additionally, the platform will include analytics and data visualization tools to help users better understand their usage patterns and optimize their device settings.

**1.3Aims/ Objectives**

To develop a central management platform for Internet of Things (IoT) devices that is user-friendly and efficient.

To improve the security of IoT devices by incorporating advanced security features into the management platform.

To provide data analytics and visualization tools to help users understand and optimize their device usage.

Objectives:

To research and analyze current IoT device management solutions and identify areas for improvement.

To design and develop a user-friendly interface for the central management platform.

To integrate advanced security features, such as encryption and authentication, into the platform.

To test the developed platform with a group of user to gather feedback and make necessary improvements.

To deploy the platform to a small group of early adopters to gather additional feedback and make any final adjustments before a full launch.

To develop data analytics and visualization tools and integrate them into the platform.

To measure the success of the platform through key performance indicators such as user satisfaction, security, and ease of use.

Project aims , objectives and uses Expected Outcome

▪ Home Automation.

▪ Industrial Machine Controlling.

▪ Weather Monitoring.

▪ Robot’s Arm and Robot Control Car Controlling.

▪ Smart Irrigation System Control.

▪ Using Lab use for IOT Smart Trainee Module Board.

▪ Power System And Grid Network Controlling . .

▪ Medical and ICU Instruments Controlling

* 1. **Brief Methodology / Technologies/ Procedures**
  2. **Working Principle**

Display

Adafruit IOT Server

LOAD

IFTTT Server

Sensor And Computer vision section

**Fig 1.2 Control Block of this project**

The project is mainly based on Internet of Things. I have use all the currently updated technologies of IOT.

Five lamps are kept here as load. Four lamps are connected to the switching module and one lamp is connected to the voltage plane.

Used dth11 sensor as temperature sensor and methane gas sensor as gas sensor.

And used RBD Voltage Dimmer module as voltage dimmer.

Used adafruit.io website server as virtual control platform.

I used a web server called IFTTT for controlling.

And used four relay modules.

Used tv remote control system for manual control.

I have arranged to control the project through remote control while being very close to the project.

An important component of this project is computer vision, which enables visual inspection of controlled locations from remote locations.

With the help of this jute, it is basically possible to show the visualization of the controlled place from a very remote place or watch it in the form of video or CCTV footage.

If desired, it can be used for objectification or human detection related or various types of controlling related to computer subjects.

For the time being, I set my mobile phone and used it as a webcam to facilitate the project showing and stream all the data of the visualization through the internet through an IP link based set.

It uses Python as the programming language and Computer Vision as the library file and media from Computer Vision.

In this project it will basically detect the motion of any type of object and can detect the movement.

Along with the whole thing is to see from a far away place.

I used it as an app called IP Webcam and IPL Invest as a browser.

Where there is facility of recording and facility of viewing through internet.

* 1. **Implementation Schedule**

**Phase 5**

**Maintenance**

**Phase 3**

**Testing**

**Phase 2**

**Design**

**Phase 4**

**Visualization**

**Phase 1**

**Research**

**Phase 1 : Research and Analysis (Week 1-4)**

Research current IoT and computer vision solutions and identify areas for improvement

Analyze user needs and requirements

Develop project scope and objectives

**Phase 2: Design and Development (Week 5-12)**

Design user interface for the central management platform

Develop platform architecture and integrate security features

Develop computer vision algorithms and integrate with the platform

Test platform with a small group of users for feedback

**Phase 3: Deployment and Testing (Week 13-16)**

Deploy platform to a small group of early adopters for additional feedback

Make final adjustments based on user feedback

Prepare for full launch

Phase 4: Data Analytics and Visualization (Week 17-20)

Develop data analytics and visualization tools

Integrate tools into the platform

Test and optimize analytics and visualization features

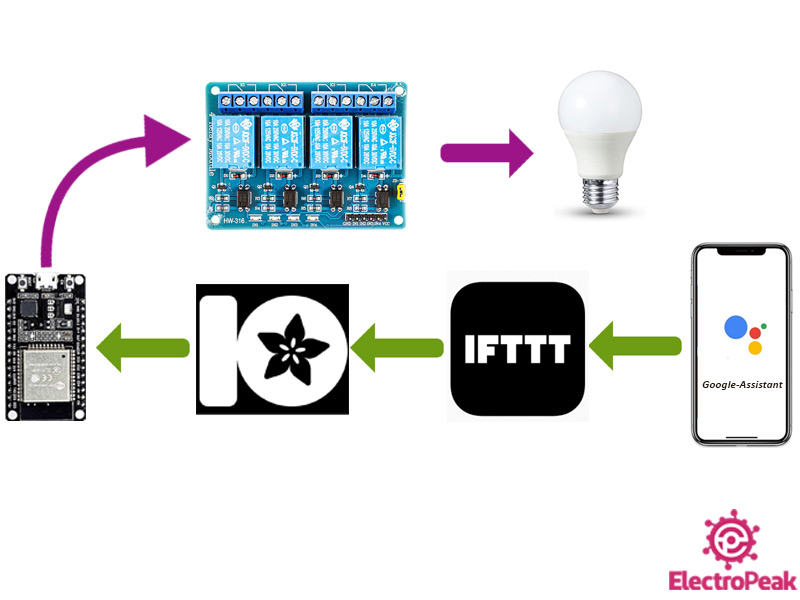
**Phase 5: Launch and Maintenance (Week 21-24)**

Launch the platform to the public

Monitor and maintain the platform

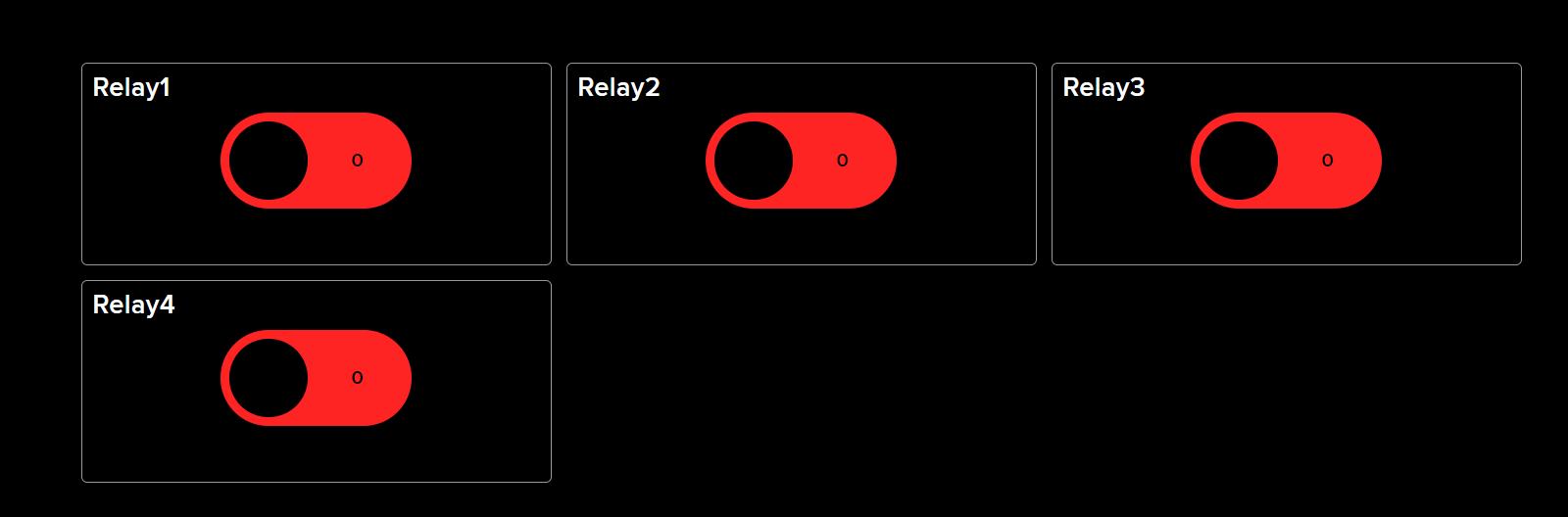
Continuously gather user feedback and make improvements as necessary

2.1 Introduction



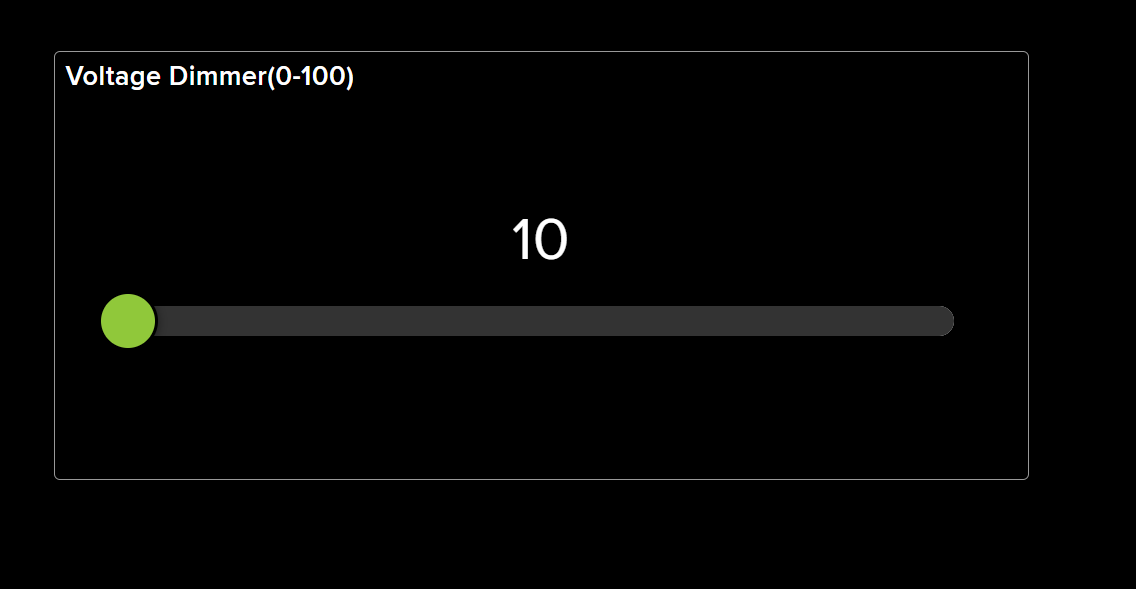
**Fig 2.1 Control Steps Block[27]**

In This Chapter We Will Know How We Will Use Each Of The Control Block To Perform The Various Switching And Controlling Functions Of This project .



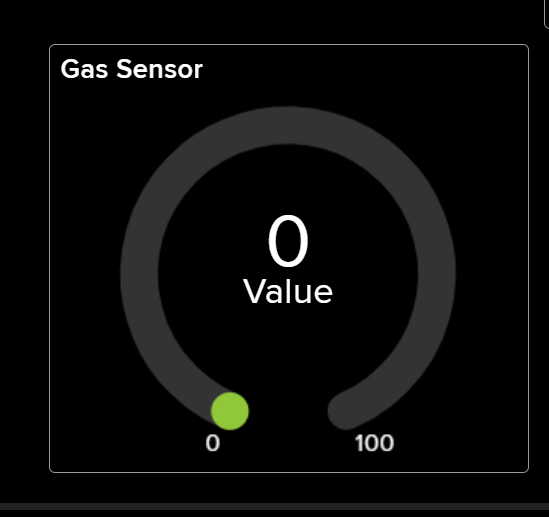
**Fig 2.2 Switching Control Block**

This Figure shows the switching block diagram. Which is expressed as relation Relay 1 , Relay 2, Relay 3, Relay 4. IOT based switching tasks can be done by virtually pressing this buttons.



**Fig 2.3 Voltage Dimmer Control Block**

And the voltage dimmer block diagram is shown in This Figure. IOT based voltage steamer or voltage by changing it. It has ranges set from 0 to 100 to show a voltage variation in the range of 0 to 100.



**Fig 2.4 Gas Sensor Value Showing Block**

This Figure shows the amount of gas sensors that can be used to monitor the amount of gas at a controlled location from a remote location.

With the help of which is the amount of methane gas or LPG gas in a particular place.



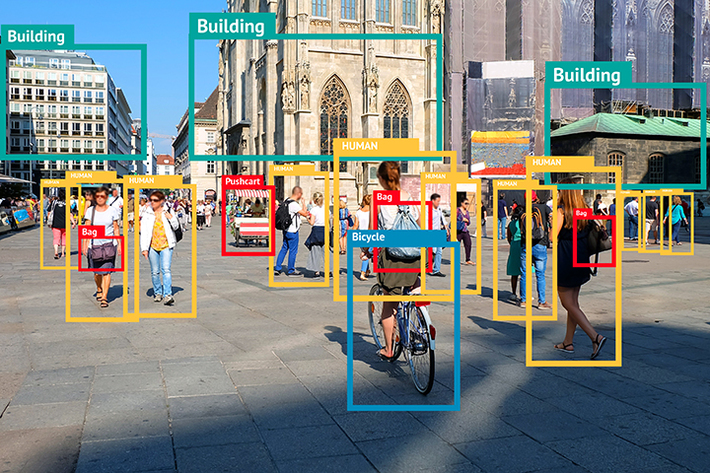
**Fig 2.5 Temperature and Humidity Showing Block**

The values ​​of temperature and humidity sensor are shown with This Figure. Temperature of Humidity values ​​can be seen with its help.

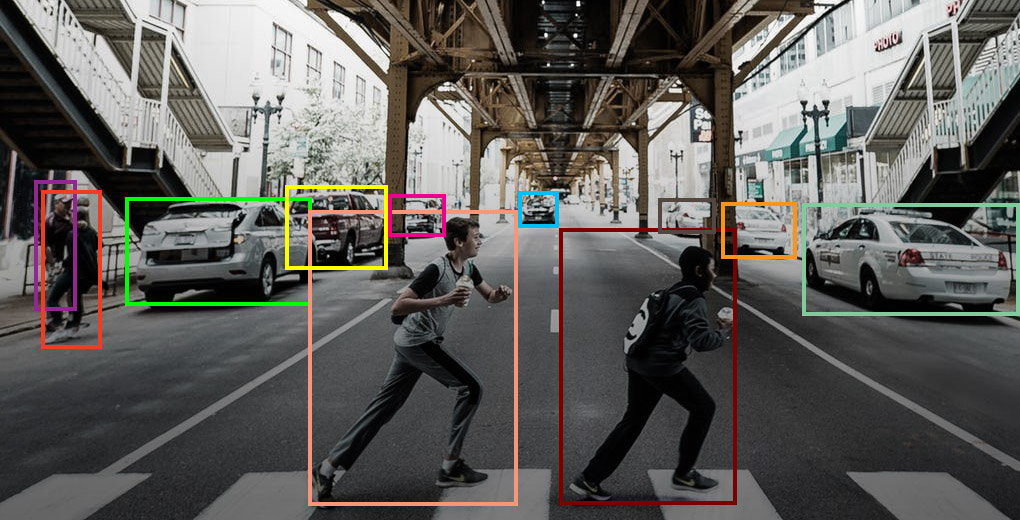


**Fig 2.6 Manual remote Control Based Load Switching**

Figure three shows the use of remote control system. This will act as manual control for this project. This allows project to be controlled while in close proximity to the project.



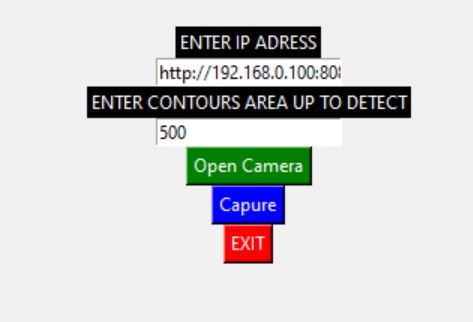
**Fig 2.7 Computer Vision Object Movement Showing frame**



**Fig 2.8 Computer Vision Object Movement Showing frame2**

Computer vision is shown at in this figure of this chapter. With the help of computer vision, it is possible to show the video visualization view of my controlled space from a remote location. It can detect the movement of an object, the presence of a human face, and the presence of a human hand.

This is best for viewing convenience. [8]



**Fig 2.9 Self Developed Computer Vision Sofware Object Movement initial frame**

IoT (Internet of Things) and Computer Vision are two distinct technologies that have different applications and uses.

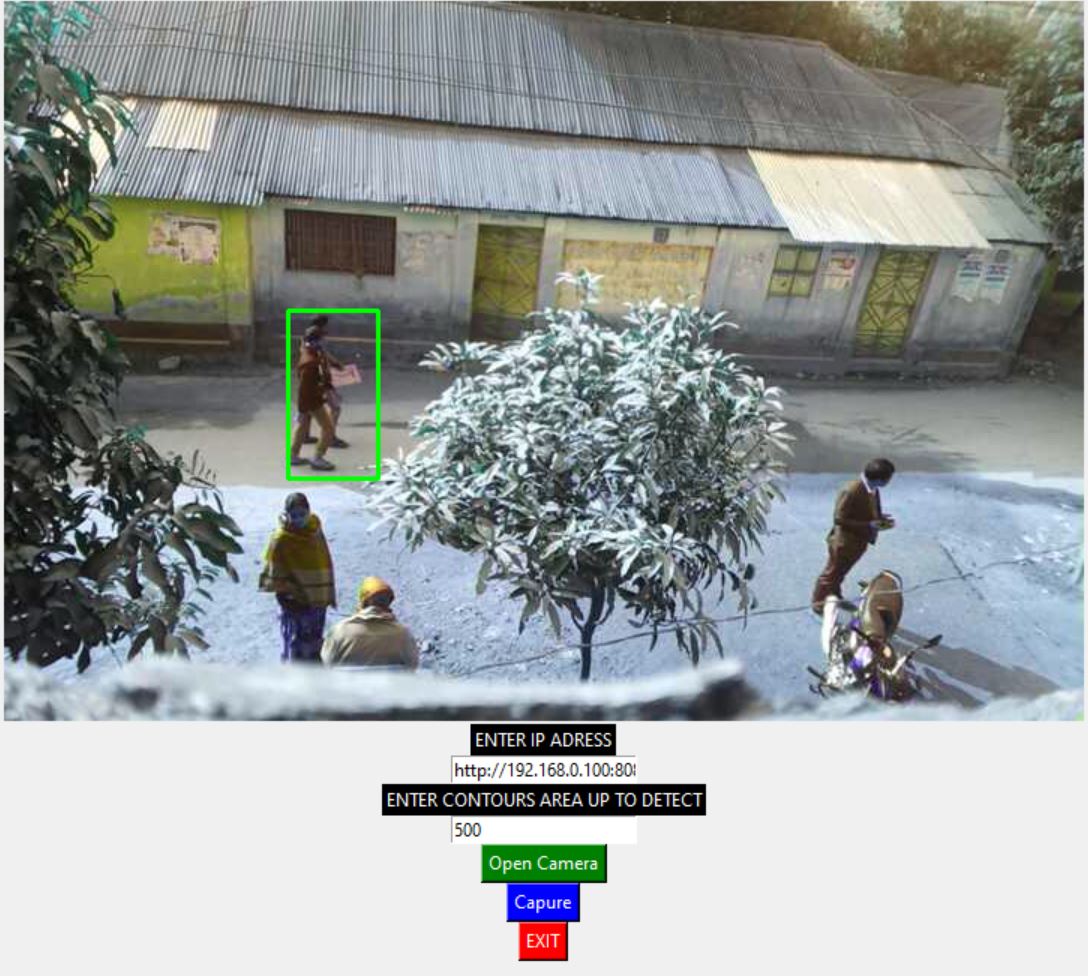
IoT refers to the interconnectedness of physical devices and objects that are connected to the internet, allowing them to communicate and share data with each other. This project is used in a wide range of applications, including home automation, industrial automation, and smart cities.

Computer vision, on the other hand, is a field of artificial intelligence that focuses on the development of algorithms and models that enable machines to interpret and understand visual information. This project is used in a variety of applications, such as image and video recognition, object detection, and facial recognition.

One key difference between the two technologies is the type of data they process. IoT systems typically process data from sensors and other types of input devices, while computer vision systems process visual data, such as images and videos. Additionally, IoT systems are often used for monitoring and control, whereas computer vision systems are used for perception and analysis.

Another difference is the scale of deployment, IoT systems are typically deployed on a large scale, for example, in smart cities and industrial automation, whereas computer vision systems are used in smaller scale applications such as security systems and self-driving cars.

In summary, IoT and computer vision are two distinct technologies that have different applications and uses, IoT systems are used for monitoring and control and are typically deployed on a large scale, whereas computer vision systems are used for perception and analysis and are used in smaller scale applications[4]

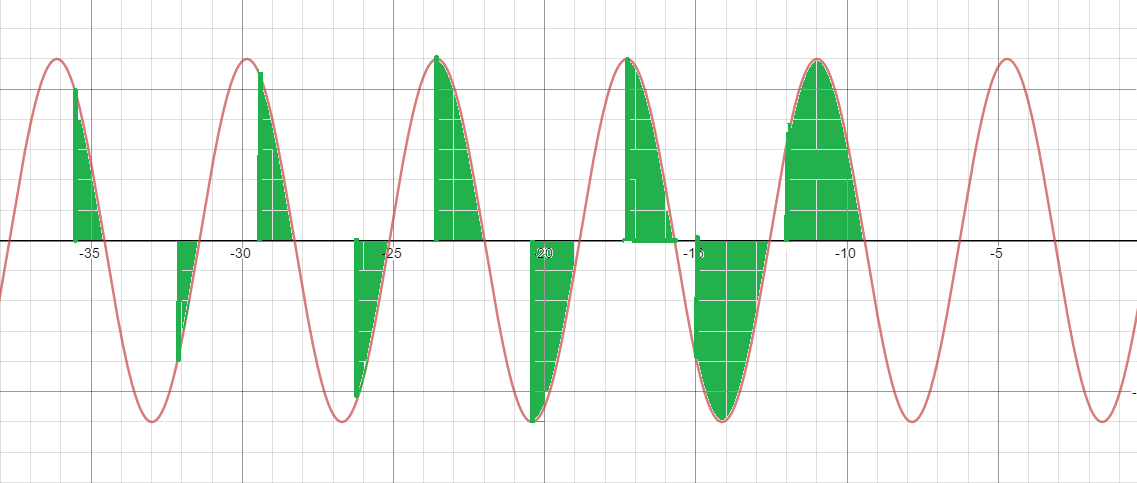
****

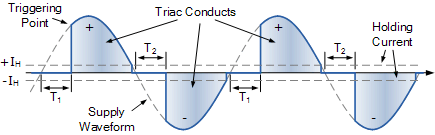
**Fig 2.10 After clicking “open camera” button Computer Vision Movement frame.**

****

**Fig 2.11 After clicking “capture” button saved captured image .**

**Gesture Movement based phase angle control part:**

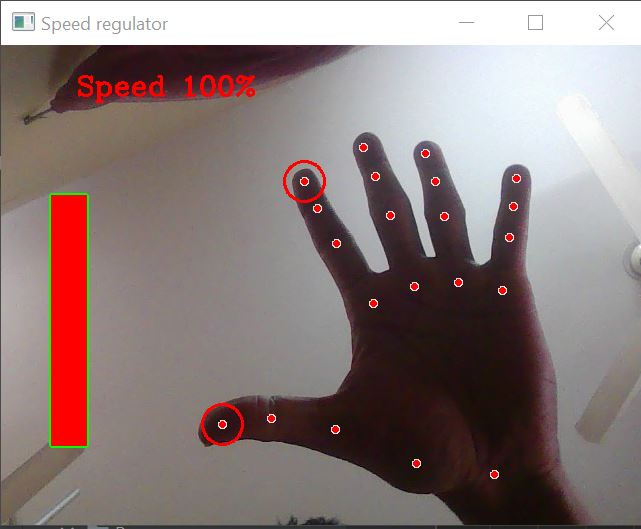
****

****

**Fig 2.12 Gesture Control phase Voltage Triggering point view.[35][34]**

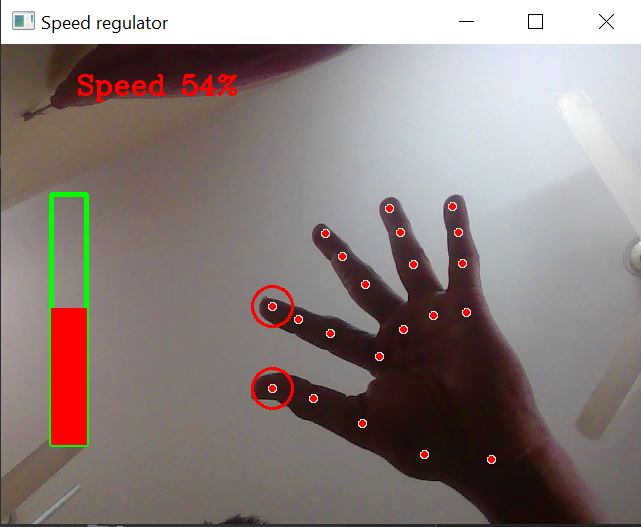
Computers can comprehend human body language with the aid of gesture recognition. Instead of relying solely on text-based or graphical user interfaces, this promotes the development of a more powerful connection between humans and technology (GUIs). The movements of the human body are interpreted by the computer camera in this gesture recognition experiment. The computer then uses this information as input to manage apps. In order to adjust the voltage phase amplitude level, this project aims to provide an interface that dynamically captures human hand gestures.

The Python programming language now has support for massive, multi-dimensional arrays and matrices, as well as a large number of high-level mathematical functions to work on these arrays, thanks to the NumPy package. A free machine learning resource is Mediapipe. Google's open-source Mediapipe machine learning library offers encapsulation for python, js, and other languages and includes some solutions for face and gesture detection. A high-fidelity hand and finger tracking solution is MediaPipe Hands. It employs machine learning (ML) to extrapolate 21 crucial 3D hand details from a single frame. It can be used to obtain the coordinates of the hand's important spots. **[28][29][30][31][32][33][34]**

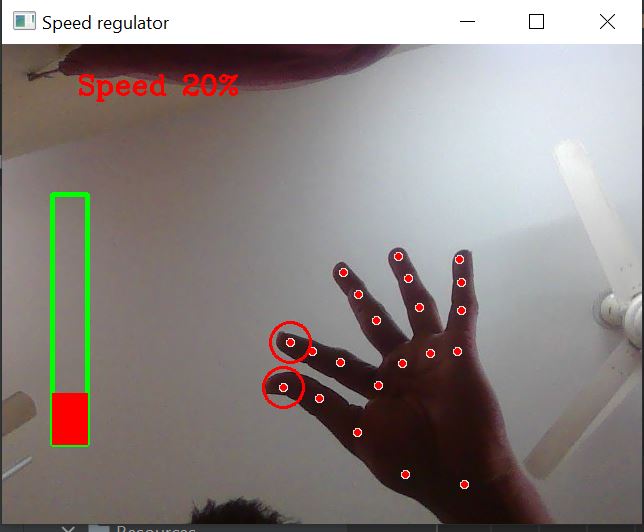
****

**Fig 2.13 Gesture Control High Level gap between index and thumb finger .**

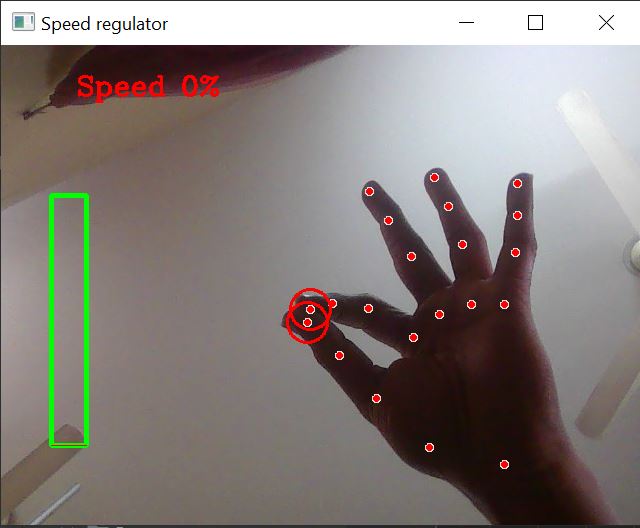
I'll be utilizing this library's hands module to make two great projects in this article. The hands module generates a localisation of my hand based on 21 points. This means that if this project provide this module an image of showing hand, it will return a 21-point vector with the coordinates of 21 significant landmarks that are located on showging hand. **[28][29][30][31][32][33][34]**

****

**Fig 2.14 Gesture Control Midium Level gap between index and thumb finger.**

****

**Fig 2.15 Gesture Control Low Level gap between index and thumb finger**

****

**Fig 2.16 Gesture Control Zero Level gap between index and thumb finger .**

****

****

**Fig 2.17 After Gesture Control Zero Level to High Level Voltage amplitude and brightness fluctuate Result.[35]**

This project makes advantage of our device's camera. It recognizes my hand as having points on it so that it can measure the space between the tips of my thumb and index fingers. The Voltage amplitude of the gadget is directly proportional to the distance between points 4 and 8.

Track down hand landmarks Measure the distance between the tips of Showing thumb and index finger. Map the distance between the tips of Showingr thumb and index finger using a Voltage amplitude range. In my situation, the distance between the tips of my thumb and index finger fell between the range of 30 to 350, and the voltage level fell within the range of 0,100. **[28][29][30][31][32][33][34]**

Chapter 2

Literature Review

2.1 Introduction

IOT Because of how quickly technology is developing and how much easier daily life has gotten, home automation has been increasingly popular in recent years. Nowadays, almost everything is digitized and mechanized. This paper proposes a system for combining sensors, actuators, and other data sources for various home automations. The system, called Toggle, works by relying on the adaptability and strength of an Application Programming Interface (API), which forms the cornerstone of an easy-to-understand and frequently-used communication mechanism. The majority of the time, Toggle is used with sensors or actuators that implement the Toggle API and have an upstream network connection.. The vast majority of Toggle's devices are made with Raspberry Pi boards and/or ESP8266/ESP8285 processors. Users using a smartphone app may now control a variety of household appliances and sensors. The Toggle system is versatile, user-friendly, and may be further improved by using a number of accessories. **[37] [38] [39] [40] [41]**

The primary goal of computer vision, a multidisciplinary field, is to perceive the surroundings as accurately as human perception can. Research on the architecture and design of computer vision systems is just as significant as research on methods and algorithms that improve a system's capabilities. In order to aid academics working in this area, we propose an open-source, Python-based End-to-End Computer Vision Framework in this paper. The framework lays a lot of emphasis on the system's adaptability and scalability due to the continual requirement to install new computer vision algorithms or machine learning models for ongoing research projects.**[37] [38] [39] [40] [41]**

One of the most natural ways to communicate with a computer is by using hand gestures, and if moving hand movements can be effectively translated in real-time, there may be a variety of uses. Using motion history images (MHI) and feedforward neural networks, the author of this study has created and designed a system that can identify motions in front of a web camera in real time. In order to capture moving portions in the picture, the backdrop from collected frames is first eliminated using a Gaussian mixture-based background/foreground segmentation technique. In order to decrease random noise, the frame is subsequently cleaned up using median filtering. By concatenating these processed frames, a cumulative motion is produced using binary thresholding and Otsu's binarization, which will determine the right threshold value.

**[37] [38] [39] [40] [41]**

**2.2Related Research/ Works Compare and Contrast**

**Related work for IOT project:**

IoT-based Smart Switching and Controlling: The use of IoT technology in smart switching and controlling systems has been the subject of much research, with an emphasis on enhancing automation, effectiveness, and safety. As a result, several IoT-based solutions for home automation, industrial control, and smart grid management have been created.

**Wireless Communication Technologies:** The evolution of wireless communication technologies like Zigbee, Bluetooth, and Wi-Fi has made it possible to integrate the Internet of Things (IoT) into smart switching and controlling systems, enabling dependable and low-latency communication between devices.

**Cloud Computing**: Due to its ability to scale computing and storage resources for large data processing as well as enable remote monitoring and management of devices, cloud computing has emerged as a key component in IoT-based smart switching and controlling systems.

**Interoperability and Standardization:** As they guarantee the seamless integration and compatibility of various systems and devices, interoperability and standardization are essential for the development of IoT-based smart switching and controlling systems.

IoT integration in smart switching and controlling systems introduces security and privacy problems since it includes the transmission of private data between devices. To overcome these difficulties, several safe and privacy-preserving methods and solutions have been investigated.

**Devices With Limited Resources:** The integration of IoT in smart switching and controlling systems poses difficulties in terms of computational and energy limits, especially for devices with limited resources as wearables and smart sensors.

**Artificial Intelligence and Machine Learning:** By enabling real-time monitoring and behavior prediction of the system, artificial intelligence (AI) and machine learning (ML) algorithms have been widely used in IoT-based smart switching and controlling systems to improve their decision-making and performance**. [37] [38] [39] [40] [41]**

**Related work for computer vision project:**

Gesture recognition based on computer vision has received a lot of research, and it has applications in fields like video games, human-computer interaction, and sign language understanding.

**Deep Learning Approaches:** Due to their capacity to learn from data and enhance performance over time, deep learning models, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs), have been widely employed in computer vision for gesture detection.

**Hand and Body Pose Estimation:** Algorithms for estimating the position and orientation of the hands and body, which are used to infer the intended gesture, are essential parts of computer vision-based gesture recognition systems.

**Multi-modal Input:** It has been investigated how to enhance the accuracy and resilience of computer vision-based gesture recognition by including multi-modal inputs such as depth information from RGB-D cameras, accelerometer and gyroscope information from wearable devices, and audio information.

**Real-time Systems:** Real-time performance has been a key focus of study in this topic since computer vision-based gesture recognition systems must work in real-time to be helpful in practical applications.

**Transfer Learning:** Transfer learning has been used in computer vision-based gesture recognition to draw on knowledge from other tasks and domains and to improve performance on a target task with less training data.

**Resource-Constrained Devices:** Research in this area has focused heavily on the integration of computer vision-based gesture recognition with resource-constrained devices, such as wearable technology and mobile phones. This poses hurdles in terms of computational and energy limits. **[37] [38] [39] [40] [41]**

**2.4 Summary**

IoT and computer vision technologies working together have the potential to change a number of industries, including gaming, smart home automation, and smart health. Through the identification of gestures, facial expressions, and body positions, computer vision may be employed in the context of IOT-based switching and controlling modules to enable real-time monitoring and control of physical devices in a more user-friendly fashion. On the other hand, IoT devices like wearables and mobile phones can offer multi-modal input, such as depth data, accelerometer data, and audio data, to increase the accuracy and robustness of gesture recognition algorithms in computer vision-based applications. Real-time performance, computational and energy limitations, data privacy and security, and the convergence of IoT and computer vision technologies bring both opportunities and challenges.

Chapter 3

Materials and Methods [or Project/System Design] [or Design Procedure]

Voltage Dimmer

**5V To Controller**

3.1 Introduction

High Voltage and power supply module

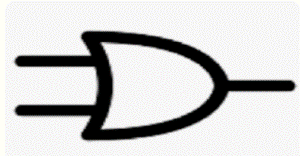
Digital output

IOT Module

Analog output

**220V to Relay** MoModule

Remote Control Module



OR Gate IC

Relay Module

IR Receiver

LOAD

**Fig 3.1 IOT and Manual Control structure diagram of this project**

The design procedure for an IoT project involves several steps that are critical to the success of the project. The first step is to identify the problem or need that the IoT system will address. This step is essential as it provides the foundation for the rest of the design process and ensures that the system is tailored to meet the specific requirements of the problem or need.

Next, research and select the appropriate hardware and software components for the system. This step involves evaluating the capabilities and limitations of different devices, sensors, and platforms to ensure that they are suitable for the project. It also includes selecting the appropriate communication protocols and data storage solutions.

After the components have been selected, the next step is to design the architecture of the system. This includes designing the communication protocols, data flow, and security measures that will be used to ensure the system's reliability and integrity.

Once the architecture has been designed, the system can be implemented, including programming the devices and integrating them with the network. This step also includes testing the system to ensure proper functionality and performance.

After the system has been tested and validated, it can be deployed in the target environment and monitored for any issues. This step is critical as it ensures that the system is operating as intended and that any issues are identified and resolved quickly.

Design Procedure Introduction for Computer Vision Project:

The design procedure for a computer vision project is a multi-step process that begins with identifying the problem or need that the system will address. This step is crucial as it defines the scope of the project and ensures that the system is tailored to meet the specific requirements of the problem or need.

[**3.2 Methods and Materials or System Design and Components**](#_Toc118501751)

**Methods:**

Identifying the problem or need that the IoT system will address.

Researching and selecting the appropriate hardware and software components for the system.

Designing the architecture of the system, including the communication protocols, data flow, and security measures.

Implementing the system, including programming the devices and integrating them with the network.

Testing the system to ensure proper functionality and performance.

Deploying the system in the target environment and monitoring it for any issues.

**Materials:**

IoT devices such as sensors, actuators, and microcontrollers.

Networking equipment such as routers and gateways.

Cloud-based platforms or servers for data storage and processing.

Power supply and backup batteries.

Cabling and wiring.

Development tools such as programming languages, software development kits (SDKs), and integrated development environments (IDEs).

Computer Vision Project:

**System Design:**

Identifying the problem or need that the computer vision system will address.

Researching and selecting the appropriate algorithms and models for the system.

Designing the architecture of the system, including the data flow, processing pipeline, and interfaces.

Collecting and preprocessing the data, such as images and videos, that the system will use to train and test its models.

Training and testing the models to ensure proper functionality and performance.

Integrating the models into the final system and testing it in real-world scenarios.

**Components:**

Camera or other image capture device

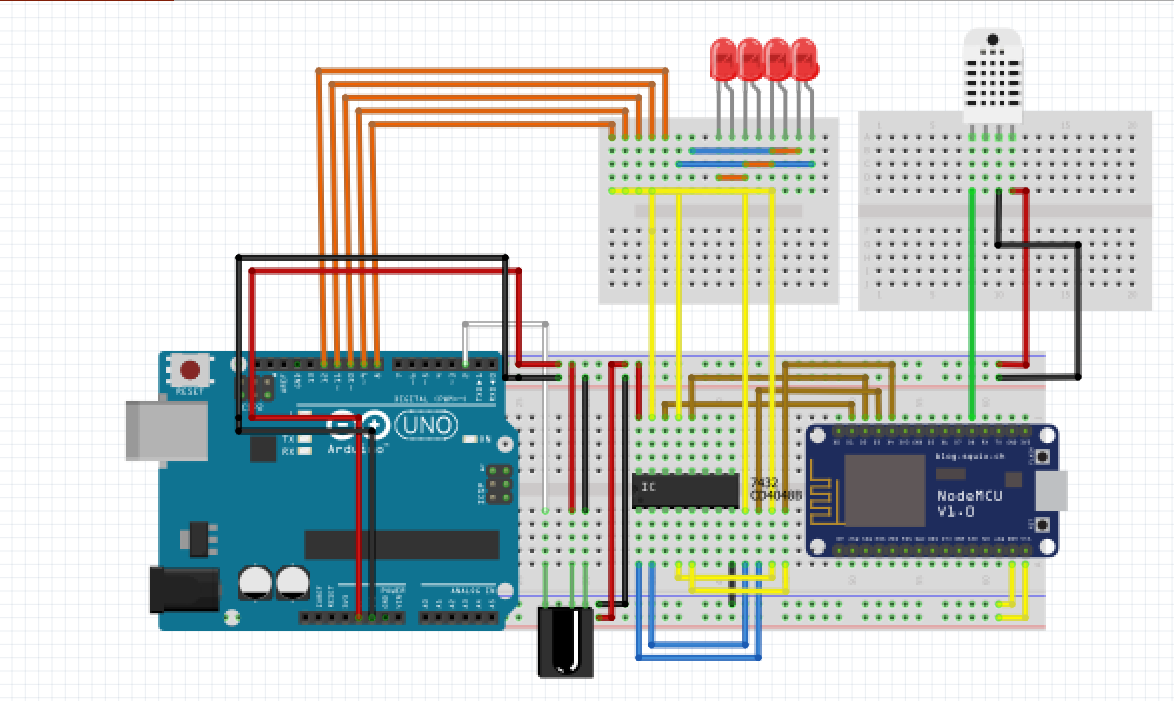
Image processing and analysis software

Algorithms such as object detection, image classification, and facial recognition

Deep learning frameworks such as TensorFlow and PyTorch

Hardware such as GPUs for training and inference

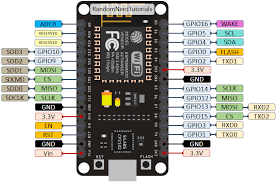
Datasets for training and testing the models.



**Fig 3.2 Component visiual Circuit diagram of this project**

Arduino uno and Node MCU ESP8266 are used as microcontrollers in this project.

Arduino Uno is used as a simple microcontroller and Node MCU ESP8266 is used to take advantage of WiFi connectivity. IR receiver sensor is used as remote control receiver. Starbrand remote is used for total control.



.**Fig 3.3 Node mcu esp8266**

A common Internet of Things (IoT) microcontroller, the ESP8266 is a low-cost WiFi device. A microprocessor, WiFi transceiver, and numerous other components are all included on a single chip in this System on a Chip (SoC). There are several variations of the ESP8266 available, including the NodeMCU, a development board that comes with the ESP8266 and a USB-to-serial converter. The Arduino IDE and other programming environments may be used to program the ESP8266. It is often used to send and receive data from cloud services as well as to connect gadgets to the Internet. It is frequently used as a WiFi access point to establish a new network or as a WiFi client to connect to an existing network. [1] [D2]



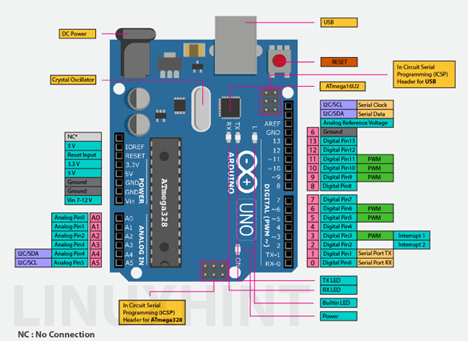
**Fig 3.4 IR Reciver**

An instrument used to detect infrared radiation and transform it into an electrical signal is known as an infrared (IR) receiver. To detect and amplify the IR signal, it is made up of a photodiode or other IR-sensitive semiconductor device and related circuitry. Remote controls, burglar alarms, and other devices that employ IR signals for communication frequently use IR receivers in a number of applications. They frequently function in tandem with IR transmitters, which send IR signals.IR receivers are available in a variety of packages and configurations, including surface mount devices (SMDs), through-hole devices, and modules with built-in amplifiers and other components. They can be interfaced with microcontrollers and other electronic devices using a variety of methods, including digital and analog inputs. [D6]



**Fig 3.4 IR transmitter Star universal brand remote.**

A device that transmits IR signals is known as an infrared (IR) transmitter. It is made up of a light-emitting diode (LED) or other IR-emitting device, together with any necessary electronics for generating and modulating the IR signal. Remote controls, burglar alarms, and other devices that communicate via IR signals frequently employ IR transmitters in a variety of applications. They frequently work along with IR receivers, that are employed to find the IR signals. A typical example of a gadget that makes use of an IR transmitter is a TV remote control. An IR receiver on the TV picks up the IR signals that the remote control delivers, which are then received by the TV. The signals contain data that the TV utilizes to regulate its operation, such as the channel number or voltage amplitude level. [D7]



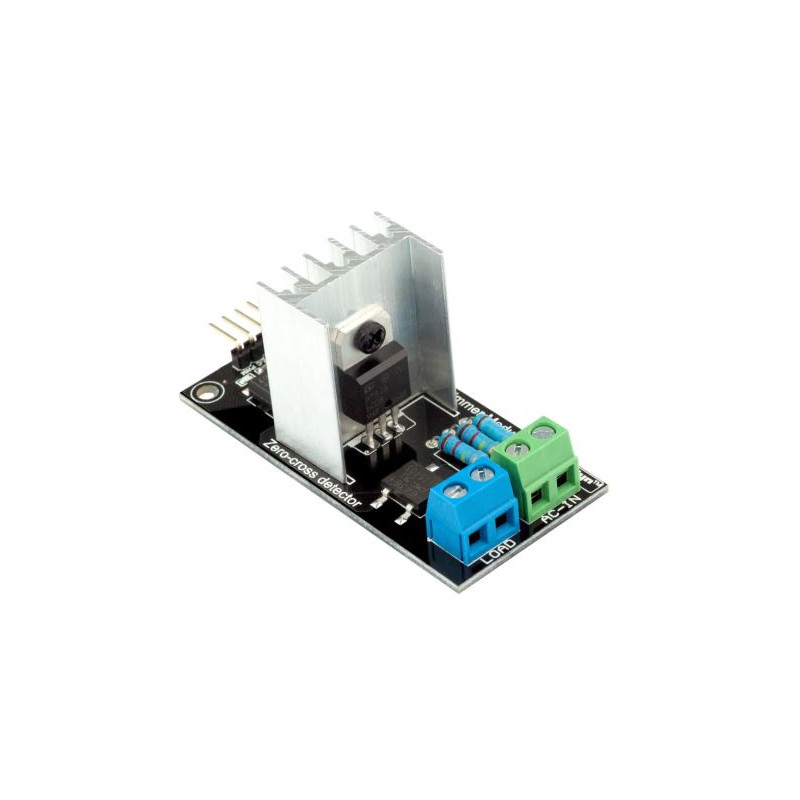
**Fig 3.5 Arduino uno r3**

A collection of microcontroller boards and an integrated development environment (IDE) for writing and uploading code to the boards make up the open-source hardware platform known as Arduino. A well-liked variation of the Arduino platform, the Arduino Uno, is built on the ATmega328 microprocessor. The Arduino Uno contains a 16 MHz crystal oscillator, 6 analog input pins, 14 digital input/output pins, a USB port, a power connector, an ICSP header, and a reset button. It may be powered by an external power source or a USB connection. The Arduino IDE, which runs on Windows, Mac, and Linux, may be used to program the Arduino Uno. A library of pre-written code called "sketches" is available in the Arduino IDE, which employs a form of C++, and may be edited and uploaded to the Arduino Uno. [D1]



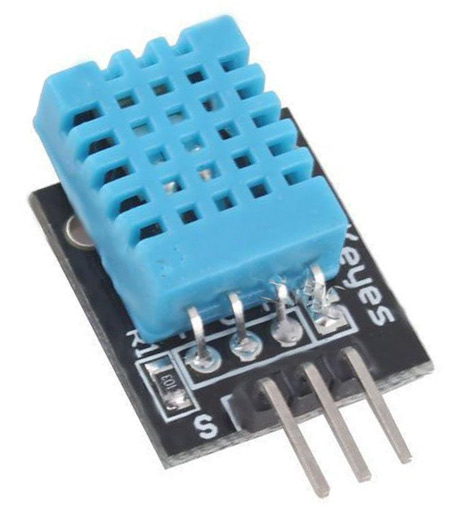
**Fig 3.6 Bulb as an Load**

As loads, four tungstens were employed. Utilizing tungsten balls to facilitate display. A 220 volt supply is offered for use. A tungsten filament serves as the light source within a 220V tungsten bulb, which is a kind of incandescent light bulb. Due to its low cost and high performance, tungsten bulbs are extensively utilized in a range of applications, including home and commercial lighting. A thin tungsten filament is heated to a high temperature by an electrical current in tungsten bulbs to produce light. Due to the filament's high temperature, light is then produced.The intensity of the light produced by a tungsten bulb is directly proportional to the filament temperature, so increasing the filament temperature will increase the brightness of the bulb.



**Fig 3.7 RBD Voltage Dimmer**

Voltage dimmers are used to control the voltage from a remote location as Voltage dimmer. There is a field ready detector and pwm modulation system. It is mainly composed of zero crossing detector circuit and triac based drive circuits. An RBD voltage dimmer is a type of electronic dimmer that uses a phase-cut technique to control the brightness of a lighting load. It operates by rapidly turning on and off the electricity to the load. The average voltage to the load may be changed by varying the ratio of on-time to off-time. In turn, this makes it possible to regulate the load's brightness. RBD dimmers come in a variety of form factors, such as wall-mounted dimmers, in-line dimmers, and dimmer switches, and are often used in residential and commercial lighting applications. They frequently work with a variety of lighting loads, including halogen, incandescent, and some types of LED lights. [D9]



**Fig 3.8 DTH11 Sensor**

Temperature and humidity are sensed with a DTH11 sensor. essentially runs on five volts. gives the system information by way of digital output.

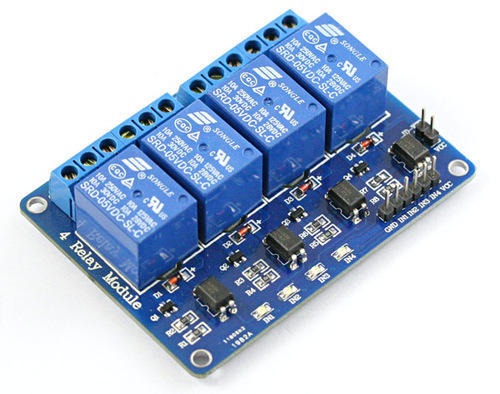
A common humidity and temperature sensor used in do-it-yourself applications is the DHT11. It may be purchased as a single sensor or as a module with a 4-pin connection and a microcontroller. Through the 4-pin connection, the DHT11 delivers a digital output while also measuring humidity and temperature. It has a temperature range of 0 to 50 °C and a relative humidity range of 20 to 80%.. It has an accuracy of +/-5% RH and +/-2°C. The DHT11 can be interfaced with a microcontroller or a single-board computer such as an Arduino or Raspberry Pi using a 4-pin connector or a set of wires. The data can then be processed and used to control devices or to display the humidity and temperature on a display or online. [D3]



**Fig 3.9 The MQ6 Gas Sensor**

Gas sensing is done with the MQ6 sensor. essentially runs on five volts. gives the system information by way of analog output. Additionally, there is a variable nobe system on the sensitivity increase or decrease.

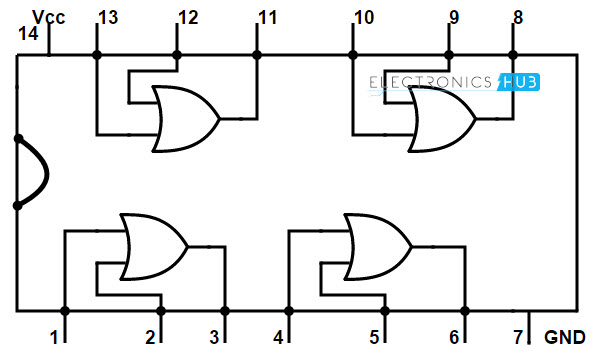
Gas sensors like the MQ6 are frequently used to find combustible gases including propane, butane, and methane. It may be purchased as a single sensor or as a module with a 4-pin connection and a microcontroller. A metal oxide semiconductor (MOS) element's electrical resistance is measured by the MQ6 sensor in order to function.which changes in response to the presence of gas molecules. The sensor outputs an analog voltage that is proportional to the concentration of gas in the air. The MQ6 sensor can be interfaced with a microcontroller or a single-board computer such as an Arduino or Raspberry Pi using a 4-pin connector or a set of wires. The data can then be processed and used to control devices or to display the gas concentration on a display or online. [D4]



**Fig 3.10 5v four channel Relay Module**

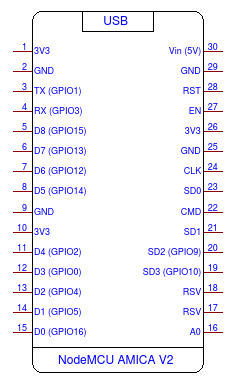
4-channel relay is used as a switching device in this project . Basically operates on five volts. Here a negative input of 5 volts has to be given to operate.

A 4-channel 5V relay is an electromechanical switch that is controlled by an electrical current. It consists of a coil, which is activated by a small DC voltage, and a set of contacts that open or close when the coil is energized. A 4-channel 5V relay has four separate channels, each with a set of normally open (NO) and normally closed (NC) contacts. When the relay is activated, the contacts on the corresponding channel change state, either connecting or disconnecting the circuit. A 4-channel 5V relay can be controlled by a microcontroller or other electronic device using a small DC voltage, typically in the range of 5V. The relay can be used to control a variety of electrical loads, including motors, lights, and other devices. [D5]



**Fig 3.11 Quard OR gate IC 7432**

The IC for the or gate is 7432. This Orgate's operations will be managed by both remote control and IOT control. A quad 2-input OR gate is the 7432. It is a digital logic gate that combines two binary inputs using the OR operation. If either or both of the inputs are 1, an OR operation yields a 1, and if both inputs are 0, it returns a 0. Four distinct 2-input OR gates, each with its own set of inputs and outputs, are available on the 7432. It comes in many packaging kinds, such as DIP, SOIC, and SOP. Numerous digital circuits, including those in computer systems, logic systems, and other electronic systems, can make use of the 7432.. It is often used in combination with other logic gates to create more complex circuits. [D8]

****

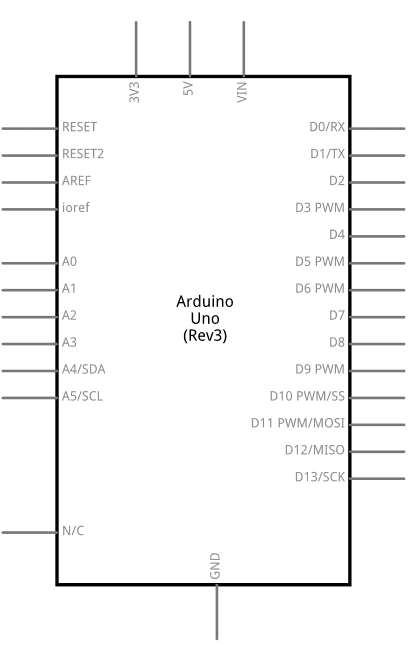
**Fig 3.12 Scametric arrangement for Node mcu esp8266**

IoT control uses the Nodemcu esp8266 microcontroller. This is a development board for WiFi.

The voltage Dimmer Module's output and zero cross detector pins are designated as D0 and D1 respectively.

The temperature sensor's input is received on pin D7.

Relay control is accomplished using pins D5, D6, D2, and D8.. [D2]

****

**Fig 3.13 Scametric arrangement of arduino uno for manual control**

Arduino Uno is used as microcontroller for manual control.Connected to it is the remote control section.

With microcontroller digital 2 number pin is used as receiver pin from ir sensor . Digital Pin 8,10,11,12 is used for IR remote control based relay controlling. [D1]

[**3.3 Design Specifications. Standards and Constraints**](#_Toc118501752)

Display and Detecting Section in Computer

And Image Capturing section

Processing by Developed Computer Vision Software

Sending Through Internet

Video Recognize

section

**fig 3.14 Block diagram of Computer vision processing.**

After converting a .exe file will become a software

Project arrangement with the required frame

Required coding is in Python Language

**fig 3.15 Developing Block step For Computer vision Software**

**3.4 Simulation/Experimental Setup**

**Standards:**

The system has to abide by all applicable legal and regulatory requirements regarding data security and privacy.

The system must adhere to pertinent image and video processing standards like OpenCV.

The system has to adhere to the necessary requirements for deep learning frameworks like TensorFlow and PyTorch.

The system must function within the limitations of the budget and resources that are available.

The limitations of the available data and the quality of that data must be taken into account by the system.

The system must function within the limitations of the processing resources and power that are available.

The system must function within the boundaries of the moral and legal concerns around the use and processing of data. [3] [6]

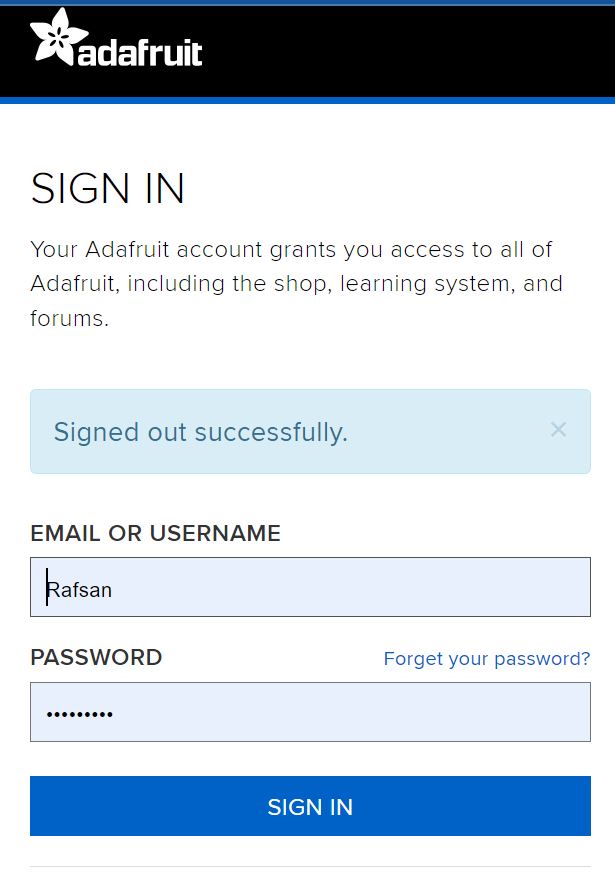
**Entering or Sing in Part**

**Feed Creating Part**

**Dashboard Creating Part**

**Block creating And feed injecting on dashboard part**

**fig 3.16 IOT platform Creating block step on adfruit server .**

****

**fig 3.17 IOT platform Creating step 1**

The first step in creating a IOT control block is to complete the signin part. And have to enter the Enter with verified mail.

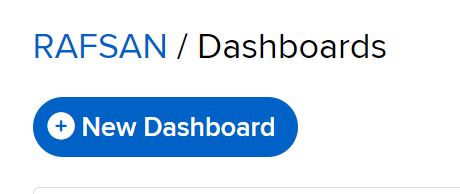
Because all subsequent information will go to email and all other mail-based policy controls will come here.

****

**fig 3.18 IOT platform Creating step 2**

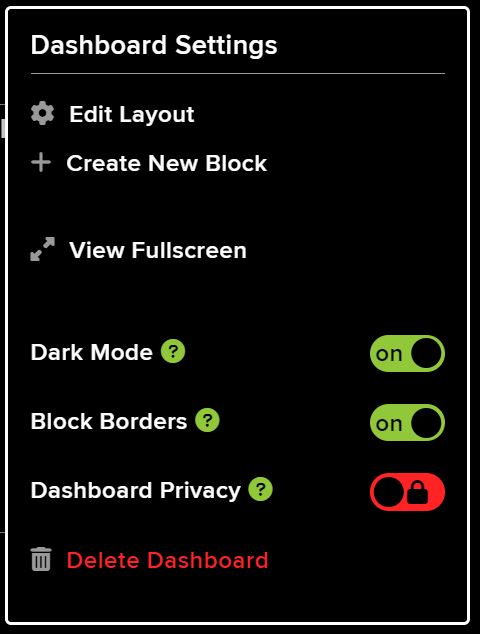
when the sign-in process is finished. I'll create feeds. I will use this to submit data from various locations and do out display chores on the sent write.

This will be referred to as the IOT block's second component. The foundation of Adafruit IO is feeds. Both the uploaded data and the meta-data about the data that sensors submit to Adafruit IO are stored in them. the time and date that it was posted, for instance. Alternatively, the GPS locations where the data originated. Do you want to send that data flow to the entire globe, a particular audience, or simply a friend? To provide (and restrict) access to data, Adafruit IO has included feed sharing with fine-tuned privacy settings. [10]

****

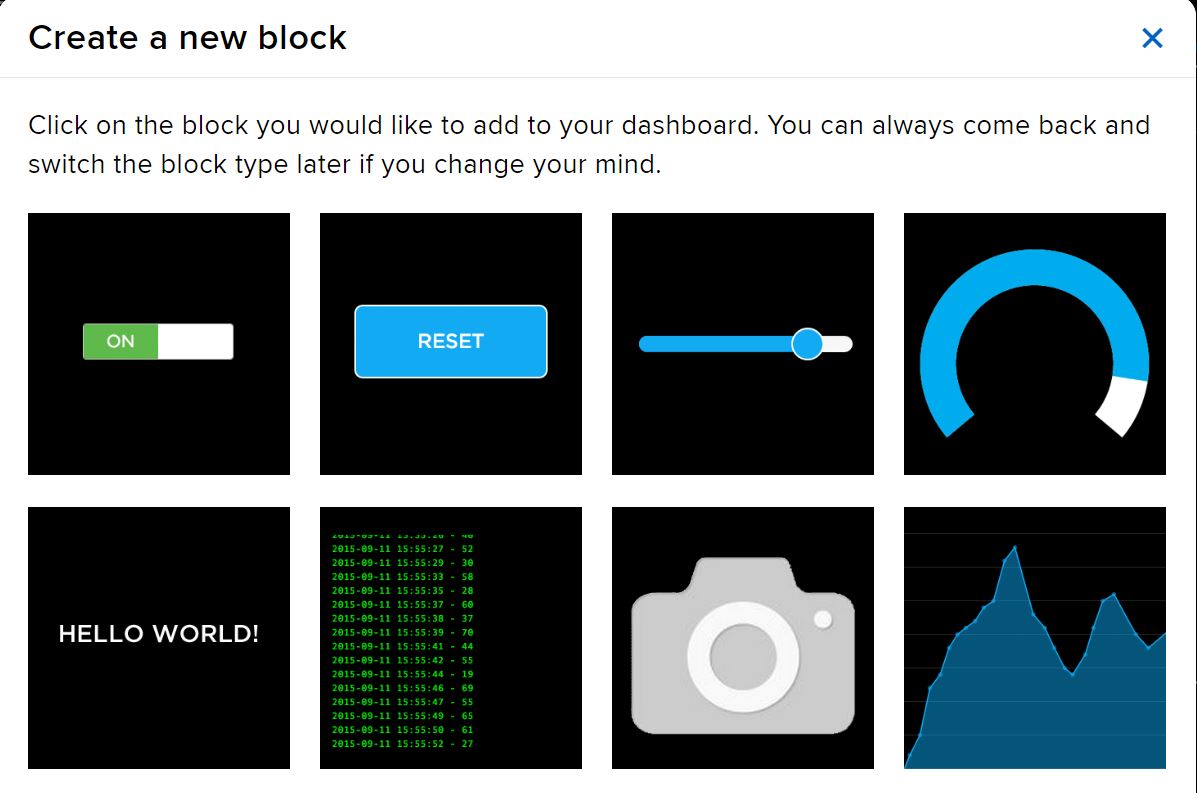
**fig 3.19 IOT platform Creating step 3**

Creating Dashboards for Service will be the project's third component. My returning data may be floatedly presented on the dashboard, and I'll have a simulated input and output system. Adafruit IO has a feature called dashboards that lets users chart, graph, measure, log, and display data. The dashboards may be viewed from anywhere in the world.. [11]

****

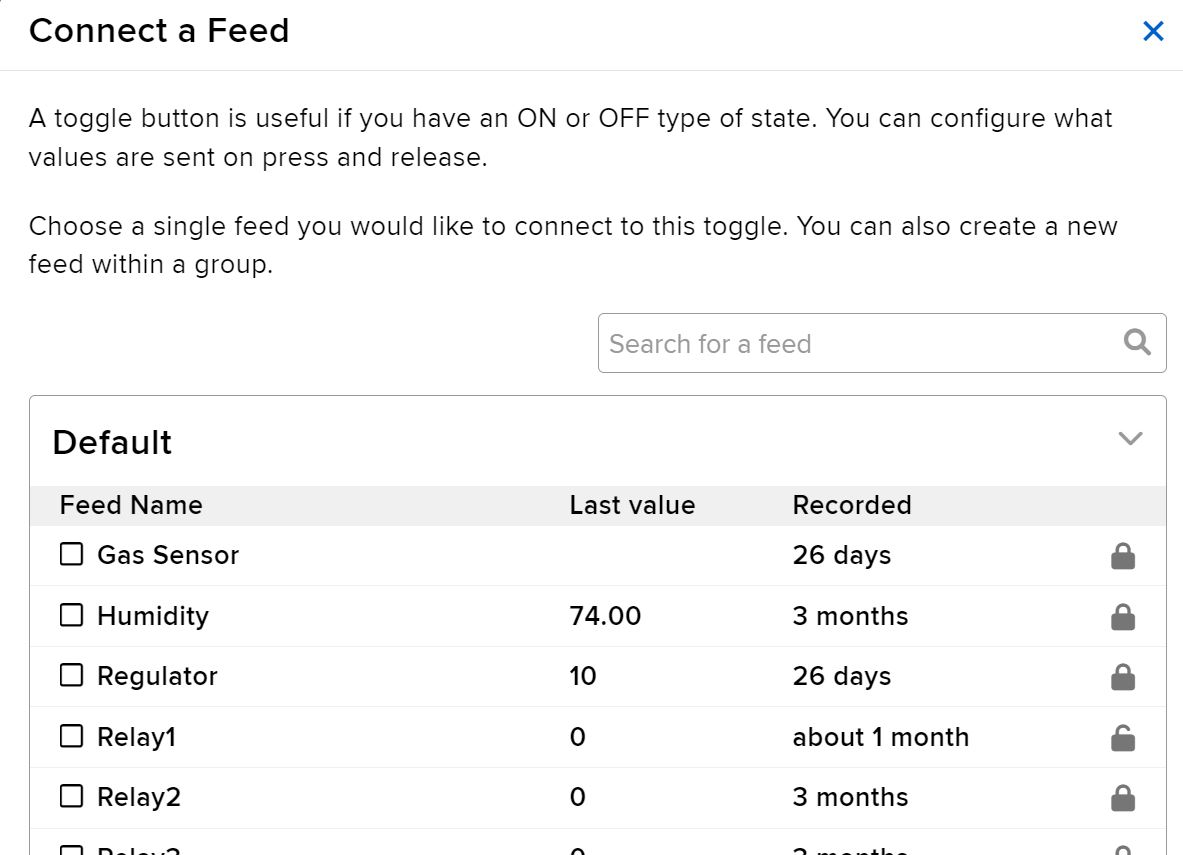
**fig 3.20 IOT platform Creating step 3**

Through this window i can create different types of action blocks and include dashboards in all those block. Block with which I are actually control This project .

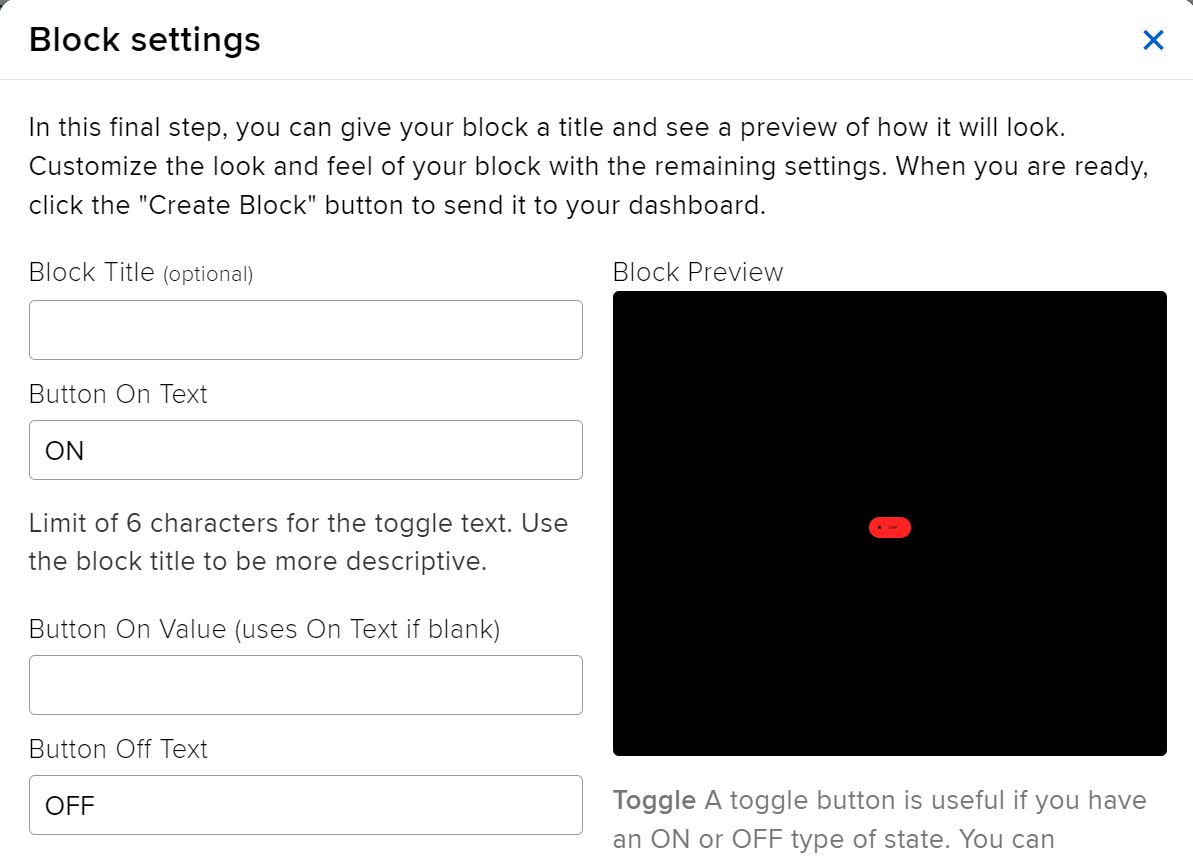
****

**fig 3.21 IOT platform Creating step 4**

And later, the data of the gods can be replaced in the specific key block through the trick. And here are the types of block electronics options that i can see with different types of block . Here it is possible to perform various processe ,dispaying and controls by silating the block according to my needs. [10]

****

**fig 3.22 IOT platform Creating step 5**

****

**fig 3.23 IOT platform Creating step 6**

And the last figure shows the final state of the block by creating a block. It has to input various parameters starting from the value budget amount of input output and there initial value. [10]

**Gesture movement based Voltage Amplitude Controller Making Steps:**

**Video Capture:** Use OpenCV's VideoCapture class to access the webcam video feed in order to capture it. use Python tools like Mediapipe NumPy, Pandas, SciPy, and scikit-learn to help with this process.

**Video pre-processing:** To isolate the hand region in the frame, convert the video frame to grayscale and use thresholding.

Identify and extract the hand contour from the threshold image using OpenCV's contour detection to recognize hand motions.

**Recognize gestures:** Using hand contour data, train a computer learning system, such as a convolutional neural network (CNN), to recognize particular hand motions.

**Control voltage:** Create code to relate changes in my system's phase voltage to recognized gestures.

**Put the system in place:** Create a comprehensive system that combines all of the parts and employs gesture control to change the phase voltage via the camera.

**Identify gestures:** Train a machine learning model, such as a Convolutional Neural Network (CNN), to recognize specific gestures based on the processed video frames.

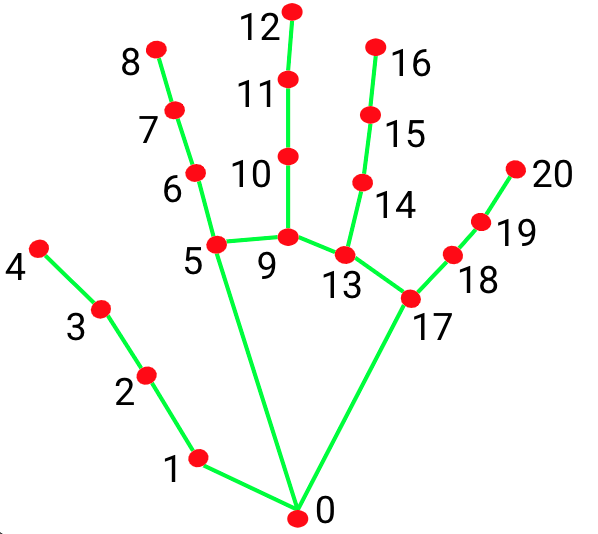
**Map gestures to control:** Write code that maps the recognized gestures to changes in the phase voltage of this project. Integrate system components: Integrate all of the components into a complete system that uses gesture control to adjust the phase voltage based on the webcam input.

An open-source platform for developing hand and body tracking applications is called MediaPipe. It includes a pre-trained TensorFlow-based hand identification model that may be used to identify hands in real-time video captured by a webcam or mobile device. Single-shot multi-box detection (SSD) is the foundation of the hand detection model, which has been real-time performance enhanced.

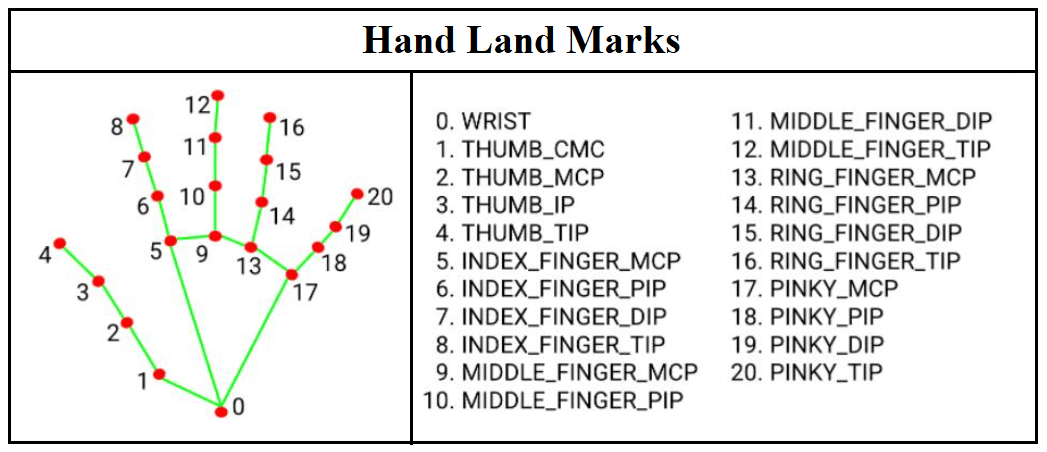
**Install MediaPipe**: To install MediaPipe and its dependencies, refer to the installation guidelines provided by operating system.

Run the hand detection pipeline: To run the hand detection pipeline on a video stream from a webcam or mobile camera, use MediaPipe's command line interface.

Imagine the outcomes: By processing the output in Ir own code or using MediaPipe's built-in visualizers, i can visualize the hand detection pipeline's output of hand bounding boxes and landmarks in the video stream.

****

**Fig 3.24 21 Hands Landmarks point [35]**

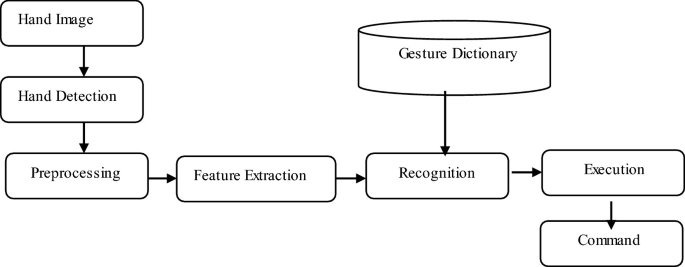
****

**Fig 3.21 25 Hands Landmarks point Specification [35]**

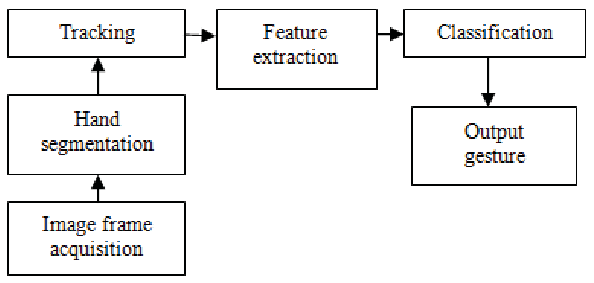
The hand detection process may be modified, and I can add my own code to handle the output or change the configuration file. For instance, I might utilize the hand detection data to operate a user interface or deal with 3D objects.

Hand landmarks, which are specific locations on the hand such the fingers, knuckles, and wrist, can be used to characterize the form and placement of the hand. In MediaPipe, hand landmark estimates are made using the hand bounding boxes generated by the hand detection model.

The hand landmarks are calculated using a machine learning model that was trained on a big dataset of hand pictures.The image plane's landmarks' 2D coordinates are output by the model. The landmarks can then be applied to tasks like 3D hand tracking, hand posture estimation, and hand gesture recognition.

****

**Fig 3.26 Gesture Control Steps 1 [35] [30] [32]**

****

**Fig 3.27 Gesture Control Steps 2 [35] [30] [32]**

**Serial Communication**

**Hand Gap Detecting**

**Section and decision point Creating**

**Microcontroller**

**Processing**

**Fig 3.28 Block steps of Gesture Control Development**

**[35]**

Chapter 4

Results and Discussions

4.1 Results

Enhanced control and monitoring of physical systems and equipment, including the temperature and humidity in a warehouse.

Procedures that control a building's lighting and HVAC systems are more automated and efficient.

Expanded data gathering and analysis, which may be utilized to inform choices and enhance performance. Increased system, device, and human connectedness and communication.

Enhanced system security and dependability. Enhanced system perception and comprehension of visual input, such as pictures and films. Tasks like object identification, picture categorization, and facial recognition will be more accurate and faster. Increased automation and efficiency of processes that rely on visual data analysis.

Improved decision-making and actionable insights by providing the system with the ability to extract meaningful information from the visual data.

Improved security and safety by providing the system with the ability to identify and respond to potential threats or anomalies.

Through this chapter i will know how to control my system in different ways and what will be the results. IFTTT stands for If This Then That. It is a free web-based service for creating chains of simple conditional statements, called as applets. These applets are triggered by changes that occur within other web services such as Gmail, Facebook, Telegram, Instagram, Google Assistant or Pinterest. For instance, an applet is capable of sending an e-mail message if the user tweets using a hashtag, or copy a photo on Facebook to a user's archive if someone tags a user in a photo. In addition to the web-based application, the service runs on IOS and Android. [7] [5] 26]

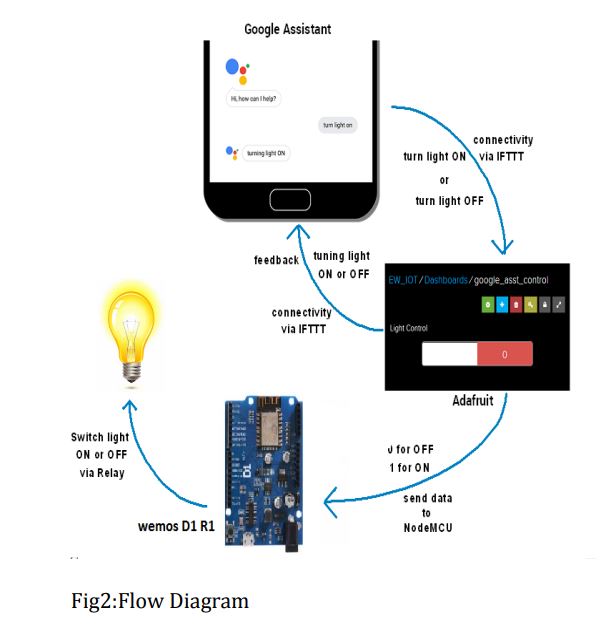


Fig: Flow Diagram [26]



Fig 4.1 IFTTT Service and Appleates1

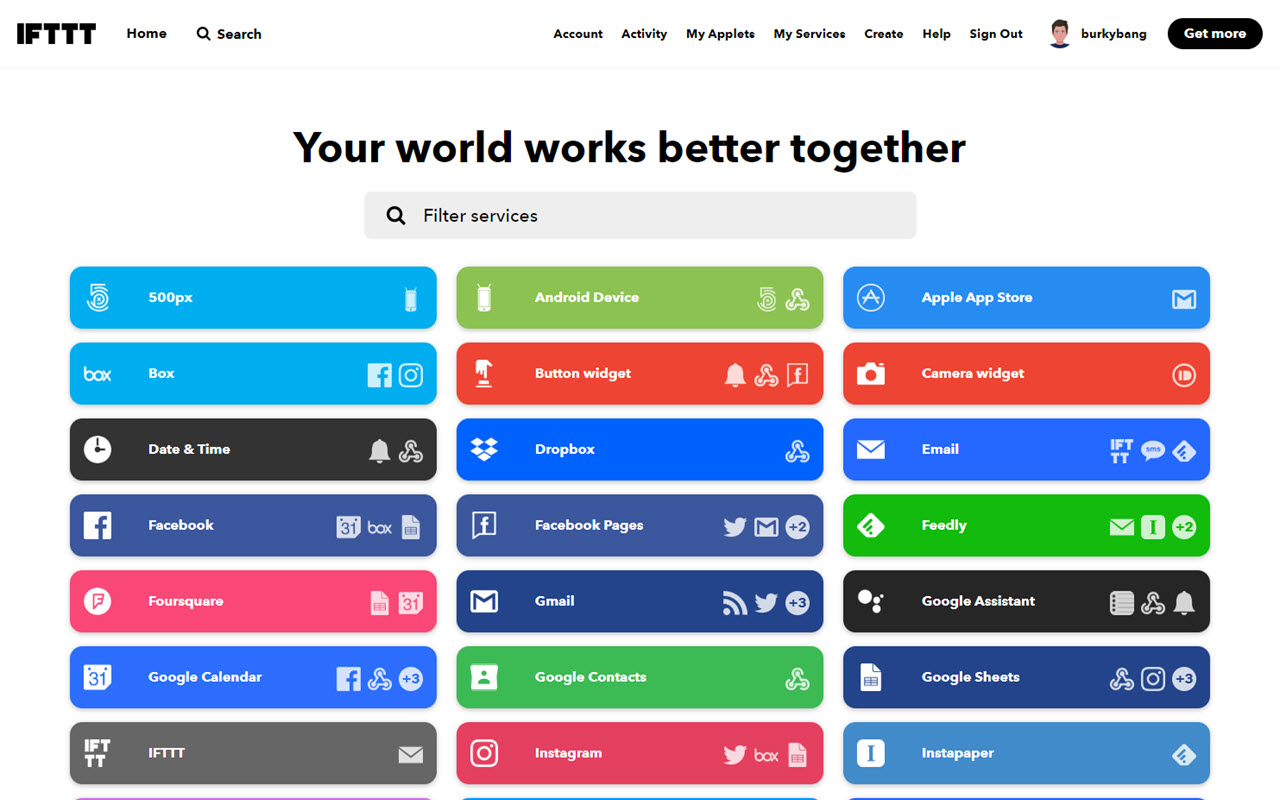


Fig 4.2 IFTTT Service and Appleates2

A web server named IFTTT serves as the primary controller of the IOT project component. making use of this server. It is feasible to manage several dimensions without altering the original source code by utilizing only this Server. Amazon, Alexa, Fitbit, Gmail, Discord, Dropbox, Android device, and Android SMS. Along with this, you may also use them for a number of other things. [26]

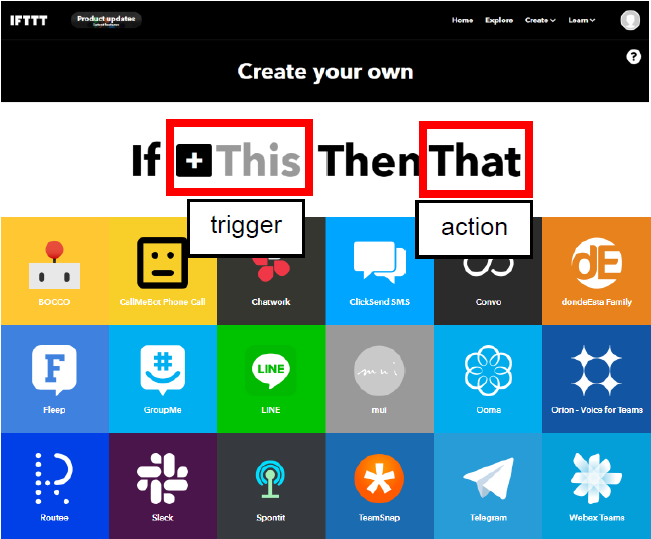


Fig 4.3 IFTTT Service and Appleates3

This enables a separate web server to set a control depending on a data point and behave accordingly. More than a 1000 services and apps are available. How to link this project to the IFTTT server to make all forms of controlling it extremely simple. Particularly, the task of data collecting may be done quite effectively. One can use an example. Controlling depending on any form of data can email the user alert messages. Much more could be done, similar to this effort.. [26]



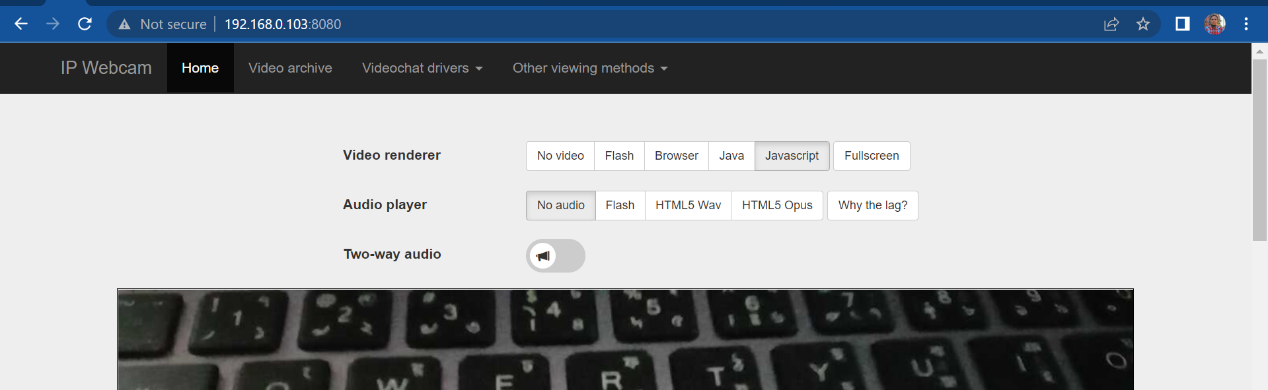
Fig 4.4 Self-developed apps for use in projects

An attempt has been made to develop the total controlling protocols used in the project into an app[16]



Fig 4.5 Computer vision or video visualization protocols

An application called IP Webcam is used for computer vision or video visualization. For now, their own mobile phones have been used for object movement detection in Computer vision.

Fig 4.6 IP link http://192.168.0.103:8080/ based computer vision or video visualization through Internet.

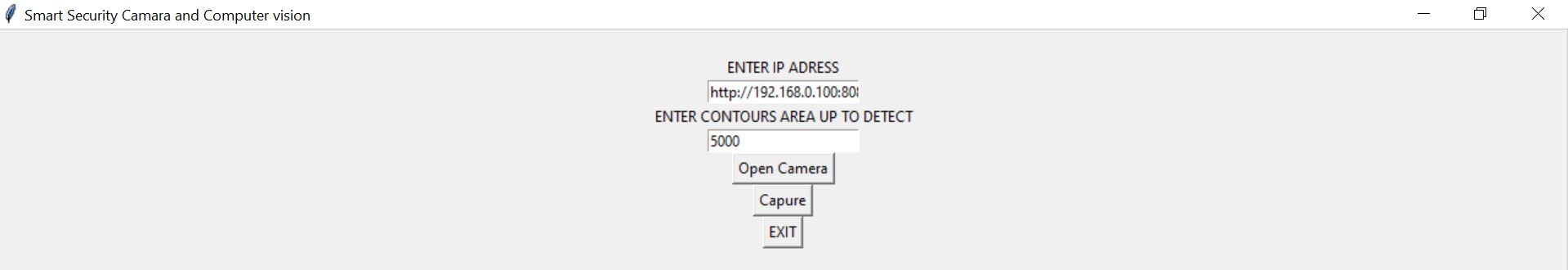


Fig 4.7 IP link <http://192.168.0.103:8080/> and contours area selecting based computer vision or video visualization software through Internet.

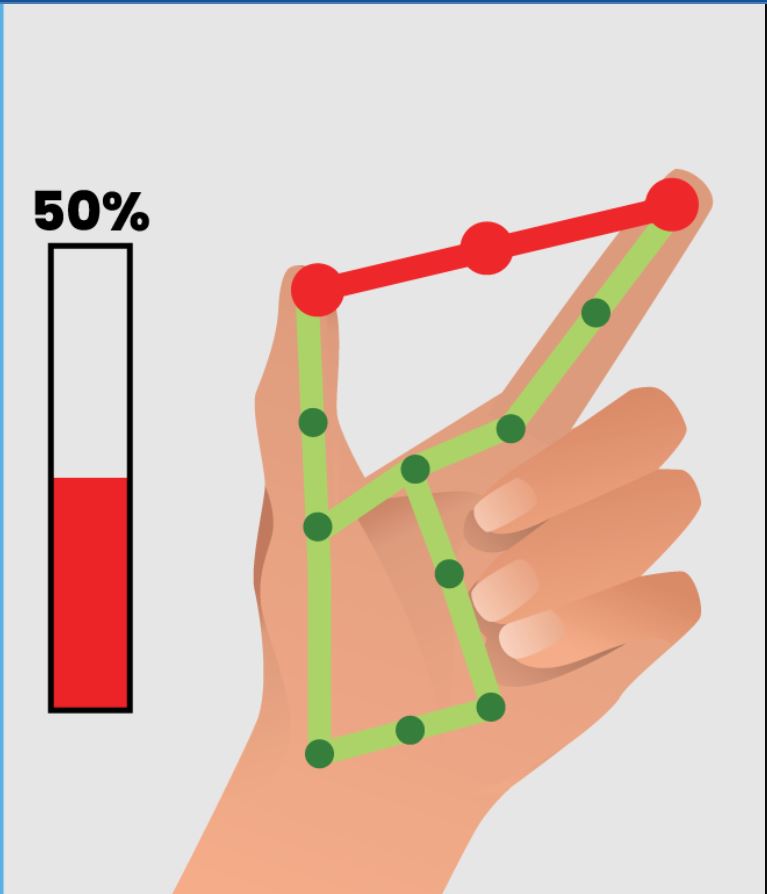


Fig 4.8 Index and thumb Finger Gesture Movement based phase angle control part .

Discussions for IoT (Internet of Things) Project:

Data Security and Privacy: One of the main concerns with IoT systems is the security and privacy of the data that is collected and transmitted. This includes ensuring that the data is protected from unauthorized access and that it is only used for the intended purposes.

Scalability: Another important consideration for IoT systems is scalability. As more devices and systems are connected to the internet, the amount of data that needs to be processed and stored increases. This requires a robust architecture that can handle large Voltage amplitude s of data and support the addition of new devices and systems.

Interoperability: IoT systems often need to integrate with existing systems and devices. This requires the use of common communication protocols and data storage solutions to ensure that the systems can communicate and share data seamlessly.

Power and Energy Efficiency: IoT systems often rely on battery-powered devices, which requires careful consideration of power consumption to ensure that the devices can operate for extended periods without needing to be recharged. [4] [5] [1] [26]

Discussions for Computer Vision Project:

Data Quantity and Quality: The system's performance is greatly influenced by the quantity and quality of the data used to develop and evaluate computer vision models. To ensure the system's capacity to generalize and adapt to new settings, it is essential to ensure that the data are correct, diverse, and adequate.

Selection and fine-tuning of algorithms Performance of a computer vision system depends on the models and algorithms utilized. It is crucial to thoroughly assess the strengths and weaknesses of various strategies and to fine-tune the models to maximize their performance for the particular job at hand. Bias and Fairness: Computer vision systems are prone to biases if not designed and trained carefully. These biases can lead to errors and inaccuracies in the system's performance, particularly when analyzing images and videos of certain groups of people. It is important to consider these issues and take steps to mitigate any potential biases in the system.

Real-world deployment and upkeep: The deployment of computer vision systems in real-world situations calls for a strong and dependable infrastructure to make sure the system is capable of operating in various contexts, handling errors, and adapting to changing conditions. To ensure the system's functionality and security, it also has to be maintained, watched over, and updated. [9] [7] [6]

Chapter 5

Project Management

5.1 Task, Schedule and Milestones

All things together there were very few resources. This project had to be completed by combining parts from various sources. It is the superior know-how of C++ that has helped the most in its creation.

Research and Analysis (Week 1-4)

Task: Research current IoT and computer vision solutions

Schedule: 4 weeks

Milestone: Completion of research and analysis report

Design and Development (Week 5-12)

Task: Design user interface for the central management platform and develop the platform architecture

Schedule: 8 weeks

Milestone: Completion of the design and development of the platform

Computer Vision Algorithms Development (Week 5-8)

Task: Develop computer vision algorithms and integrate with the platform

Schedule: 4 weeks

Milestone: Completion of computer vision algorithms development and integration

Deployment and Testing (Week 9-12)

Task: Deploy platform to a small group of early adopters for additional feedback

Schedule: 4 weeks

Milestone: Completion of the deployment and testing phase

Data Analytics and Visualization (Week 13-16)

Task: Develop data analytics and visualization tools

Schedule: 4 weeks

Milestone: Completion of data analytics and visualization tools

Launch and Maintenance (Week 17-20)

Task: Launch the platform to the public

Schedule: 4 weeks

Milestone: Completion of the launch and maintenance phase.

5.2 Resources and Cost Management

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | | Arduino Uno | | 1000 |
| 2 | | Node Mcu ESP8266 | | 500 |
| 3 | | Remote and IR sensor | | 200 |
| 4 | | Bulb | | 200 |
| 5 | | Wire | | 200 |
| 6 | | Bread board | | 150 |
| 7 | | DTH11 Sensor | | 75 |
| 8 | | Gas Sensor | | 75 |
| Total |  | | 2550 | |

There are several online based robotics shops in Bangladesh of which Robotics BD is one of them.

From these I collected most of the material in This project .

5.3 Lesson Learned

While doing this project, I had to get acquainted with different sources. Here I need programming knowledge and basis of electronic. And the project had to change many things to match the fantasy with the reality. I was able to match the practical subjects with the theoretical subjects of the book.

Chapter 6

Impact assessment of the project

6.1 Economical, Societal and Global Impact

An Impact Assessment Of An IOT Project Involves Evaluating The Potential Effects And Consequences Of Implementing IOT Project On Various Stakeholders. This Can Include The Economic Impact On Businesses And Consumers, The Social Impact On Individuals And Communities, And The Environmental Impact On Natural Systems.

By conducting an impact assessment, organizations can better understand the potential consequences of implementing IoT project and take steps to ensure that it is deployed in a responsible and sustainable manner.

This project is almost entirely internet dependent. But there is also a system of remote control in nearby places. Its use will not require direct human or human presence. In this case, I think the presence of machines will increase instead of human workplace presence. Human dependence on a single machine and human skilling residency will be created for generations to come. [5] [7] [8] [10] [11]

**6.2 Environmental and Ethical Issues**

**Environmental Issues**: Energy consumption: IoT and computer vision systems require a significant amount of energy to operate. This can lead to increased greenhouse gas emissions and contribute to climate change.

**E-waste:** The rapid advancement of project may result in the rapid obsolescence of IoT devices and computer vision systems. This can lead to an increase in electronic waste and potential harm to the environment if the devices are not disposed of properly.

Resource depletion: The production of IoT devices and computer vision systems can consume a significant amount of natural resources, including materials such as metals and plastics.

**Ethical Issues:**

Privacy: IoT devices and computer vision systems can collect a large amount of personal data, which can be used for targeted advertising, surveillance, or other purposes. This can raise concerns about privacy and data security.

Bias: Computer vision systems can inadvertently perpetuate biases present in the data used to train them. This can lead to discriminatory outcomes for certain groups of people.

**Employment:** IoT and computer vision systems can automate tasks and processes, which can lead to job displacement. This can raise concerns about the impact on workers and the broader economy.

**Safety:** IoT and computer vision systems can be used in critical infrastructure such as transportation, healthcare, and public safety. If they are not designed and implemented properly, they can pose a risk to human lives.

**Transparency:** When the decision-making of computer vision systems are opaque, it can be difficult to understand how the system arrived at a certain decision. This can make it hard to identify and correct errors, and can be seen as a lack of transparency. [9] [5] [11]

**6.3 Utilization of Existing Standards**

**Utilization of Existing Standards:**

**Communication protocols:** IoT devices and computer vision systems rely on a variety of communication protocols to transmit data, such as TCP/IP, Zigbee, and Bluetooth. Adhering to established communication protocols ensures that devices can effectively communicate with one another and with the central management platform.

**Data formats:** Data collected from IoT devices and computer vision systems must be stored and analyzed in a consistent format. Utilizing established data formats such as JSON or XML ensures that the data can be easily analyzed and understood.

**Security standards:** IoT devices and computer vision systems handle sensitive data and must be secure to protect against hacking and unauthorized access. Utilizing established security standards such as AES, SSL, and HTTPS ensures that the data is protected and that the system is secure.

**Interoperability standards:** IoT devices and computer vision systems often need to work together and with other systems. Utilizing established interoperability standards such as OPC UA, MQTT, and CoAP ensures that the devices can communicate and function together seamlessly.

**Accessibility standards:** The platform and application should be accessible to all users, including those with disabilities. Utilizing established accessibility standards such as WCAG 2.0 and Section 508 ensures that the platform and application are accessible to all users.

**Ethical Standards:** The project should adhere to ethical standards of privacy, data security, and transparency. Utilizing established ethical standards such as GDPR, CCPA, and HIPAA ensures that the project is compliant with regulations and ethical guidelines. [1] [5] [9] [10] [11] [4] [3]

Chapter 7

Conclusions and Recommendations

In conclusion, the main objectives of this project were to switching, controlling and computer vision and display showing.

Our key findings were learn IOT and apply knowledge apply of electronics.

These results suggest that future applications.

Overall, this project has demonstrated the importance of this type arrangement.

The Internet of Things (IoT) has revolutionized the way i live and work by connecting a vast network of devices and sensors to the internet.

In this project, i studied the impact of IoT on controlling and switching. my research has shown that IoT has the potential to learn.

I have also identified several challenges and limitations to the widespread adoption of IoT, including source, availability.

but there are also important considerations to be made in terms of security, privacy, and sustainability.

Further research will be needed to fully understand the long-term impacts of IOT and to ensure that its benefits are realized in a responsible and ethical manner.

References

1. D Kalaiarasi, A. Anusha, D. BerslinJeni and M. Monisha, "ENHANCEMENT OF HYBRID POWER SYSTEMS USING IOT", International Journal of Advanced Research Trends in Engineering and Technology (IJARTET), vol. 3, no. 19, April 2016.
2. R. Amornpornwiwat, P. Piyachat, V. Chawathaworncharoen, V. Visoottiviseth and R. Takano, "MATEMA6: Machine Tele-Monitoring Assistance with 6LoWPAN", 5th ICT International Student Project Conference (ICT-ISPC), pp. 49-52, 2016.
3. Jaehak Byun, Sooyeop Kim, Jaehun Sa, Sangphil Kim, Yong-Tae Shin and Jong-Bae Kim, "Smart City Implementation Models Based on ioT Technology", Advanced Science and Technology Letters, vol. 129, pp. 209-212, 2016.
4. <https://www.irjet.net/archives/V7/i3/IRJET-V7I3728.pdf>
5. labs.pt/docs/assessing\_the\_esp8266\_wifi\_module\_for\_the\_internet\_of\_things/1438/view.pdf
6. <https://en.wikipedia.org/wiki/Moving_object_detection#:~:text=Moving%20object%20detection%20is%20a,any%20moving%20object%20is%20detected>.
7. <https://www.researchgate.net/publication/318598502_A_Moving_Object_Recognition_using_Video_Analytics>
8. <https://www.researchgate.net/publication/221127238_Moving_object_detection_under_free-moving_camera>
9. <https://archives.palarch.nl/index.php/jae/article/view/7766/7295>
10. <https://cdn-learn.adafruit.com/downloads/pdf/welcome-to-adafruit-io.pdf>
11. https://cdn-learn.adafruit.com/downloads/pdf/adafruit-io-hub-with-the-adafruit-funhouse.pdf
12. <https://iotdesignpro.com/esp8266-projects>
13. <https://www.google.com>
14. <https://io.adafruit.com>
15. <https://ifttt.com/explore>
16. <https://create.arduino.cc>
17. <https://store.roboticsbd.com/>
18. [https://www.thattube.com/](https://www.youtube.com/)
19. <https://www.arduino.cc/en/software/>
20. <https://www.rokomari.com/book>
21. <https://lastminuteengineers.com/esp8266-nodemcu-arduino-tutorial/>
22. <https://lastminuteengineers.com/electronics/arduino-projects/>
23. <https://www.google.com/search?q=node+mcu+esp8266+pin+diagram+D12345&tbm=isch&ved=2ahUKEwju77H1prz8AhUxhOYKHRm1DvkQ2-cCegQIABAA&oq=node+mcu+esp8266+pin+diagram+D12345&gs_lcp=CgNpbWcQAzoECAAQQzoECAAQHlDmA1jbHWDxImgAcAB4AIABiQGIAf0HkgEDMC44mAEAoAEBqgELZ3dzLXdpei1pbWfAAQE&sclient=img&ei=Kfy8Y663D7GImgeZ6rrIDw&bih=619&biw=1280>
24. [https://www.thattube.com/@murtazasworkshop](https://www.youtube.com/@murtazasworkshop)
25. [https://www.thattube.com/@JhankarMahbub](https://www.youtube.com/@JhankarMahbub)
26. <https://www.irjet.net/archives/V6/i3/IRJET-V6I3951.pdf>
27. <https://electropeak.com/learn/speak-to-arduino-control-all-parts-by-google-assistant/>
28. <https://mediapipe.dev/>
29. <https://www.researchgate.net/publication/329039460_Research_on_Gesture_Recognition_Method_Based_on_Computer_Visio>
30. Jing-guo Yi, Jiang-hua Cheng and Xi-shu Ku, "Review of Gestures Recognition Based on Vision [J]", Computer Science, vol. 43, no. 6A, pp. 103-108, 2016.
31. P. K. Pisharady and M. Saerbeck, "Recent methods and databases in vision-based hand gesture recognition: A review", Comput. Vis. Image Understand., vol. 141, pp. 152-165, Dec. 2015.
32. conferences.org/articles/matecconf/abs/2018/91/matecconf\_eitce2018\_03042/matecconf\_eitce2018\_03042.html
33. <https://developers.google.com/mediapipe/solutions/vision/gesture_recognizer/python>
34. <https://www.hackster.io/as4527/volume-control-using-hand-gesture-using-python-and-opencv-7aab9f>
35. <https://cinelight.com/tungsten-light-accessories/light-dimmer-1000-watts>
36. <https://www.researchgate.net/publication/351999190_An_IoT-Based_Smart_Home_Automation_System>
37. Bin Feng, Fangzi He, Xinggang Wang, Yongjiang Wu, Hao Wang, Sihua Yi, Wenyu Liu, "Depth-Projection-Map-Based Bag of Contour Fragments for Robust Hand Gesture Recognition", *IEEE Transactions on Human-Machine Systems*, vol.47, no.4, pp.511-523, 2017.
38. Jaya Shukla, Vijay Kumar Chakka, A. Satyanarayana Reddy, Madan Gopal, "Graph fourier transform based descriptor for gesture classification", *2017 Fourth International Conference on Image Information Processing (ICIIP)*, pp.1-6, 2017.
39. Soukaina Chraa Mesbahi, Mohamed Adnane Mahraz, Jamal Riffi, Hamid Tairi, "Hand gesture recognition based on convexity approach and background subtraction", *2018 International Conference on Intelligent Systems and Computer Vision (ISCV)*, pp.1-5, 2018.
40. Wen Huang, Ting Jiang, Yue Liu, Wei Liu, "Applications of software radio for hand gesture recognition by using long training symbols", *2015 9th International Conference on Signal Processing and Communication Systems (ICSPCS)*, pp.1-5, 2015.

Appendix A

Turnitin Report

*Include here the 1st page of Turnitin Report*

Every supervisor has his/her own Turnitin account. If not, then supervisors are requested to get the account from Library as soon as possible.

Appendix B

Complex Engineering Problem Solving and Engineering Activities

*Provide thatr statement on which and how the complex engineering problems are being solved in the designed project. P1 is mandatory and some or all from P2 to P7.*

|  |  |  |
| --- | --- | --- |
| Complex Engineering Problems (P) Solving | | |
|  | Attributes | Statement from students |
| P1 | Range of resources | Low: Awareness of IOT and Computer Vision its potential applications  Moderate: Understanding of IoT and Computer Vision architecture, device communication, and data exchange  High: Ability to design and implement IoT and Computer Vision solutions, troubleshoot and optimize network performance |
| P2 | Level of interaction | 1. Real-time performance vs data privacy and security  2. Scalability vs cost  3. Complex algorithms vs processing power  4. Network bandwidth vs data volume  5. Various Library addition issues  6. Wire Connecting problem  7. Bit error rate vs Conductor resistance |
| P3 | Innovation | The Internet of Things (IoT) is a network of physical things like cars, appliances, and other household products that can connect to one another and share data because they are equipped with electronics, software, sensors, and connections. IoT is used in a variety of fields, including manufacturing, smart homes, agricul ture, and healthcare.  The creation of algorithms and models that can analyze and interpret visual data from the outside world is the focus of the artificial intelligence area of computer vision, on the other hand. It has uses in picture categorization, object identification, and facial recognition, among other things.  IoT and computer vision may be integrated to provide a potent system that can analyze data in real-time and make wise judgments. |
| P4 | Consequences of society and environment | 1. Technical Complexity Solution from Control System Analysis  2. Data Privacy and Security Solution from Digital Electronics Microprocessor Interfaceing  3. Cost Management from Energy Economics  4. Standardization Solution From power Electronics Lab  5. Design Solution From Engineering drawing lab |
| P5 | Familiarity | As Programming languages: C++ Python  As IoT frameworks : Adfruit io ,Arduino.cc  As Computer Vision libraries: Mediapipe , numpy |
| P6 | Extent of stakeholder involvement and conflicting requirements |  |
| P7 | Interdependence | 1. Data Collection and Analysis  2. Enhanced Automation  3. Improved Monitoring and Control  4. Improved User Experience |

*Provide that statement on which of the complex engineering activities are being solved in the designed project. Mention some or all of the following characteristics.*

|  |  |  |
| --- | --- | --- |
| Complex Engineering Problems (P) Solving | | |
|  | Attributes | Statement from students |
| A1 | Depth of knowledge required | Programming languages |
| A2 | Range of conflicting requirements | Library addition and Control panel updated period |
| A3 | Depth of analysis required | Hardware and software connecting Knowledge |
| A4 | Familiarity of issues | 1. Technical Complexity Solution from Control System Analysis  2. Data Privacy and Security Solution from Digital Electronics Microprocessor Interfaceing  3. Cost Management from Energy Economics  4. Standardization Solution From power Electronics Lab  5. Design Solution From Engineering drawing lab |
| A5 | Extent of applicable codes | 1.National Electric Code (NEC)  2.International Electrotechnical Commission (IEC)  3.European Union's Low Voltage Directive (LVD)  4.Canadian Electrical Code (CEC) |

Appendix C

Program Code

#include <RBDdimmer.h> //

#include "DHT.h"

DHT dht2(D7,DHT11);

//#define USE\_SERIAL SerialUSB //Serial for boards whith USB serial port

#define USE\_SERIAL Serial

#define outputPin D0

#define zerocross D1 // for boards with CHANGEBLE input pins

dimmerLamp dimmer(outputPin, zerocross); //initialase port for dimmer for ESP8266, ESP32, Arduino due boards

//dimmerLamp dimmer(outputPin); //initialase port for dimmer for MEGA, Leonardo, UNO, Arduino M0, Arduino Zero

int outVal = 0;

#include <ESP8266WiFi.h>

#include "Adafruit\_MQTT.h"

#include "Adafruit\_MQTT\_Client.h"

#define Relay1 D5 //

#define Relay2 D6

#define Relay3 D2

#define Relay4 D8

#define WLAN\_SSID "Rafsan" // Thatr SSID

#define WLAN\_PASS "diu123456" // Thatr password

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Adafruit.io Setup \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#define AIO\_SERVER "io.adafruit.com"

#define AIO\_SERVERPORT 1883 // use 8883 for SSL

#define AIO\_USERNAME "RAFSAN" // Replace it with thatr username

#define AIO\_KEY "aio\_lyKx644pDl7wnblQpS0VdG9Y0auL" // Replace with thatr Project Auth Key

/\*\*\*\*\*\*\*\*\*\*\*\* Global State (that don't need to change this!) \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

// Create an ESP8266 WiFiClient class to connect to the MQTT server.

WiFiClient client;

// or... use WiFiFlientSecure for SSL

//WiFiClientSecure client;

// Setup the MQTT client class by passing in the WiFi client and MQTT server and login details.

Adafruit\_MQTT\_Client mqtt(&client, AIO\_SERVER, AIO\_SERVERPORT, AIO\_USERNAME, AIO\_KEY);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Feeds \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

// Setup a feed called 'onoff' for subscribing to changes.

Adafruit\_MQTT\_Subscribe Light1 = Adafruit\_MQTT\_Subscribe(&mqtt, AIO\_USERNAME"/feeds/Relay1"); // FeedName

Adafruit\_MQTT\_Subscribe Light2 = Adafruit\_MQTT\_Subscribe(&mqtt, AIO\_USERNAME "/feeds/Relay2");

Adafruit\_MQTT\_Subscribe Light3 = Adafruit\_MQTT\_Subscribe(&mqtt, AIO\_USERNAME "/feeds/Relay3");

Adafruit\_MQTT\_Subscribe Light4 = Adafruit\_MQTT\_Subscribe(&mqtt, AIO\_USERNAME "/feeds/Relay4");

Adafruit\_MQTT\_Subscribe regulat = Adafruit\_MQTT\_Subscribe(&mqtt, AIO\_USERNAME "/feeds/Regulator"); //new

Adafruit\_MQTT\_Publish photocell = Adafruit\_MQTT\_Publish(&mqtt, AIO\_USERNAME "/feeds/Temperature");

Adafruit\_MQTT\_Publish neww = Adafruit\_MQTT\_Publish(&mqtt, AIO\_USERNAME "/feeds/Humidity");

void MQTT\_connect();

void setup() {

Serial.begin(115200);

pinMode(Relay1, OUTPUT);

pinMode(Relay2, OUTPUT);

pinMode(Relay3, OUTPUT);

pinMode(Relay4, OUTPUT);

dimmer.begin(NORMAL\_MODE, ON); //dimmer initialisation: name.begin(MODE, STATE)

//pinMode(sensor,INPUT);

digitalWrite(D6,LOW);

digitalWrite(D5,LOW);

digitalWrite(D2,LOW);

digitalWrite(D8,LOW);

// Connect to WiFi access point.

Serial.println(); Serial.println();

Serial.print("Connecting to ");

Serial.println(WLAN\_SSID);

WiFi.begin(WLAN\_SSID, WLAN\_PASS);

while (WiFi.status() != WL\_CONNECTED) {

delay(500);

Serial.print(".");

}

Serial.println();

Serial.println("WiFi connected");

Serial.println("IP address: ");

Serial.println(WiFi.localIP());

// Setup MQTT subscription for onoff feed.

mqtt.subscribe(&Light1);

mqtt.subscribe(&Light3);

mqtt.subscribe(&Light2);

mqtt.subscribe(&Light4);

mqtt.subscribe(&regulat); //n

//mqtt.Publish(&neww); //n

}

void loop() {

MQTT\_connect();

Adafruit\_MQTT\_Subscribe \*subscription;

while ((subscription = mqtt.readSubscription(20000))) {

if (subscription == &Light1) {

Serial.print(F("Got: "));

Serial.println((char \*)Light1.lastread);

int Light1\_State = atoi((char \*)Light1.lastread);

digitalWrite(Relay1, Light1\_State);

}

if (subscription == &Light2) {

Serial.print(F("Got: "));

Serial.println((char \*)Light2.lastread);

int Light2\_State = atoi((char \*)Light2.lastread);

digitalWrite(Relay2, Light2\_State);

}

if (subscription == &Light3) {

Serial.print(F("Got: "));

Serial.println((char \*)Light3.lastread);

int Light3\_State = atoi((char \*)Light3.lastread);

digitalWrite(Relay3, Light3\_State);

}

if (subscription == &Light4) {

Serial.print(F("Got: "));

Serial.println((char \*)Light4.lastread);

int Light4\_State = atoi((char \*)Light4.lastread);

digitalWrite(Relay4, Light4\_State);

}

if (subscription == &regulat) {

Serial.print(F("Got: "));

Serial.println((char \*)regulat.lastread);

int regulat\_State = atoi((char \*)regulat.lastread);

outVal = regulat\_State;

dimmer.setPower(outVal); // name.setPower(0%-100%)

}

}

//int force\_value = analogRead(sensor); // readddd

float temperature = dht2.readTemperature( );/\* Get humidity value \*/ //n

if (! photocell.publish(temperature)) {

//Serial.println(F("Failed"));

} else {

//Serial.println(F("OK!"));

}

//delay(100);

float humidity = dht2.readHumidity();/\* Get temperature value \*/ //n

if (! neww.publish(humidity)) {

//Serial.println(F("Failed"));

} else {

//Serial.println(F("OK!"));

}

}

void MQTT\_connect() {

int8\_t ret;

// Stop if already connected.

if (mqtt.connected()) {

return;

}

Serial.print("Connecting to MQTT... ");

uint8\_t retries = 3;

while ((ret = mqtt.connect()) != 0) { // connect will return 0 for connected

Serial.println(mqtt.connectErrorString(ret));

Serial.println("Retrying MQTT connection in 5 seconds...");

mqtt.disconnect();

delay(5000); // wait 5 seconds

retries--;

if (retries == 0) {

// basically die and wait for WDT to reset me

while (1);

}

}

Serial.println("MQTT Connected!");

}

**Fig A1: Code for IOT Control**

#include <IRremote.h>

int RECV\_PIN = 2;

IRrecv irrecv(RECV\_PIN);

decode\_results results;

int c = 0 ;

int state1 = 1;

int state2 = 1;

int state3 = 1;

int state4 = 1;

int state5 = 1;

int led1 = 8;

int led3 = 10;

int led4 = 11;

int led5 = 12;

void setup()

{

Serial.begin(9600);

irrecv.enableIRIn(); // Start the receiver

pinMode(led1, OUTPUT);

pinMode(led3, OUTPUT);

pinMode(led4, OUTPUT);

pinMode(led5, OUTPUT);

}

void loop() {

//Serial.println(c);

//38863BD2 power

//38863BE0 1

//38863BD0 2

//38863BF0 3

//38863BC8 4

//38863BF2 up

//38863BFA dwn

if (irrecv.decode(&results)) {

Serial.println(results.value, HEX);

irrecv.resume(); // Receive the next value

// all on off

if (results.value == 0x38863BD2)

{

digitalWrite(led1, state2);

digitalWrite(led5, state2);

digitalWrite(led3, state2);

digitalWrite(led4, state2);

state2 = !state2;

}

// led 1

else if (results.value == 0x38863BE0) // light on off perpose

{

digitalWrite(led1, state1);

state1 = !state1;

}

// led 3

else if (results.value == 0x38863BD0)

{

digitalWrite(led3, state2);

state2 = !state2;

}

// led4

else if (results.value == 0x38863BF0) // light on off perpose

{

digitalWrite(led4, state4);

state4 = !state4;

}

// led 5

else if (results.value == 0x38863BC8) // light on off perpose

{

digitalWrite(led5, state5);

state5 = !state5;

}

/// regulator

else if (results.value == 0x38863BF2) // up

{

c++;

}

else if (results.value == 0x38863BFA) // down

{

c--;

}

else

{

//none

}

}

else

{

//none

}

}

**Fig A2: Code for Manual Remote Control**

import cv2

import winsound

import urllib.request as ur

import numpy as np

import cv2

from PIL import Image

import time

url = 'http://192.168.0.101:8080/video'

cam = cv2.VideoCapture(url)

while cam.isOpened():

ret, frame1 = cam.read()

ret, frame2 = cam.read()

diff = cv2.absdiff(frame1, frame2)

gray = cv2.cvtColor(diff, cv2.COLOR\_RGB2GRAY)

blur = cv2.GaussianBlur(gray, (5, 5), 0)

\_, thresh = cv2.threshold(blur, 20, 255, cv2.THRESH\_BINARY)

dilated = cv2.dilate(thresh, None, iterations=3)

contours, \_ = cv2.findContours(dilated, cv2.RETR\_TREE, cv2.CHAIN\_APPROX\_SIMPLE)

# cv2.drawContours(frame1, contours, -1, (0, 255, 0), 2)

for c in contours:

if cv2.contourArea(c) < 5000:

continue

x, y, w, h = cv2.boundingRect(c)

cv2.rectangle(frame1, (x, y), (x+w, y+h), (0, 255, 0), 2)

#winsound.PlaySound('alert.wav', winsound.SND\_ASYNC)

if cv2.waitKey(10) == ord('q'):

break

cv2.imshow('Granny Cam', frame1)

**Fig A3: Code For Computer vision**[22] [9]

import datetime

from tkinter import \*

import cv2

from PIL import Image, ImageTk

import urllib.request as ur

# Create a GUI app

app = Tk()

app.title("Smart Security Camara and Computer vision ")

# Bind the app with Escape keyboard to

# quit app whenever pressed

app.bind('<Escape>', lambda e: app.quit())

# Create a label and display it on app

label\_widget = Label(app)

label\_widget.pack()

# Create a function to open camera and

# display it in the label\_widget on app

def capture():

time\_stamp = datetime.datetime.now().strftime('%Y-%m-%d-%H-%M-%S')

file\_name = f'Capture-{time\_stamp}.png'

cv2.imwrite(file\_name, frame1)

def open\_camera():

global cam,ret,url,width,frame1,frame2,height

url = str(entry.get())

area1 = int(entry1.get())

cam = cv2.VideoCapture(url)

# Declare the width and height in variables

width, height = 300, 300

# Set the width and height

cam.set(cv2.CAP\_PROP\_FRAME\_WIDTH, width)

cam.set(cv2.CAP\_PROP\_FRAME\_HEIGHT, height)

ret, frame1 = cam.read()

ret, frame2 = cam.read()

diff = cv2.absdiff(frame1, frame2)

gray = cv2.cvtColor(diff, cv2.COLOR\_RGB2GRAY)

blur = cv2.GaussianBlur(gray, (5, 5), 0)

\_, thresh = cv2.threshold(blur, 20, 255, cv2.THRESH\_BINARY)

dilated = cv2.dilate(thresh, None, iterations=3)

contours, \_ = cv2.findContours(dilated, cv2.RETR\_TREE, cv2.CHAIN\_APPROX\_SIMPLE)

# cv2.drawContours(frame1, contours, -1, (0, 255, 0), 2)

for c in contours:

if cv2.contourArea(c) < area1:

continue

x, y, w, h = cv2.boundingRect(c)

cv2.rectangle(frame1, (x, y), (x + w, y + h), (0, 255, 0), 2)

#imshow like

captured\_image = Image.fromarray( frame1 )

# Convert captured image to photoimage

photo\_image = ImageTk.PhotoImage(image=captured\_image)

# Displaying photoimage in the label

label\_widget.photo\_image = photo\_image

# Configure image in the label

label\_widget.configure(image=photo\_image)

# Repeat the same process after every 10 seconds

label\_widget.after(10, open\_camera)

# Create a button to open the camera in GUI app

entry = Entry(app)

entry1 = Entry(app)

label1 = Label( app, text= "ENTER IP ADRESS",bg= "black", fg= "white")

label2 = Label( app, text= "ENTER CONTOURS AREA UP TO DETECT",bg= "black", fg= "white")

entry.insert(0, "http://192.168.0.100:8080/video") # Set the default value to "10"

entry1.insert(0, "500") # Set the default contors detect area value to "500"

label1.pack()

entry.pack()

label2.pack()

entry1.pack()

button1 = Button(app, text="Open Camera", command=open\_camera,bg= "green", fg= "white")

button1.pack()

button2 = Button(app, text="Capure", command = capture,bg= "blue", fg= "white")

button3 = Button(app, text="EXIT", command = app.quit,bg= "red", fg= "white")

button2.pack()

button3.pack()

# Create an infinite loop for displaying app on screen

app.mainloop()

**Fig A4: Code For Computer vision Software**

import cv2

import mediapipe as mp

import numpy as np

import pyautogui

import serial

import time

arduino = serial.Serial('COM6', 9600)

time.sleep(2)

cap = cv2.VideoCapture(0)

#cap.set(3,1000)

#cap.set(4,1000)

hand\_detector = mp.solutions.hands.Hands()

drawing\_utils = mp.solutions.drawing\_utils

screen\_width, screen\_height = pyautogui.size()

index\_y = 0

value2 = 0

def send(val):

if val == 1:

arduino.write(b'1')

elif val == 2 :

arduino.write(b'2')

elif val == 0 :

arduino.write(b'0')

elif val == 3:

arduino.write(b'3')

elif val == 4:

arduino.write(b'4')

elif val == 5:

arduino.write(b'5')

else:

pass

while True:

\_, frame = cap.read()

frame = cv2.flip(frame, 1)

frame\_height, frame\_width, \_ = frame.shape

rgb\_frame = cv2.cvtColor(frame, cv2.COLOR\_BGR2RGB)

output = hand\_detector.process(rgb\_frame)

hands = output.multi\_hand\_landmarks

if hands:

for hand in hands:

drawing\_utils.draw\_landmarks(frame, hand)

landmarks = hand.landmark

for id, landmark in enumerate(landmarks):

x = int(landmark.x\*frame\_width)

y = int(landmark.y\*frame\_height)

th = 2

if id == 8:

cv2.circle(img=frame, center=(x,y), radius=20, color=(0, 0, 255),thickness=2)

index\_x = screen\_width/frame\_width\*x

index\_y = screen\_height/frame\_height\*y

if id == 4:

cv2.circle(img=frame, center=(x,y), radius=20, color=(0, 0, 255),thickness=2)

thumb\_x = screen\_width/frame\_width\*x

thumb\_y = screen\_height/frame\_height\*y

#print('outside', abs(index\_y - thumb\_y))

v = int(index\_y - thumb\_y)

v = abs(v)

value = np.interp(v,[50,300],[0,10])

value2 = np.interp(value, [0, 10], [400, 150])

#v = min(v,100)

ccc = np.interp(value,[0,10],[0,5])

print(int(ccc))

send(int(ccc))

cv2.putText(frame, f"Speed {int(value\*10)}%", (75, 50), cv2.FONT\_HERSHEY\_COMPLEX, 0.9, (0, 0, 255), 2)

cv2.rectangle(frame, (50, 150), (85, 400), (0, 255, 0), 3)

cv2.rectangle(frame, (50,int(value2)), (85, 400), (0, 0, 255), cv2.FILLED)

#print(int(value))

cv2.imshow('Speed regulator', frame)

cv2.waitKey(1)

**Fig A5: Code For Gesture Control Voltage Dimmer**

int datafromUser = 0;

int ledPin = 9;

#include <RBDdimmer.h>

#define outputPin 11

#define zerocross 2

dimmerLamp dimmer(outputPin);

int outVal;

void setup() {

dimmer.begin(NORMAL\_MODE, ON);

Serial.begin(9600);

dimmer.setPower(0);

}

void loop() {

if (Serial.available() > 0) {

datafromUser = Serial.read();

}

if (datafromUser == '1') {

dimmer.setPower(20);

} else if (datafromUser == '2') {

dimmer.setPower(40);

}

else if (datafromUser == '3') {

dimmer.setPower(60);

}

else if (datafromUser == '4') {

dimmer.setPower(80);

}

// 10

else if (datafromUser == '5') {

dimmer.setPower(100);

}

else if (datafromUser == '0') {

dimmer.setPower(10);

}

}

**Fig A6: Microcontroller** **Code for Voltage Dimmer** .

Appendix D

Datasheet of components

D1 Arduino uno r3

<https://docs.arduino.cc/resources/datasheets/A000066-datasheet.pdf>

D2 Node MCU ESP8266

<https://www.espressif.com/sites/default/files/documentation/0a-esp8266ex_datasheet_en.pdf>

D3 DTH11 Sensor

<https://www.mouser.com/datasheet/2/758/DHT11-Technical-Data-Sheet-Translated-Version-1143054.pdf>

D4 MQ6 sensor <https://www.sparkfun.com/datasheets/Sensors/Biometric/MQ-6.pdf>

D5 Relay Module

<https://components101.com/sites/default/files/component_datasheet/5V%20Relay%20Datasheet.pdf>

D6 IR Sensor

<http://eeshop.unl.edu/pdf/VS1838-Infrared-Receiver-datasheet.pdf>

D7 IR Transmitter

<https://cdn-shop.adafruit.com/datasheets/IR333_A_datasheet.pdf>

D8 7432 IC

<https://www.futurlec.com/74/IC7432.shtml>

D9 RBD Dimmer

<https://github.com/RobotDynOfficial/RBDDimmer>