

# **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 Degree  
of Bachelor of Science in Electrical and Electronic Engineering award.

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



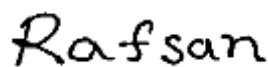
Department of Electrical and Electronic Engineering  
Faculty of Engineering  
DAFFODIL INTERNATIONAL UNIVERSITY

MAY, 2023

## DECLARATION

I hereby declare that this project “Internet of Things based Smart Switching and Controlling Module with Computer Vision.” represents my 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: Md. Hasemi Rafsan Jani Shohan

ID: 191-33-849

## **APPROVAL**

The project is entitled “Internet of Things based Smart Switching and Controlling Module with Computer Vision.” Was 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 degree of Bachelor of Science in Electrical and Electronic Engineering requirements in May, 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 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 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 the development of this project.

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

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

# TABLE OF CONTENTS

<b>LIST OF FIGURES</b>	<b>vii</b>
<b>LIST OF TABLES</b>	<b>ix</b>
<b>LIST OF ABBREVIATIONS</b>	<b>x</b>
<b>LIST OF SYMBOLS</b>	<b>xi</b>
<b>ACKNOWLEDGMENT</b>	
<b>ABSTRACT</b>	
 <b>CHAPTER 1: INTRODUCTION</b>	
1.1 Background / Motivation / Introduction	1
1.2 Problem Statement and /or Proposed Solution (s)	2
1.3 Aims/ Objectives	2
1.4 Brief Methodology / Technologies/ Procedures	3
1.5 Implementation Schedule / Gantt Chart	4
1.6 Organization / Structure of the Report	4
 <b>CHAPTER 2: LITERATURE REVIEW</b>	
2.1 Introduction	16
2.2 Related Research/ Works	16
2.3 Compare and Contrast	18
2.4 Summary	18
 <b>CHAPTER 3: MATERIALS AND METHODS [OR PROJECT/SYSTEM DESIGN] [OR DESIGN PROCEDURE]</b>	
3.1 Introduction	19
3.2 Methods and Materials or System Design and Components	20
3.3 Design Specifications. Standards and Constraints	29
3.4 System Analysis or Design Analysis	30
3.5 Simulation/Experimental Setup [or Implementation]	35
3.6 Summary	37
 <b>CHAPTER 4: RESULTS AND DISCUSSIONS</b>	40
4.1 Results/ Observations/	38-47
4.2 Discussions	47
 <b>CHAPTER 5: PROJECT MANAGEMENT</b>	
5.1 Task, Schedule, and Milestones	48
5.2 Resources and Cost Management	49
5.3 Lesson Learned	49
 <b>CHAPTER 6: IMPACT ASSESSMENT OF THE PROJECT</b>	

6.1 Economical, Societal, and Global Impact	50
6.2 Environmental and Ethical Issues	50
6.3 Utilization of Existing Standards or Codes	51
6.4 Other Concerns	51
<b>CHAPTER 7: CONCLUSIONS AND RECOMMENDATIONS</b>	<b>52</b>
7.1 Conclusions	52
7.2 New Skills and Experiences Learned	52
7.3 Future Recommendations	52
<b>REFERENCES</b>	<b>53-55</b>
<b>APPENDIX A: TURNITIN REPORT</b>	
<b>APPENDIX B: COMPLEX ENGINEERING PROBLEM SOLVING AND ENGINEERING ACTIVITIES</b>	<b>57-59</b>
<b>APPENDIX C: PROGRAM CODE</b>	<b>60-72</b>
<b>APPENDIX D: DATASHEET OF COMPONENTS</b>	<b>73</b>

## LIST OF FIGURES

Fig 1.1	Logic Control Block Diagram of this project	1
Fig 1.2	Worldwide Control Block Diagram of this project	2
Fig 1.3	Control Block of this project	3
Fig 1.4	Implementation Schedule step Block Diagram of this project	4
Fig 1.5	Flow Steps Block	6
Fig 1.6	Switching Control Block	6
Fig 1.7	Voltage Dimmer Control Block	7
Fig 1.8	Gas Sensor Value Showing Block	7
Fig 1.9	Temperature and Humidity Showing Block	8
Fig 1.11	Computer Vision Object Movement Showing frame	9
Fig 1.12	Computer Vision Object Movement Showing frame2	9
Fig 1.13	Self-Developed Computer Vision Software Object Movement initial frame	10
Fig 1.14	After clicking the “open camera” button Computer Vision Movement frame is.	11
Fig 1.15	After clicking the “capture” but saving he turned the age	11
Fig 1.16	Gesture Control phase Voltage Triggering point view	12
Fig 1.17	Gesture Control High-Level gap between index and thumb finger.	13
Fig 1.18	Gesture Control Medium Level gap between index and thumb finger.	13
Fig 1.19	Gesture Control Low-Level gap between index and thumb finger	14
Fig 1.20	Gesture Control Zero Level gap between index and thumb finger.	14
Fig 1.21	After Gesture Control Zero Level to High-Level Voltage amplitude and brightness fluctuate Result and Component Circuit Diagram	15
Fig 3.1	IoT Control structure diagram of this project	19
Fig 3.2	Component visual Circuit diagram of this project	21
Fig 3.3	Node MCU esp8266	22
Fig 3.4	IR Receiver	22
Fig 3.6	Arduino uno r3	23
Fig 3.7	Bulb as a Load	24
Fig 3.8	RBD Voltage Dimmer	25
Fig 3.9	DTH11 Sensor	25
Fig 3.10	The MQ2 Gas Sensor	26
Fig 3.11	5v four-channel Relay Module	27
Fig 3.12	Quard OR gate IC 7432	27
Fig 3.13	Schematic arrangement for Node mcu esp8266	28
Fig 3.14	Schematic arrangement of Arduino Uno	28
Fig 3.15	Block diagram of Computer vision processing.	29
Fig 3.16	Developing Block steps For Computer vision Software	29
Fig 3.17	IOT platform Creating block step n fruit server.	30

Fig 3.18	IoT platform Creating step 1	31
Fig 3.19	IoT platform Creating step 2	31
Fig 3.20	IoT platform Creating step 3	32
Fig 3.21	IoT platform Creating step 3	32
Fig 3.22	IoT platform Creating step 4	33
Fig 3.23	IoT platform Creating step 5	33
Fig 3.24	IoT platform Creating step 6	34
Fig 3.25	21 Hands Landmarks point	<b>36</b>
Fig 3.26	27 Hands Landmarks point Specification	37
Fig 3.27	Gesture Control Steps 1	37
Fig 3.28	Gesture Control Steps 2	37
Fig 3.29	<b>Block steps of Gesture Control Development</b>	37
Fig 4.1	Flow Diagram	40
Fig 4.2	IFTTT Service and Appleates1	40
Fig 4.3	IFTTT Service and Appleates2	41
Fig 4.4	IFTTT Service and Appleates3	42
Fig 4.5	Google-based Service of this project	42
Fig 4.6	Android-based Service of this project	43
Fig 4.7	Electronic Mail-based Service of this project	43
Fig 4.8	Cellular-based Service of this project	43
Fig 4.9	Social Media based Service of this project	44
Fig 4.10	Self-developed apps for use in projects	44
Fig 4.11	Computer vision or video visualization protocols	44
Fig 4.12	IP link <a href="http://192.168.0.103:8080/">http://192.168.0.103:8080/</a> based computer vision or video visualization through the Internet.	45
Fig 4.13	IP link <a href="http://192.168.0.103:8080/">http://192.168.0.103:8080/</a> and contours area selection based on computer vision or video visualization software through the Internet.	45
Fig 4.14	Index and thumb Finger Gesture Movement based phase angle control part.	46



## LIST OF TABLES

<i>Table No</i>	<i>Fig. Name</i>	<i>Page No.</i>
Table 1.1	Figure Table	vii
Table 2.1	Project Cost Management	46

## **LIST OF ABBREVIATIONS**

IFTTT	If This Then That
IOT	Internet of Things
CV2	Computer Vision
IR	Infrared Receiver
MQ2	LPG Gas Sensor
GPIO	General Input output Purpose

## LIST OF SYMBOLS

<i>Symbol</i>	<i>Name of the symbol</i>
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. A voltage dimmer, temperature sensor, and gas sensor were also 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. 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 developed 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.

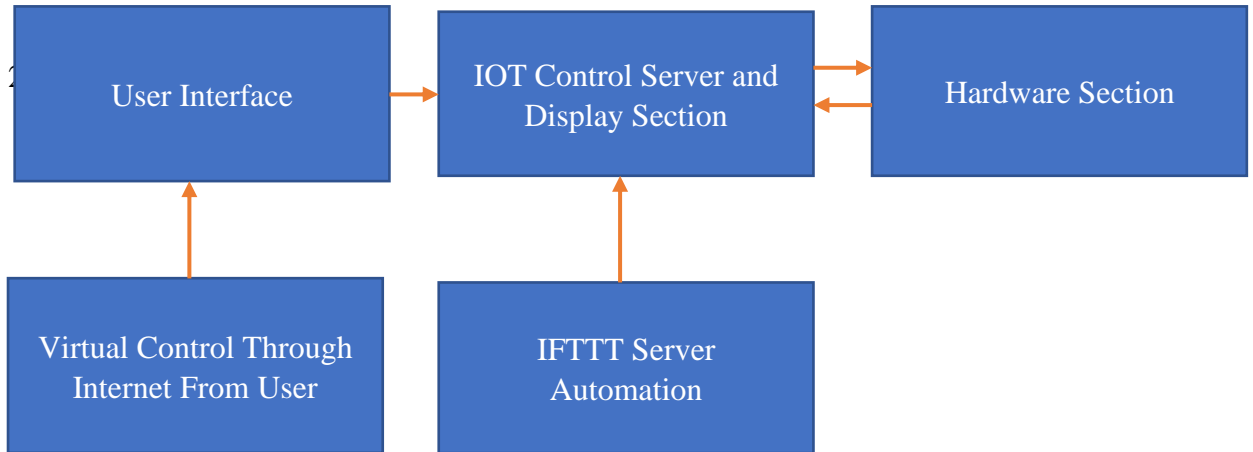
The default settings for Control Area and IP Link. Plans for the project's image capture

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 to create an Android IOT control app.

# CHAPTER 1

## 1.1 INTRODUCTION



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

The world heading towards the fourth industrial revolution. This is This project in continuation of that. In this project, I have included several factors that are playing an important role in the Fourth Industrial Revolution. This project IOT based Switching and controlling Module with Computer Vision.

By this, the analog switching method is changed and converted to IOT. It enables switching from anywhere in the world. And also connected the 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.

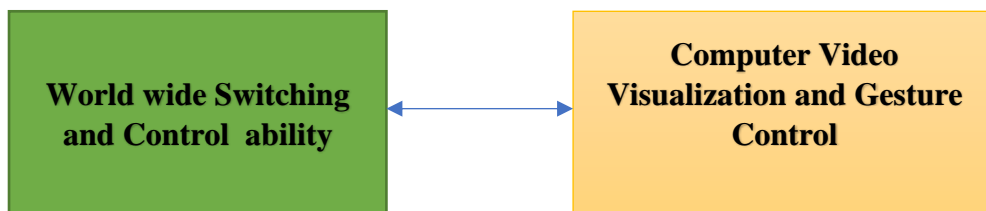
I have developed computer vision software for this project. It enables video visualization based on IP links. It enables movement detection of objects at a specific location over the world.

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 the project

I hope that my ideas from this project will contribute to developing the updated electrical pro shortly. 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 .

## 1.2 Problem Statement and /or Proposed solution(s)



**Fig 1.2 Worldwide Control Block Diagram of this project**

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 are 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 users 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.

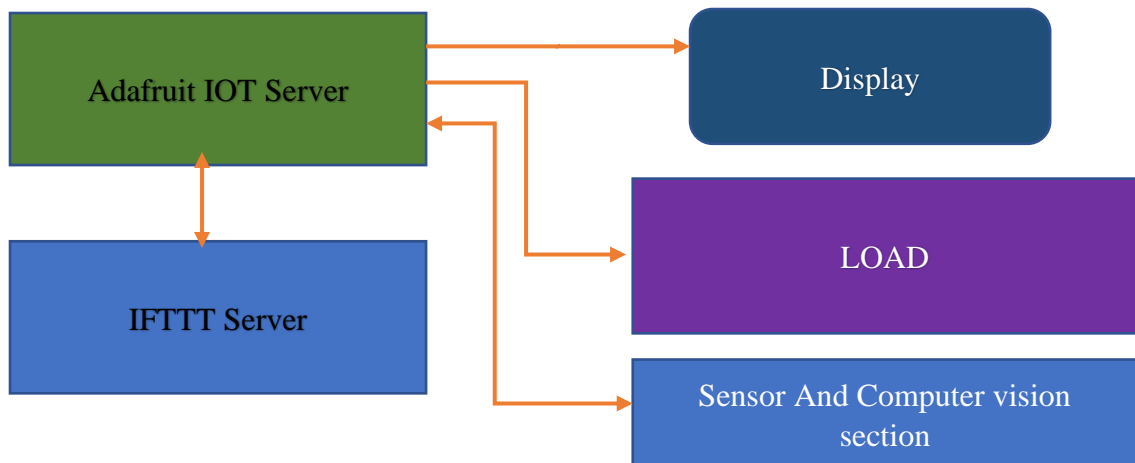
To develop data analytics and visualization tools and integrate them into the platform. To measure the success of the platform through key performance indicators such as user satisfaction, security, and ease of use.

The project aim, object and 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

### 1.3 Brief Methodology / Technologies/ Procedures

#### 1.4 Working Principle



**Fig 1.3 Control Block of this project**

The project is mainly based on the Internet of Things. I used to 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 dht11 sensor as a temperature sensor and methane gas sensor as a gas sensor.

And used the RBD Voltage Dimmer module as an analog voltage dimmer.

Used adafruit.io website server as virtual control platform.

I used a web server called IFTTT for control.

And used four relay modules.

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 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 control 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 streaming all the data of the visualization through the internet in a link-based sense.

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

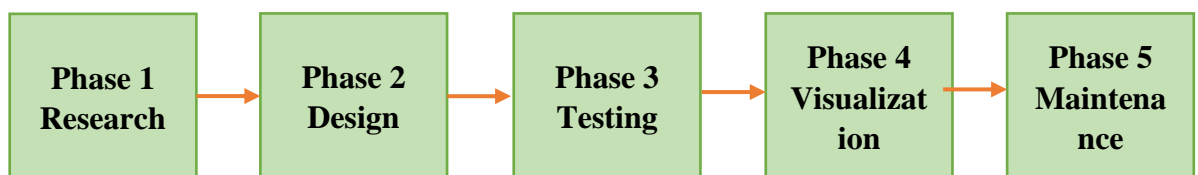
This project I barbarically the option of any type of object and can detect movement.

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

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

Where there is a facility for recording aa a and faculty of viewing three internets.

### 1.5 Implementation Schedule



**Fig 1.4 Implementation Schedule step Block Diagram of this project**

#### **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 (Weeks 5-12)**

The design user interface for the central management platform



Develop platform architecture and integrate security features

Develop computer vision algorithms and integrate them with the platform

Test platform with a small group of users for feedback

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

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

Make final adjustments based on user feedback

Prepare for the 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

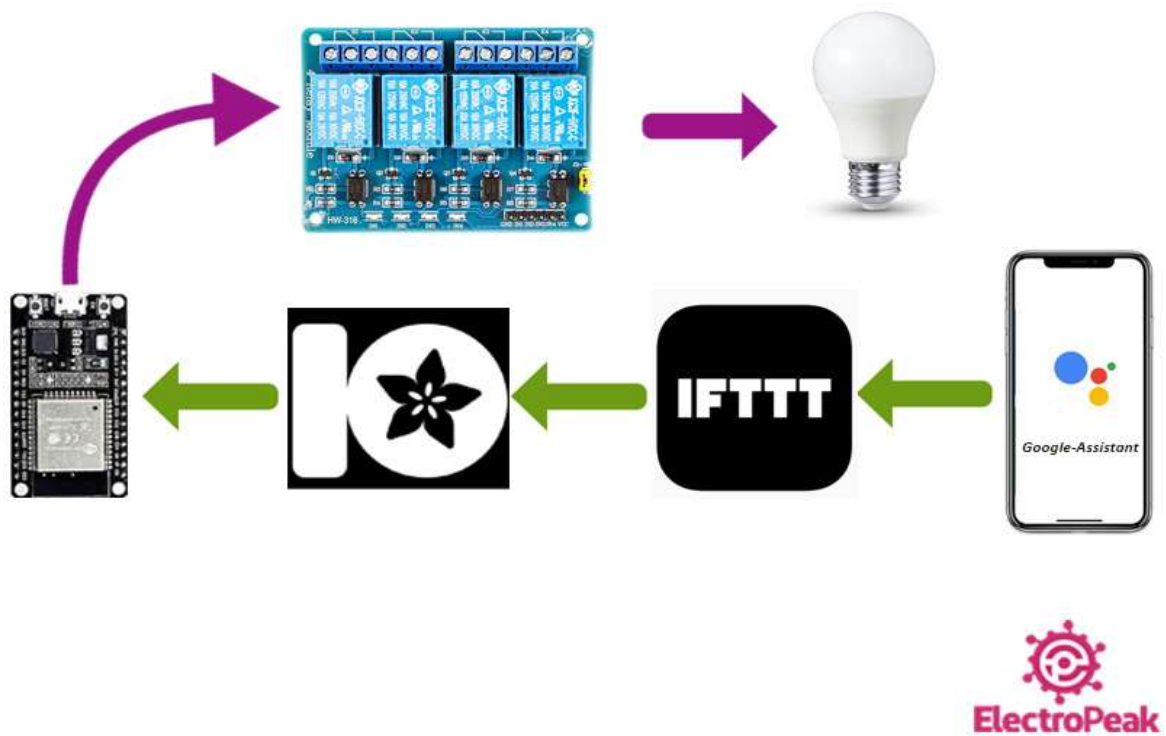
### **Phase 5: Launch and Maintenance (Week 21-24)**

Launch the platform to the public

Monitor and maintain the platform

Continuously gather user feedback and make improvements as necessary

## 2.1 Introduction



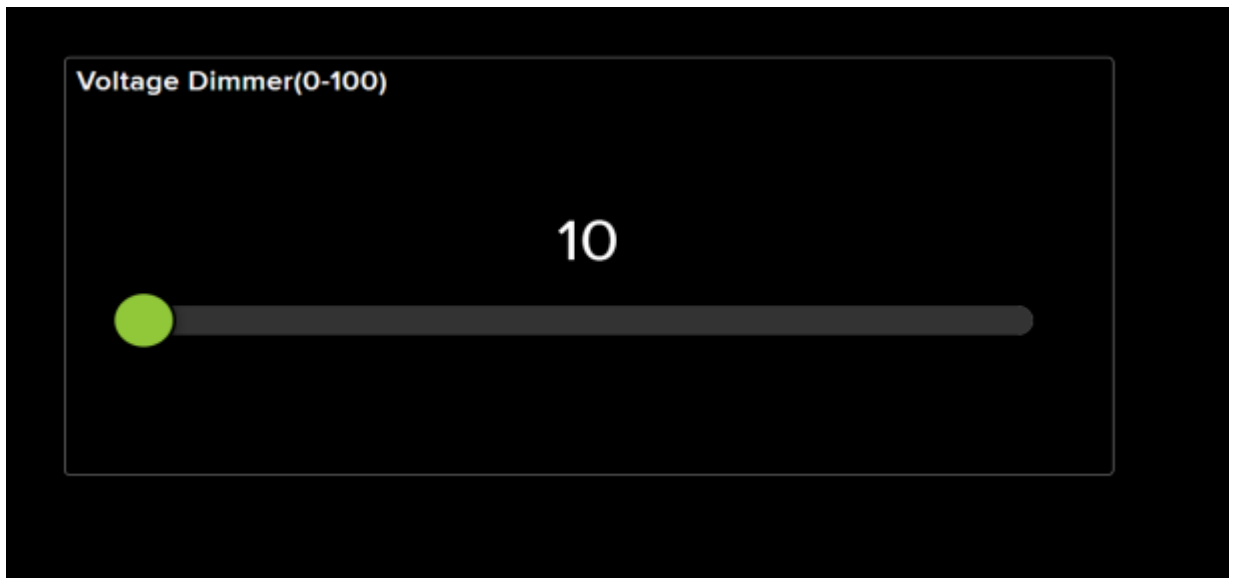
**Fig 1.5 Flow Steps Block[27]**

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 1.6 Switching Control Block**

This Figure shows the switching block diagram. This is expressed as relation Relay 1, Relay 2, Relay 3, and RRelayIOT-based based switching tasks can be done by virtually pressthesethis buttons.



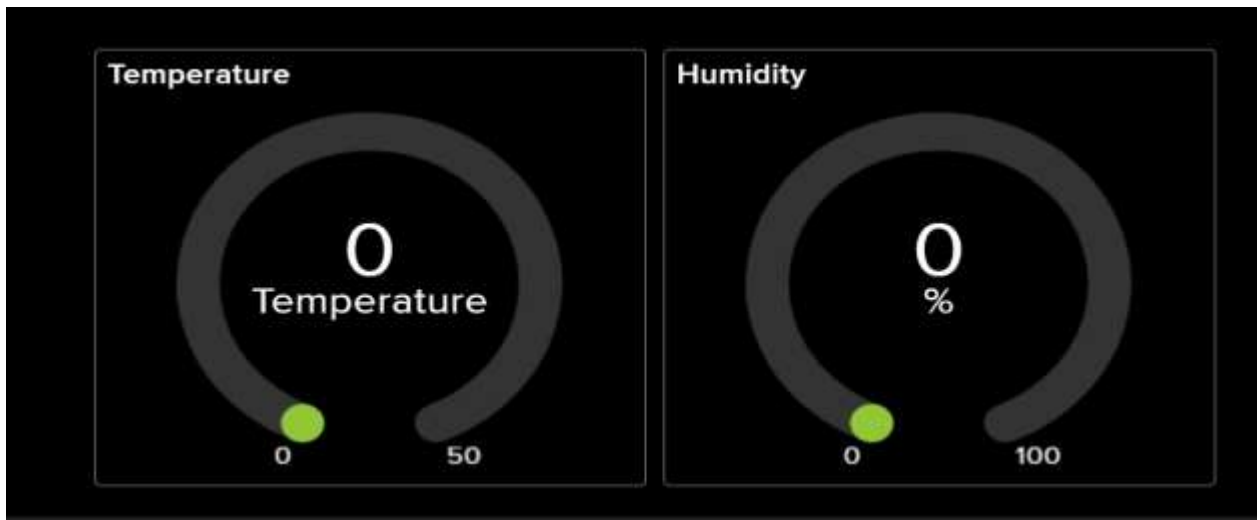
**Fig 1.7 Voltage Dimmer Control Block**

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.



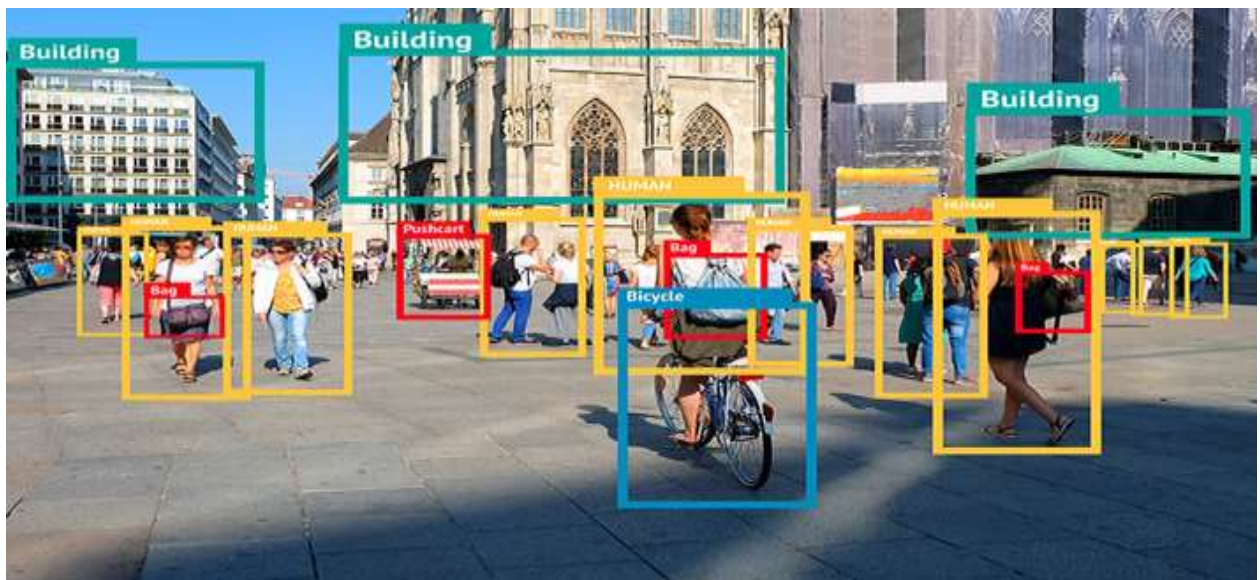
**Fig 1.8 Gas Sensor Value Showing Block**

This Figure shows the number 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.

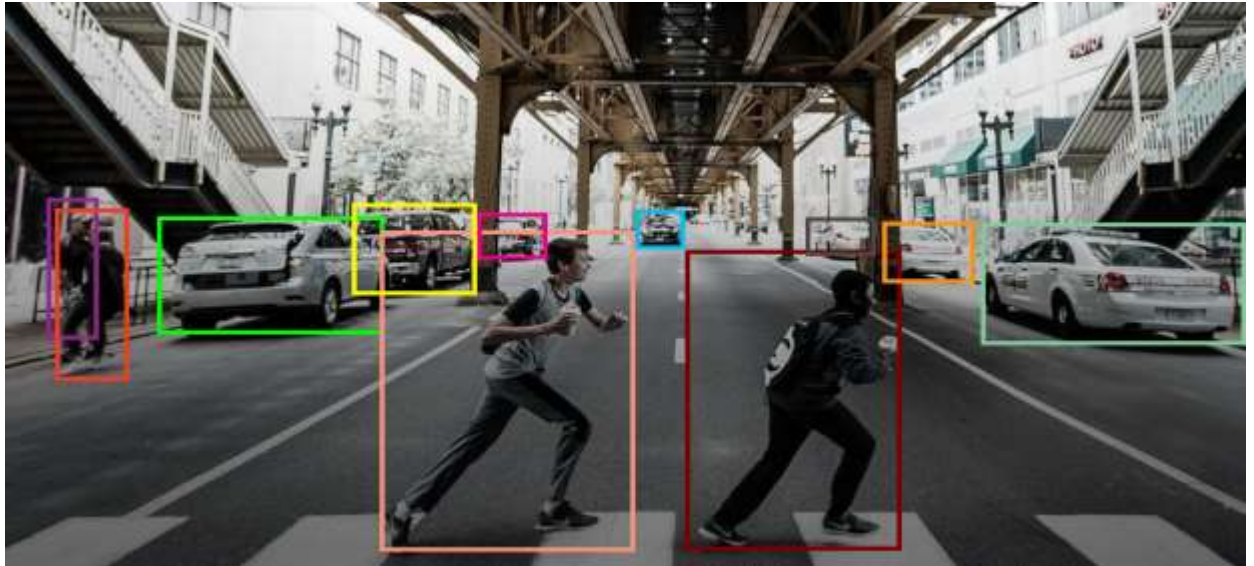


**Fig 1.9 Temperature and Humidity Showing Block**

The values of the temperature and humidity sensors are shown with this figure. The temperature and humidity values can be seen with its help.



**Fig 1.11 Computer Vision Object Movement Showing frame**



**Fig 1.12 Computer Vision Object Movement Showing frame2**

Computer vision is shown in the figure in the ofthishiss 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 1.13 Self-Developed Computer Vision Software Object Movement initial frame**

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

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

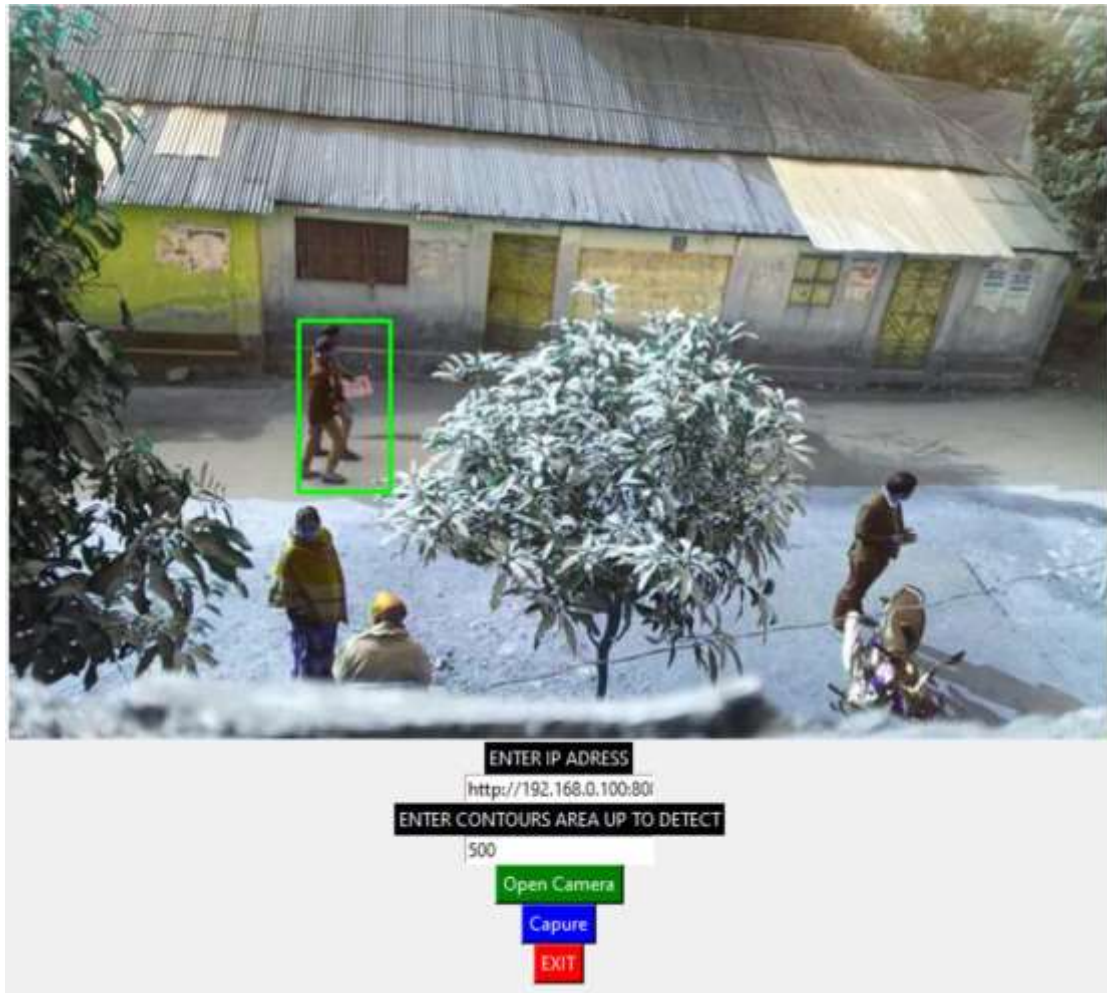
Computer vision, on the other hand, is a field of artificial intelligence that focuses on the development of algorithms and models that enable machines to interpret and understand visual information. This project is used in a variety of applications, such as image and video recognition, object detection,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



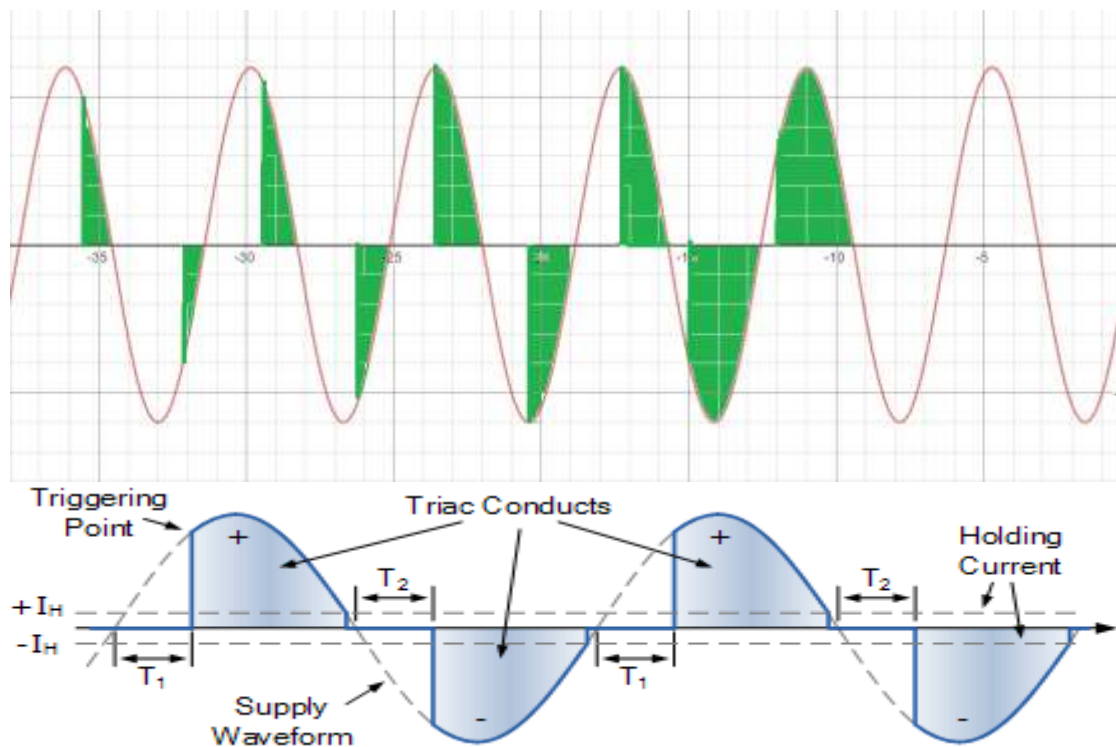


**Fig 1.14** After clicking the “open camera” button Computer Vision Movement frame.



**Fig 1.15** After clicking the “capture” but saving he turned the age

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

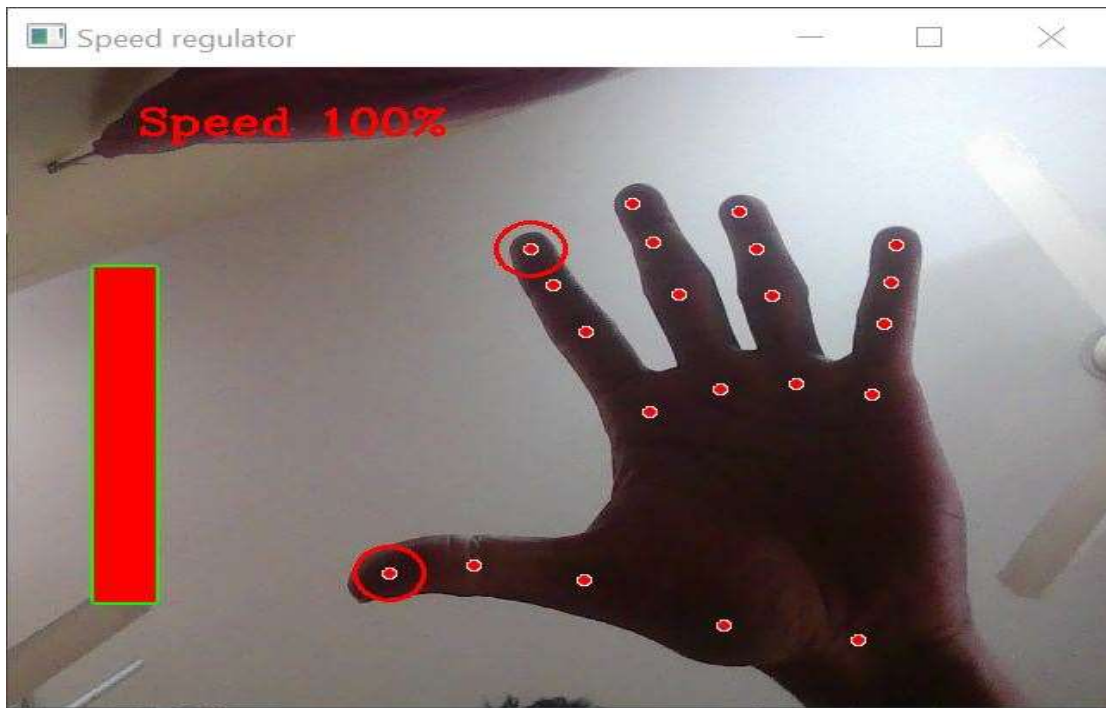


**Fig 1.16 Gesture Control phase Voltage Triggering point view.[35][34]**

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. 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 are 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 1.17 Gesture Control High-Level gap between index and thumb finger.**

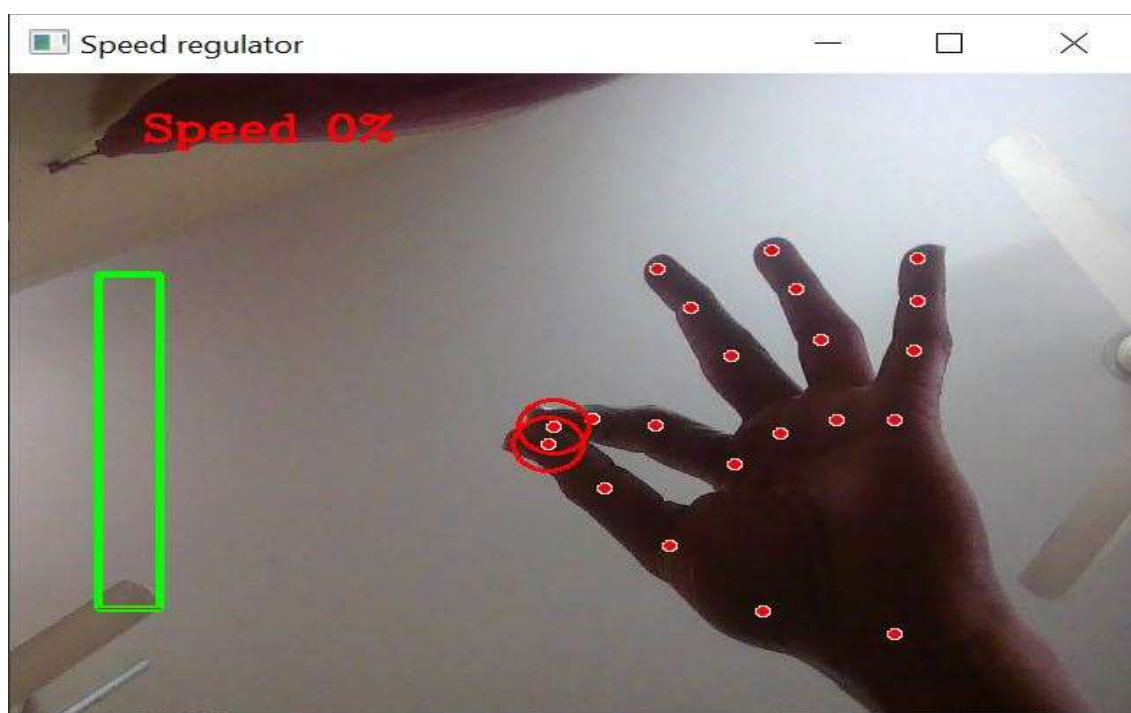
I'll be utilizing this library's hand module to make two great projects in this article. The hhand's module generates a localization of my hand based on 21 points. This means that if this project provides this module with an image e of sha owing a hand, it will return a 21-point vector with the coordinates of 21 significant landmarks that arloathehe shisha the swinging and[28][29][30][31][32][33][34]



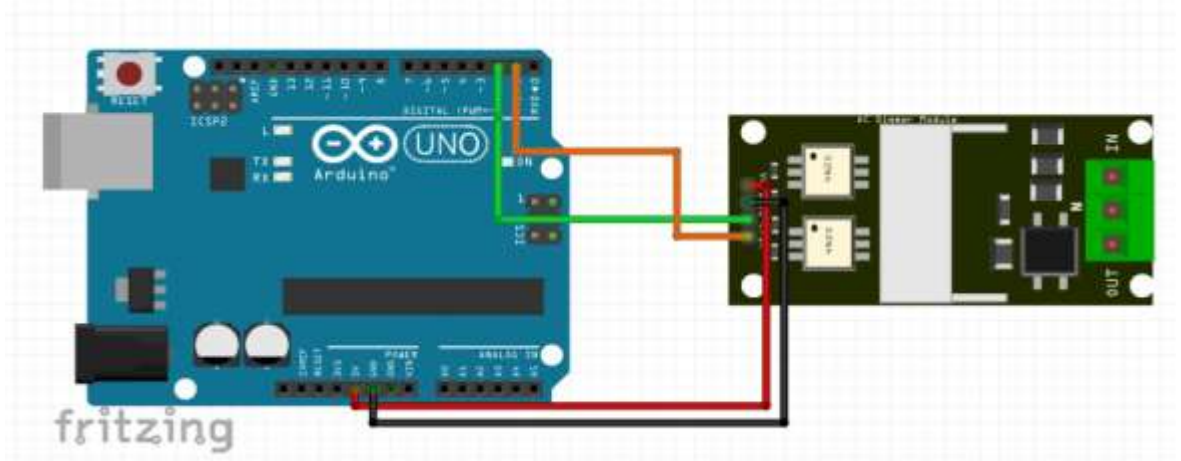
**Fig 1.18 Gesture Control Medium Level gap between index and thumb finger.**



**Fig 1.19 Gesture Control Low-Level gap between index and thumb finger**



**Fig 1.20 Gesture Control Zero Level gap between index and thumb finger.**



**Fig 1.21 After Gesture Control Zero Level to High-Level Voltage amplitude and brightness fluctuate Result and Component Circuit Diagram.[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 Measures the distance between the tips of Showing thumb and index finger. Map the distance between the tips of Showingr'sr 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 automation. 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 Personal Computer 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, and user-friendly and may be further improved by several 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. 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 emphasizes 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 developed and designed a system that can identify motions in front of a web camera in real time. To capture moving portions in the picture, the backdrop from collected frames is first eliminated using a Gaussian mixture-based background/foreground segmentation technique. 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 control 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 developed.

**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 control 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 control 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 control 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 such 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, 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.

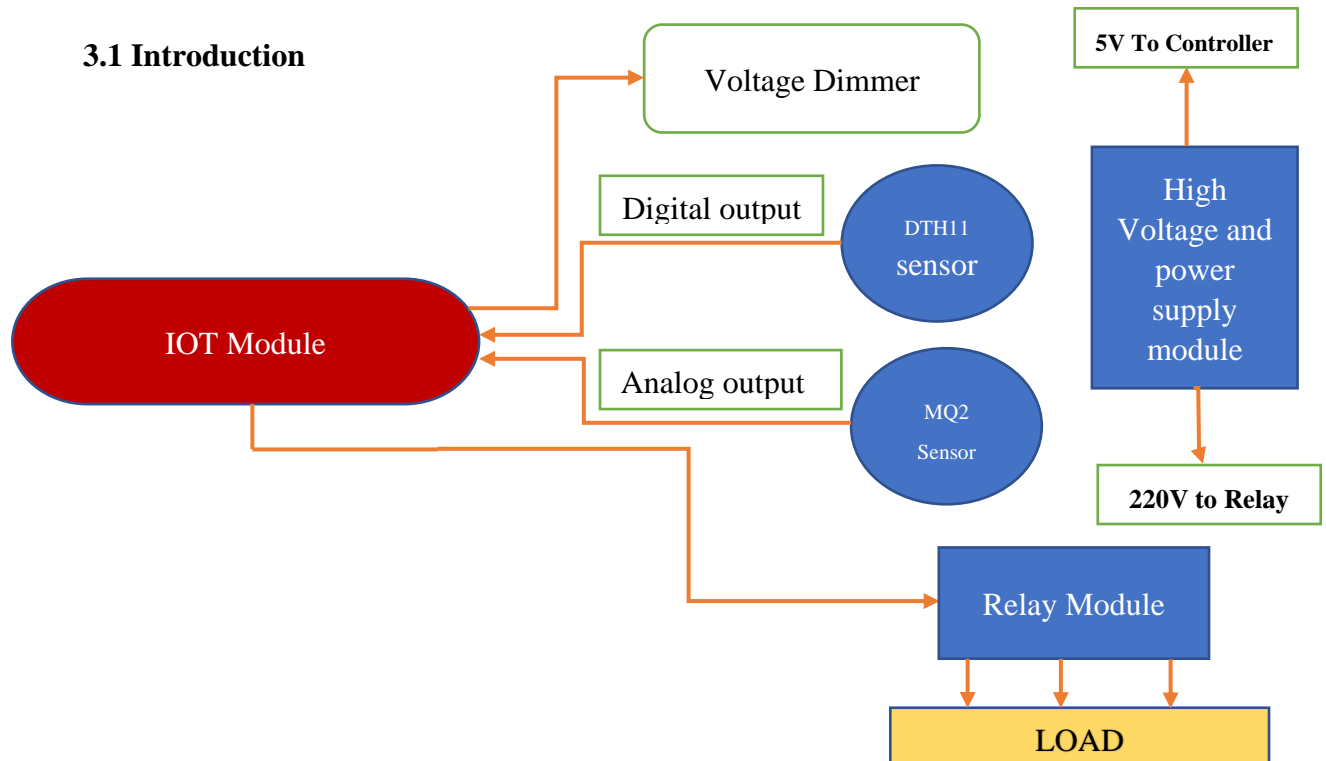
**Resource-Constrained Devices:** Research in this area has focused heavily on the integration of computer vision-based gesture recognition with resource-constrained devices, such as wearable technology and mobile phones. This poses hurdles in terms of computational and energy limits. [37] [38] [39]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

## 2.4 Summary

IoT and computer vision technologies working together have the potential to change several 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]



**Fig 3.1 IOT 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**

#### **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

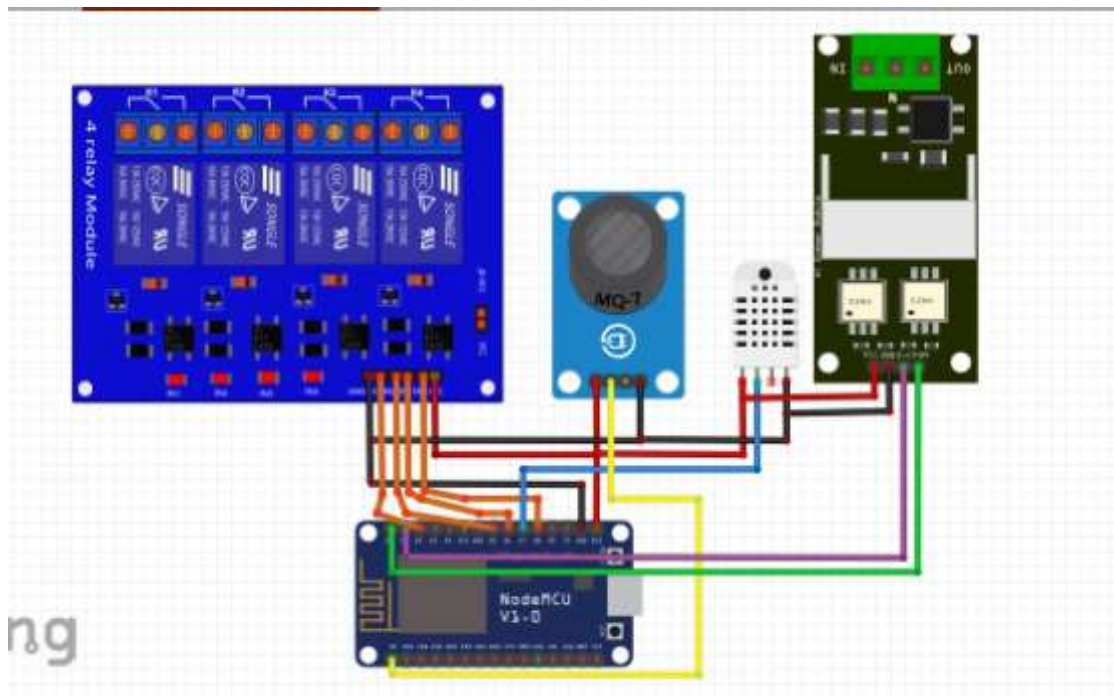
Image processing and analysis software

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 visual Circuit diagram of this project**

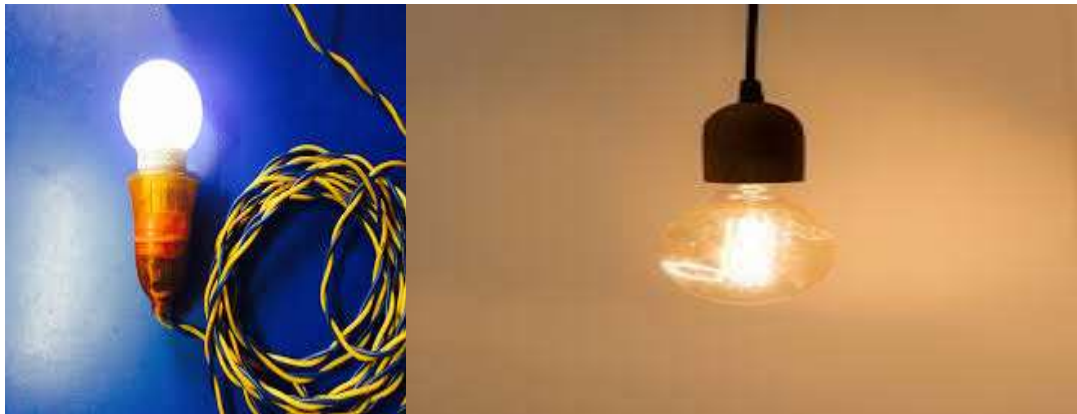
Arduino Uno and Node MCU ESP8266 are used as microcontrollers in this project.

Arduino Uno is used as a simple microcontroller and Node MCU ESP8266 is used to take advantage of WiFi connectivity.



**Fig 3.6 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]



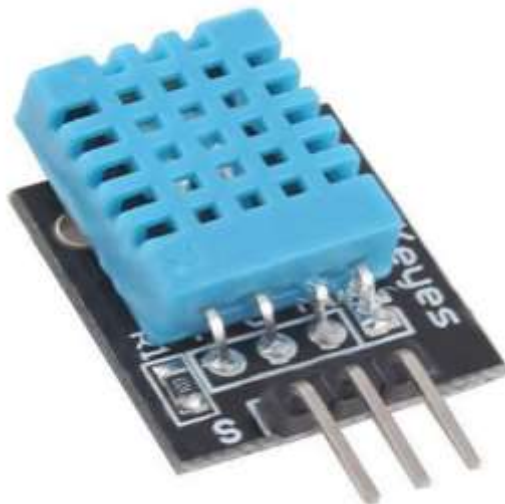
**Fig 3.7 Bulb as a Load**

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 their 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 3.8 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 circuits and trial-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 3.9 DTH11 Sensor**

Temperature and humidity are sensed with a DHT11 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-Thatrself 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 Personal Computer 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 3.10 The MQ2 Gas Sensor**

Gas sensing is done with the MQ2 sensor. essentially runs on five volts. gives the system information by way of analog output. Additionally, there is a variable node system on the sensitivity increase or decrease.

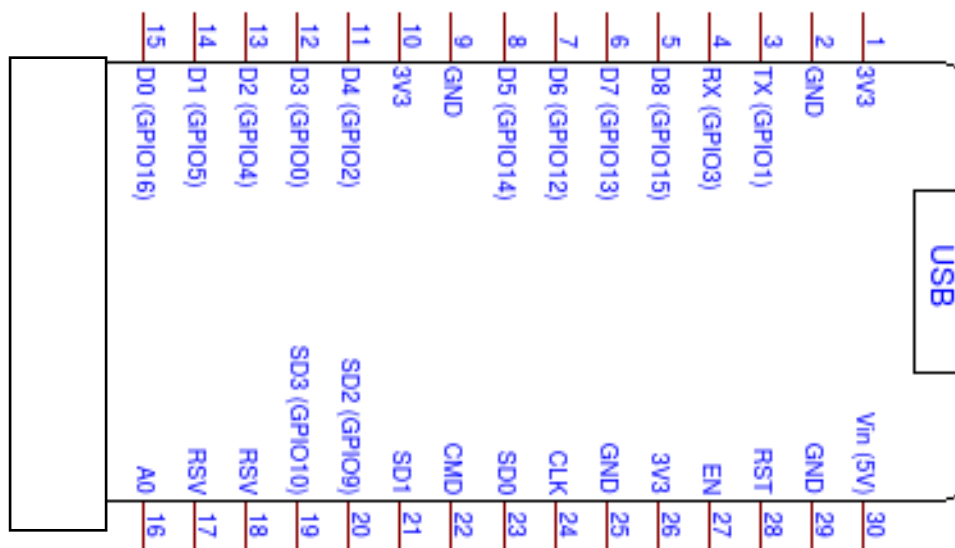
Gas sensors like the MQ2 are frequently used to find combustible gases including propane, butane, and methane. It may be purchased as a single sensor or as a module with a 4-pin connection and a microcontroller. A metal oxide semiconductor (MOS) element's electrical resistance is measured by the MQ2 sensor 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 MQ2 sensor can be interfaced with a microcontroller or a single-board computer such as an Arduino or Personal Computer 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 3.11 5v four channel Relay Module**

A 4-channel relay is used as a switching device in this project. BOperateson five volts. Here a negative input of 5 volts has to be given cooperating too to the operating 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]



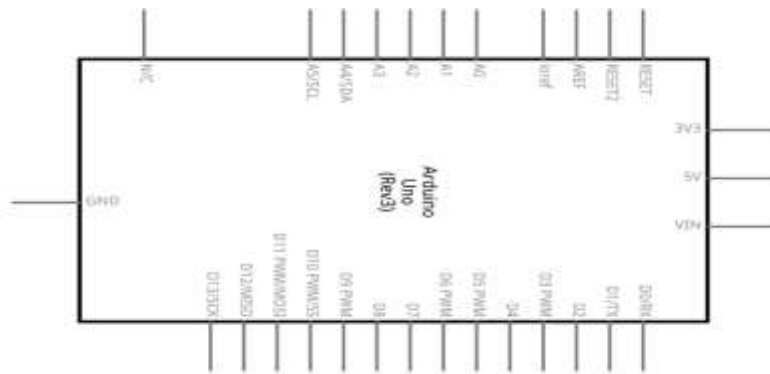
**Fig 3.13 Scametric arrangement for Node mcu esp8266**

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.

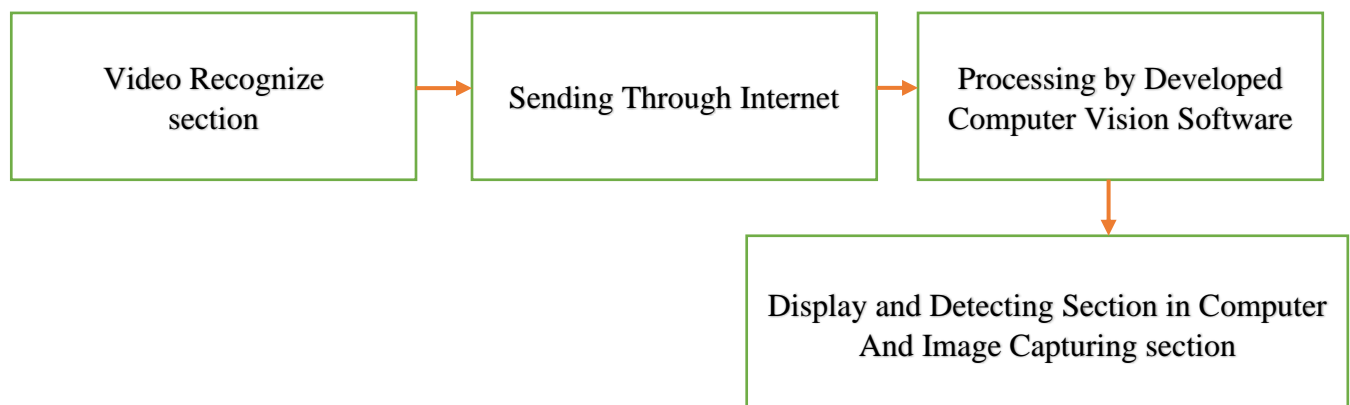
Relay control is accomplished using pins D5, D6, D2, and D8. [D2]



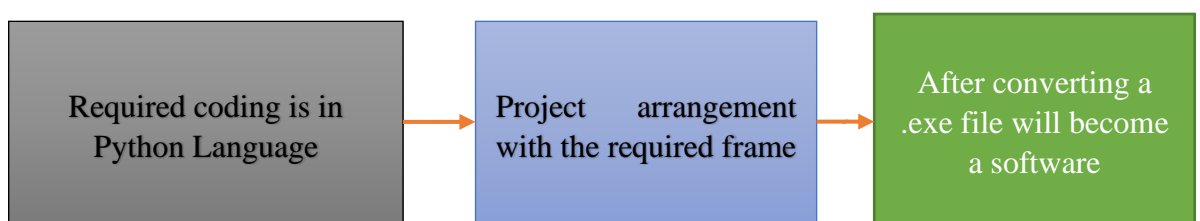
**3.14 Scametric arrangement of Arduino Uno**

Arduino Uno is used as a microcontroller for Computer vision control. With the microcontroller, the digi2-numbermber pin is used as a gesture control section. Digital Pi issued for control-based relaybasedrelay controlling. [D1]

### 3.3 Design Specifications. Standards and Constraints



**fig 3.15 Block diagram of Computer vision processing.**



**fig 3.16 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.

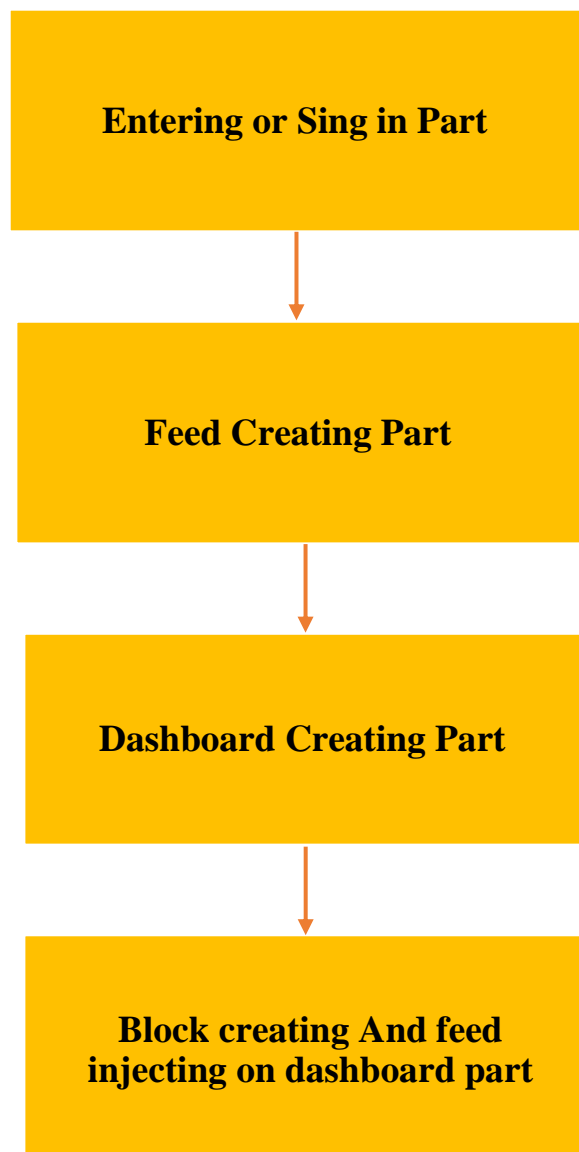
The system has to the are requirements for deep learning frameworks like TensorFlow and PyTorch.

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]





**fig 3.17 IOT platform Creating block step n fruit server.**

The image shows the Adafruit website's sign-in interface. At the top is the Adafruit logo. Below it, the text "SIGN IN" is displayed. A message states: "Your Adafruit account grants you access to all of Adafruit, including the shop, learning system, and forums." A light blue notification bar indicates "Signed out successfully." with a close button. Below this are two input fields: "EMAIL OR USERNAME" containing the text "Rafsan" and "PASSWORD" containing masked characters. A link "Forget your password?" is next to the password field. At the bottom is a blue "SIGN IN" button.

**fig 3.18 IOT platform Creating step 1**

The first step in creating an IOT control block is to complete the signing part. A 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.

## RAFSAN / Feeds



**fig 3.19 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.

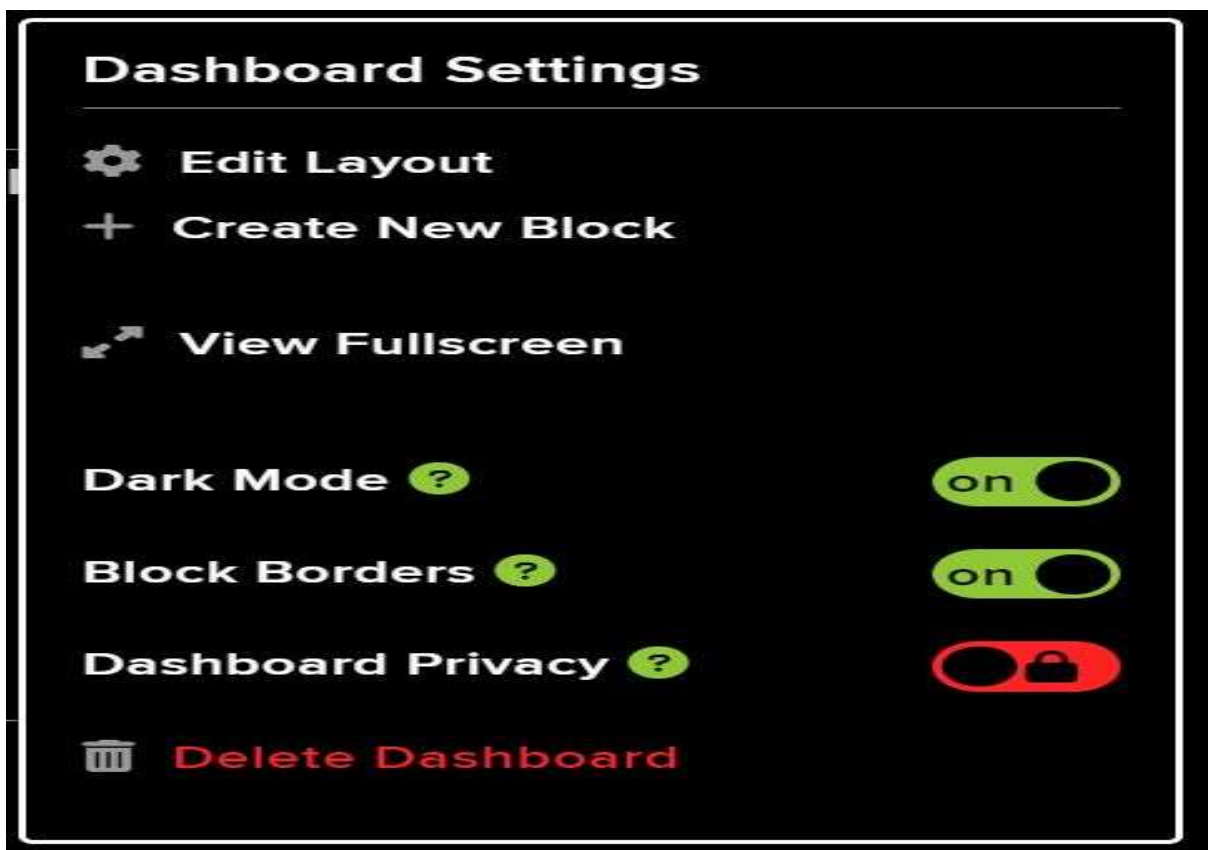
This will be referred to as the IOT block's second component. The foundation of Adafruit IO is fed. Both the uploaded data and the meta-data about the data that sensors submit to Adafruit IO are stored in them. the time and date that it was posted, for instance. Alternatively, the GPS locations where the data originated. Do ThYouant 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]

## RAFSAN / Dashboards



**fig 3.20 IOT platform Creating step 3**

Creating Dashboards for Service will be the project's third component. My returning data may be floated and 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 3.21 IOT platform Creating step 3**

Through this window, I can create different types of action blocks and include dashboards in all those blocks. Block with which contrololcontrolject.

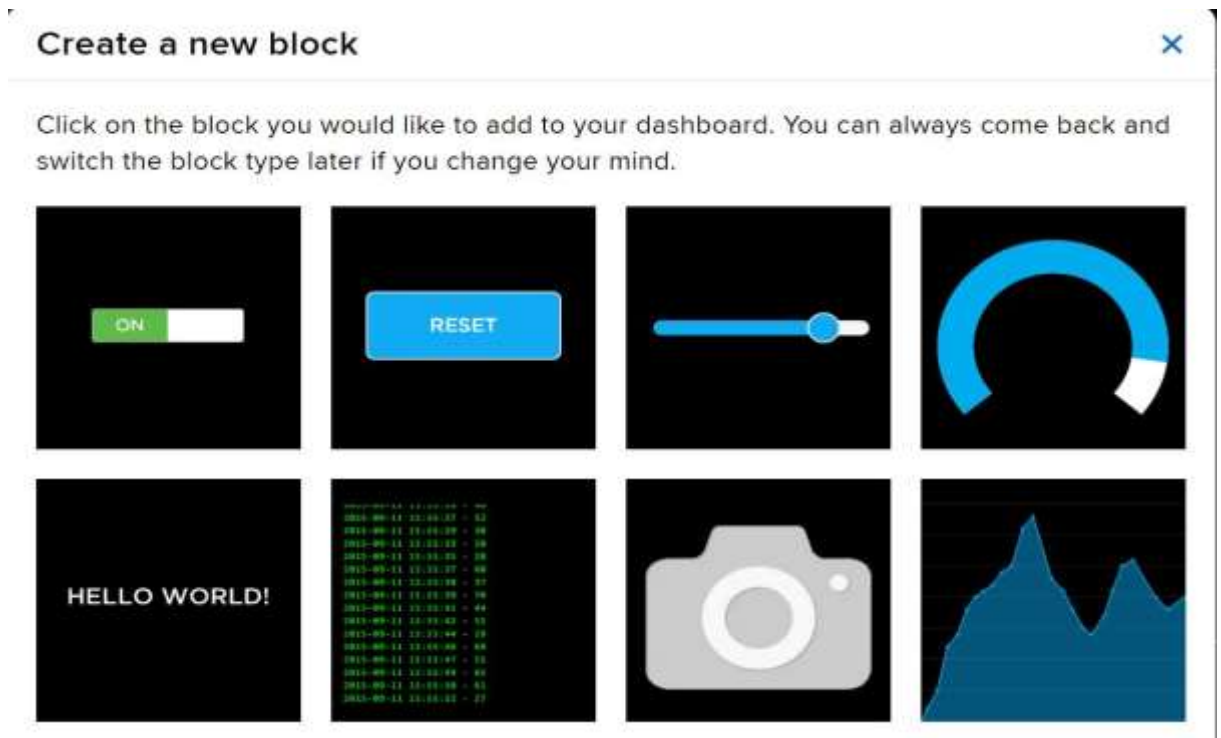
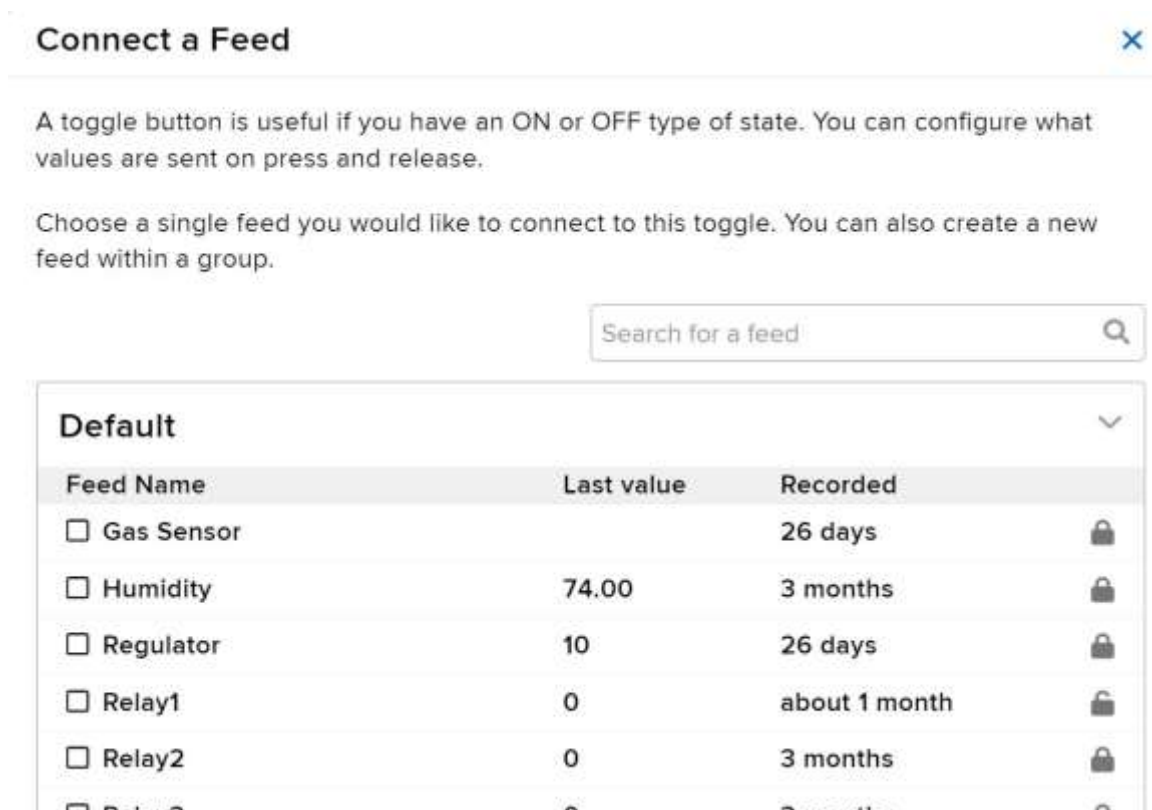


fig3.22 IOT platform Creating step 4

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 blocks. Here it is possible to perform various processes, displays, and controls by bbyisolateinthe lock according to my needs. [10]



**fig 3.23 IOT platform Creating step 5**

Block settings

In this final step, you can give your block a title and see a preview of how it will look. Customize the look and feel of your block with the remaining settings. When you are ready, click the "Create Block" button to send it to your dashboard.

Block Title (optional)

Button On Text

ON

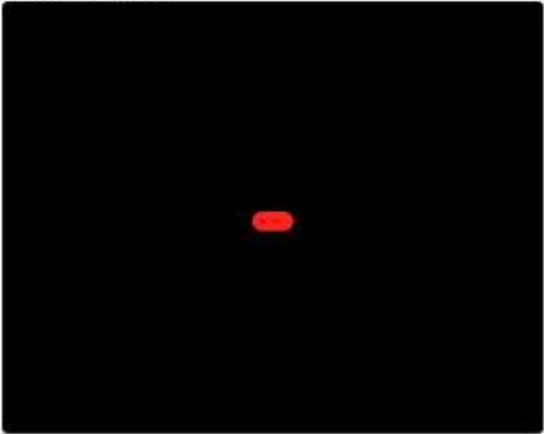
Limit of 6 characters for the toggle text. Use the block title to be more descriptive.

Button On Value (uses On Text if blank)

Button Off Text

OFF

Block Preview



**Toggle** A toggle button is useful if you have an ON or OFF type of state. You can

**fig 3.24 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 their initial value. [10]

32

©Daffodil International University

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

**Video Capture:** Use OpenCV's VideoCapture class to access the webcam video feed to capture it. use Python tools like Mediapipe NumPy, Pandas, SciPy, and scikit-learn 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 my system's phase voltage changes 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.

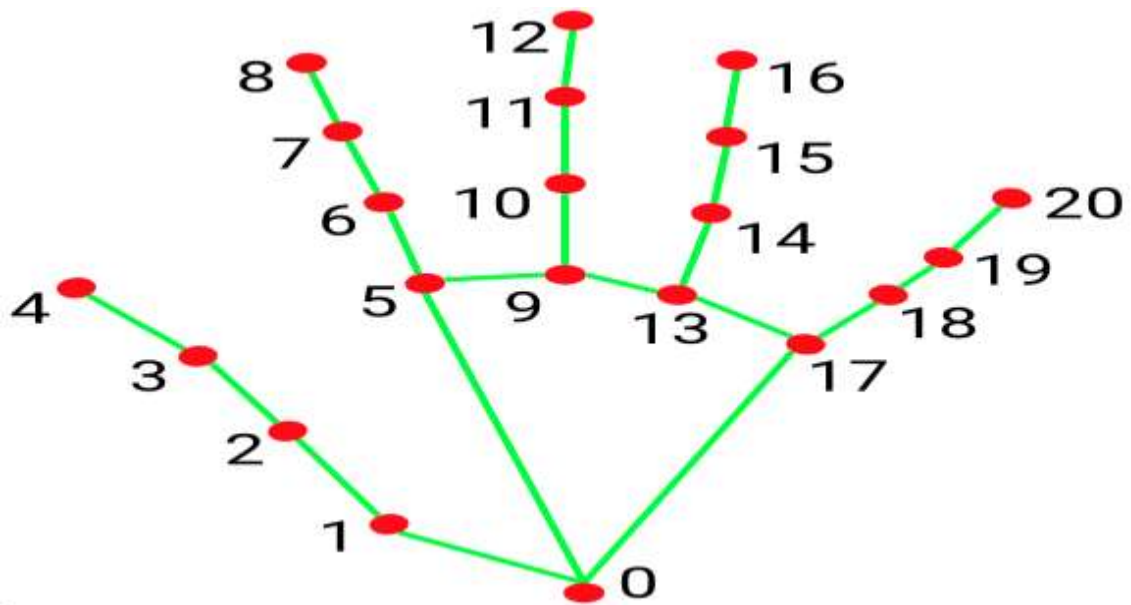
**Map gestures to control:** Create code that associates changes in the phase voltage of this project with detected motions. integrating system parts Create a system that combines all the parts into one whole and utilizes gesture control to change the phase voltage in response to camera input.

MediaPipe is an open-source framework for creating body and hand-tracking apps. A TensorFlow-based hand recognition model that has already been trained is included, and it may be used to recognize hands in a real-time video taken with a camera or a mobile device. The hand detection model's base is single-shot multi-box detection (SSD), which has been improved for real-time performance.

**Install MediaPipe** Refer to the installation instructions supplied by the operating system to install MediaPipe and its dependencies.

Run the pipeline for hand detection: Use MediaPipe's command line interface to execute the hand detection pipeline on a video stream from a webcam or mobile camera.

Consider the results: I can see the hand-bounding boxes and landmarks produced by the hand detection pipeline in the video stream by processing the output in my code or using MediaPipe's built-in visualizers.



**Fig 3.25 21 Hands Landmarks point [35]**

Hand Land Marks	
	0. WRIST
	1. THUMB_CMC
	2. THUMB_MCP
	3. THUMB_IP
	4. THUMB_TIP
	5. INDEX_FINGER_MCP
	6. INDEX_FINGER_PIP
	7. INDEX_FINGER_DIP
	8. INDEX_FINGER_TIP
	9. MIDDLE_FINGER_MCP
	10. MIDDLE_FINGER_PIP
	11. MIDDLE_FINGER_DIP
	12. MIDDLE_FINGER_TIP
	13. RING_FINGER_MCP
	14. RING_FINGER_PIP
	15. RING_FINGER_DIP
	16. RING_FINGER_TIP
	17. PINKY_MCP
	18. PINKY_PIP
	19. PINKY_DIP
	20. PINKY_TIP

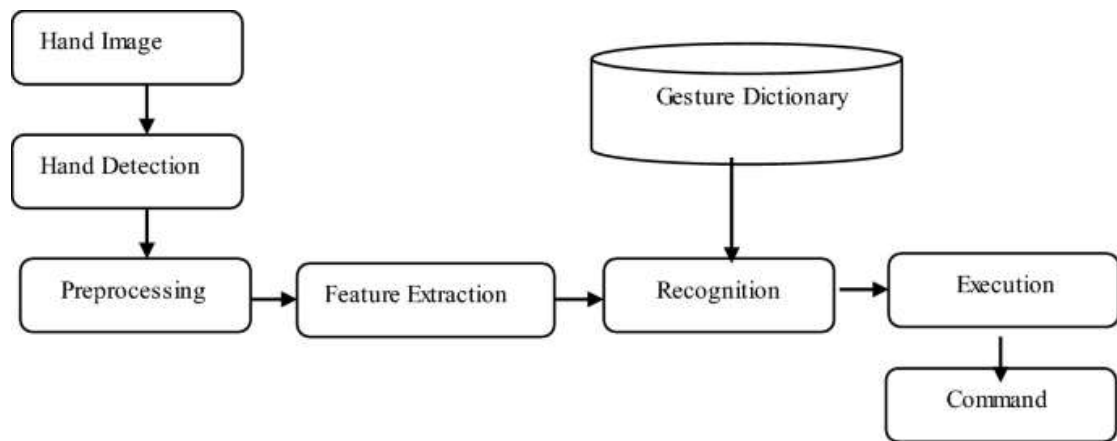
**Fig 3.26 20 Hands Landmarks point Specification [35]**

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

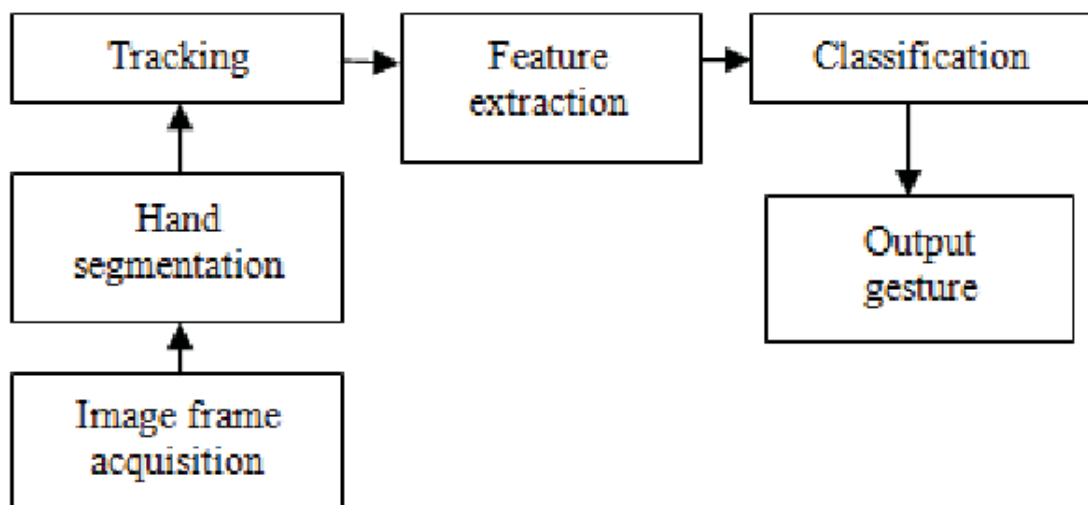
Hand landmarks, which are specific locations on the hand such as the fingers, knuckles, and wrist, can be used to characterize the form and placement of the hand. In MediaPipe, hand landmark estimates are made using thand-boundingg boxes generated by the hand detection model.

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

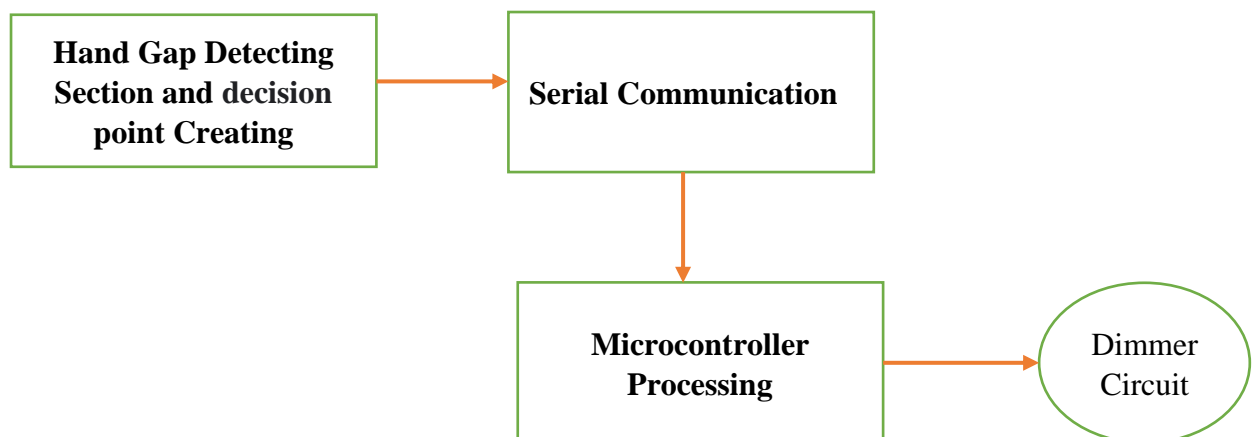
by the model. The landmarks can then be applied to tasks like 3D hand tracking, hand posture estimation, and hand gesture recognition.



**Fig 3.27 Gesture Control Steps 1** [35] [30] [32]



**Fig 3.28 Gesture Control Steps 2** [35] [30] [32]



**Fig 3.29 Block steps of Gesture Control Development** [35]

## **CHAPTER 4**

### **RESULTS AND DISCUSSIONS**

#### **4.1 Results**

Better monitoring and control of physical systems and equipment's temperature and humidity  
The procedures utilized to control the climate and lighting systems in a building are more automated and efficient.

enhanced data gathering and analysis, which might be used to inform choices and boost efficiency. improved system, device, and human connectedness and communication.

increased dependability and security of the system. improved system perception and understanding of visual input, including images and videos. The accuracy and speed of tasks like object recognition, image classification, and facial recognition will increase. increased automation and effectiveness of procedures involving the analysis of visual data.

enhanced decision-making and useful insights by enabling the system to draw out relevant 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.

I will learn several methods of controlling my system through this chapter, along with the outcomes. If This Then That is referred to as IFTTT. It is a free web-based tool for building applets, which are collections of straightforward conditional expressions. Changes made to other web services like Gmail, Facebook, Telegram, Instagram, Google Assistant, or Pinterest cause these applets to be activated. For instance, an applet can transfer a Facebook photo to a user's archive if someone tags them in a photo or send an email message if the user tweets with a specific hashtag.

Along with running on the web, the service is also available on IOS and Android.

In this project, we developed a system that detects and keeps track of the presence of individuals in space using IoT devices and computer vision. A computer, a camera module, and a motion sensor make up the system. Using a trained deep learning model, Python software running on the personal computer examines camera-captured photos to identify persons. Every time a person is identified or leaves the room, the system notifies a mobile app.

In a lab setting, we put the system to the test and assessed its effectiveness in terms of detection precision, response time, and power usage. We compared the system's output to ground truth labels that had been manually annotated by human Gesture and Movement observers to assess detection accuracy. The system properly identified people in 95% of the gesture movements



when we randomly selected 100 individuals from the video feed. The rate of false positives was 5%.

We timed how long it took the system to send a notification when someone entered or departed the room to determine response time. The average reaction time, which we determined to be 1 second, was sufficient for real-time applications.

Overall, the results show that utilizing IoT devices and computer vision for Gesture recognition and Movement monitoring is both practical and efficient. The system has a range of applications, including home automation, security, and surveillance, and it has excellent accuracy, quick response times, and moderate power consumption.

It is crucial to remember that there are restrictions and possible gaps in the system. For instance, the technology may be improved to recognize and track certain persons rather than just detecting people as it does now. The system may also be impacted by occlusions and lighting, which could result in a decrease in detecting precision. The system's performance can change based on the particular dataset and application area because it uses a pre-trained model.

Finally, this project is an illustration of how the practical application of IoT and computer vision in person detection and monitoring might transform many aspects of our life. Even more advanced systems with a larger range of capabilities may result from additional research and development. [7] [5] 26]

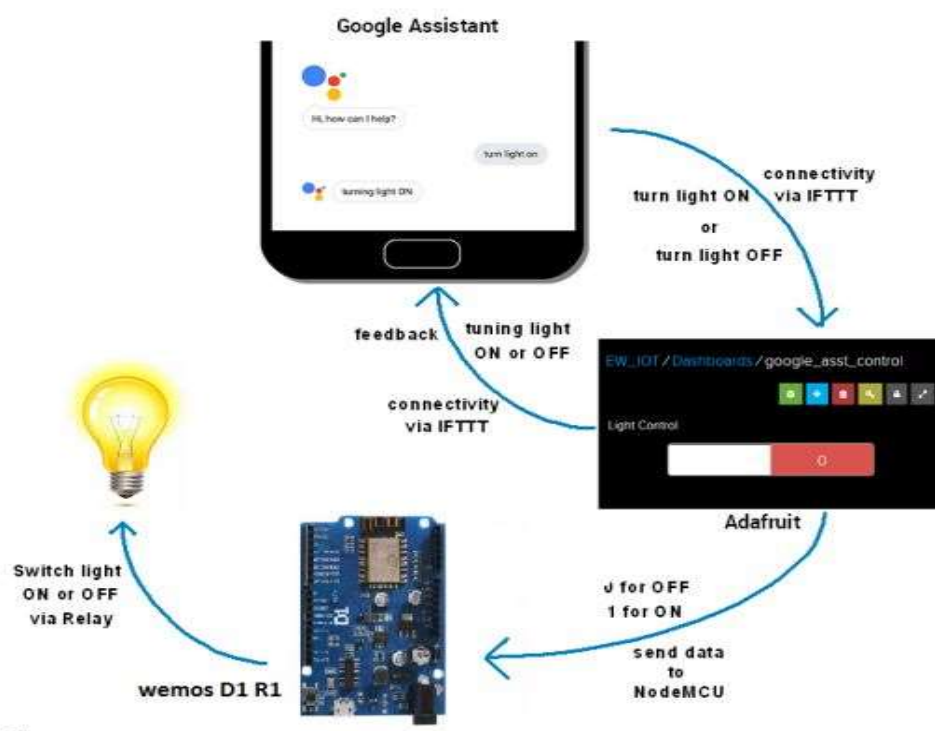
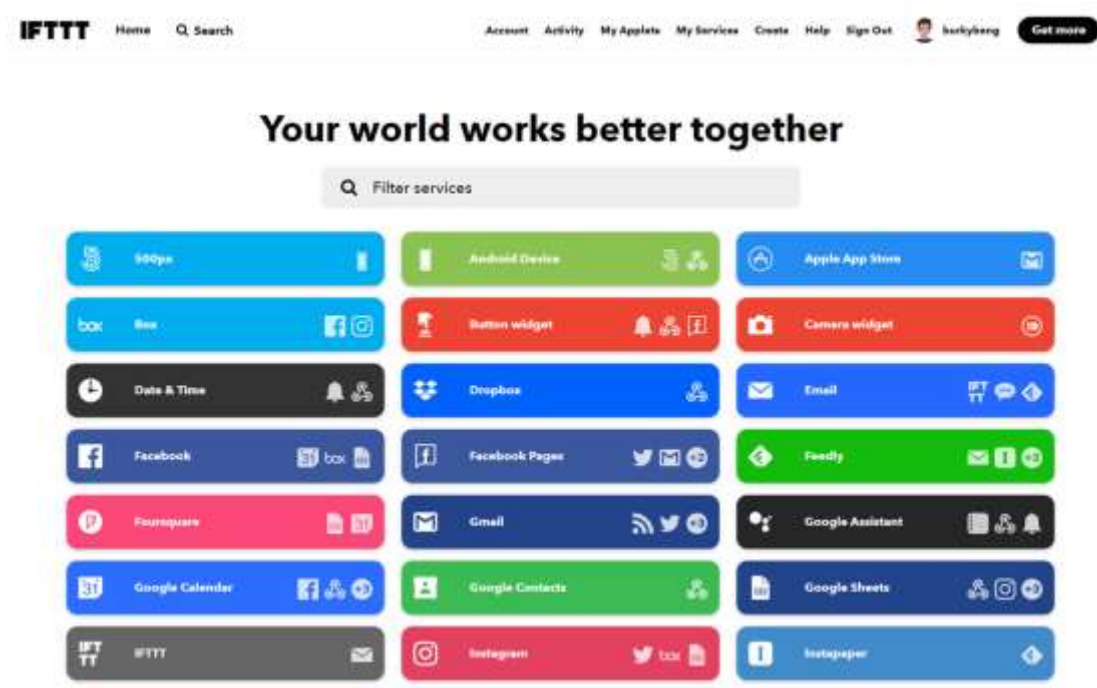


Fig2:Flow Diagram

**Fig: 4.1 Flow Diagram [26]**

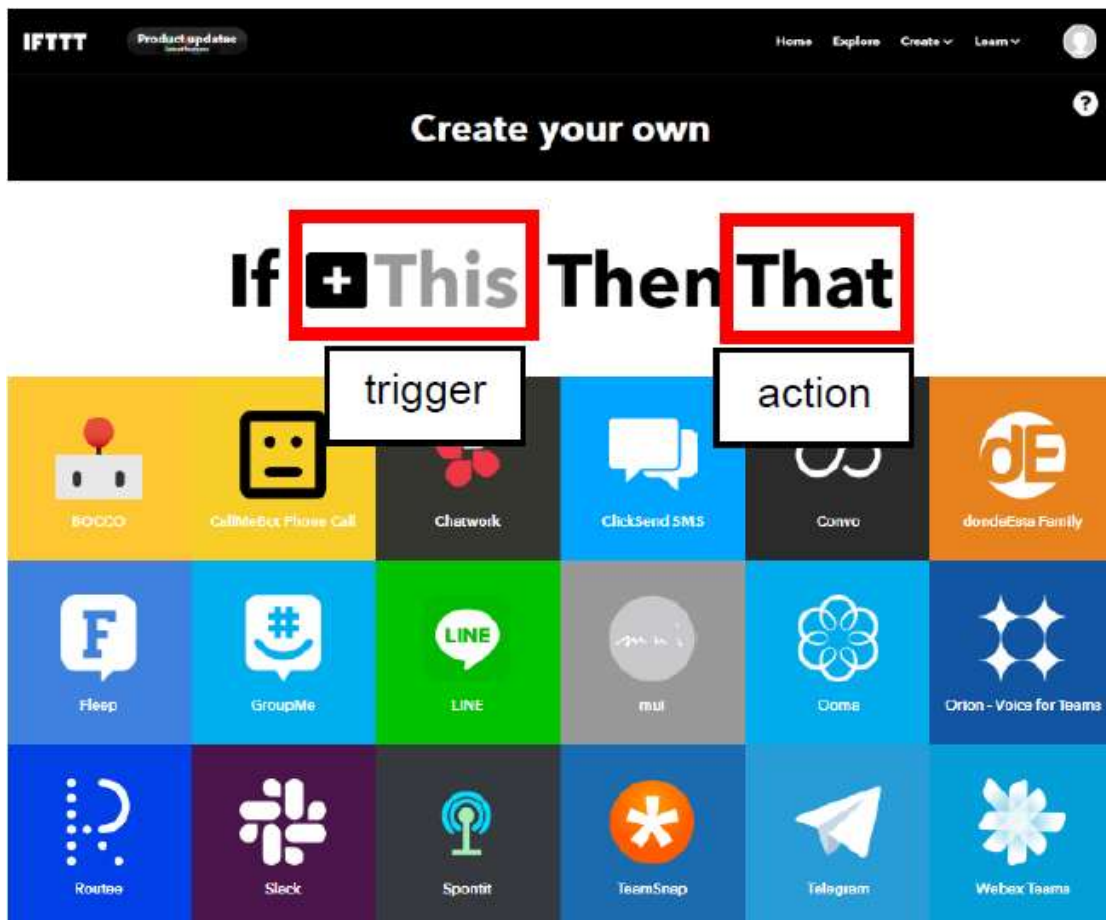


**Fig 4.2 IFTTT Service and Appletes1**



**Fig 4.3 IFTTT Service and Appletes2**

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 source code by utilizing only this Server. Amazon, Alexa, Fitbit, Gmail, Discord, Dropbox, Android device, a device steroid SMS. Along with this, They may also use them for several others[26]



**Fig 4.4 IFTTT Service and Applets3**

This enables a separate web server to set a control depending on a data point and behave accordingly. More than 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 4.5 Google-based Service of this project**

Google Services: Google provides a wide range of services that can be used for various purposes, including controlling IoT devices. We will be using Google Assistant and Google Home as triggers for our IFTTT applets.



**Fig 4.6 Android-based Service of this project**

Android provides a wide range of services that can be used for various purposes, including controlling IoT devices. We will be using the Android voice assistant as the trigger for our IFTTT applets.



**Fig 4.7 Electronic Mail-based Service of this project**

IFTTT and Adafruit IO can also be used to control IoT devices and automate tasks using email services. This figure, controls IO feeds using IFTTT and email services.



**Fig 4.8 Cellular-based Service of this project**

IFTTT and Adafruit IO can also be used to control IoT devices and automate tasks using cellular services.



**Fig 4.9 Social Media based Service of this project**



**Fig 4.10 Social media-based Service of this project**

IFTTT and Adafruit IO can also be used to control IoT devices and automate tasks using Social media platform services.



**Fig 4.10 Self-developed apps for use in projects**

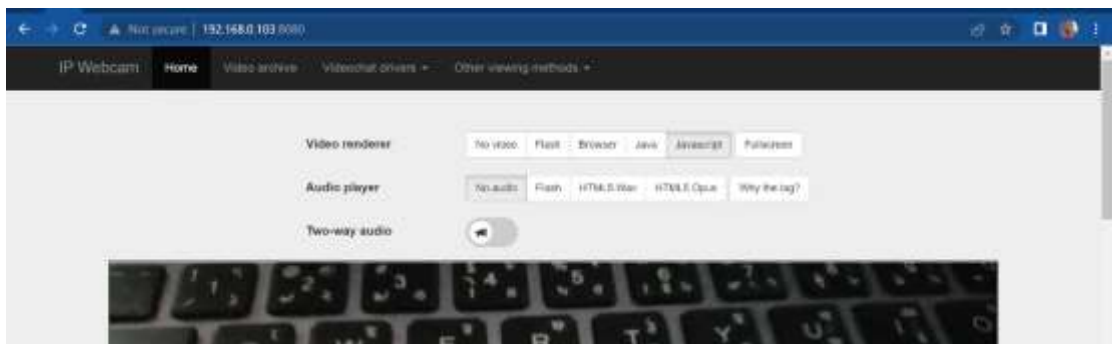
An attempt has been made to develop the total controlling protocols used in the project into an app[16]





**Fig 4.11 Computer vision or video visualization protocols**

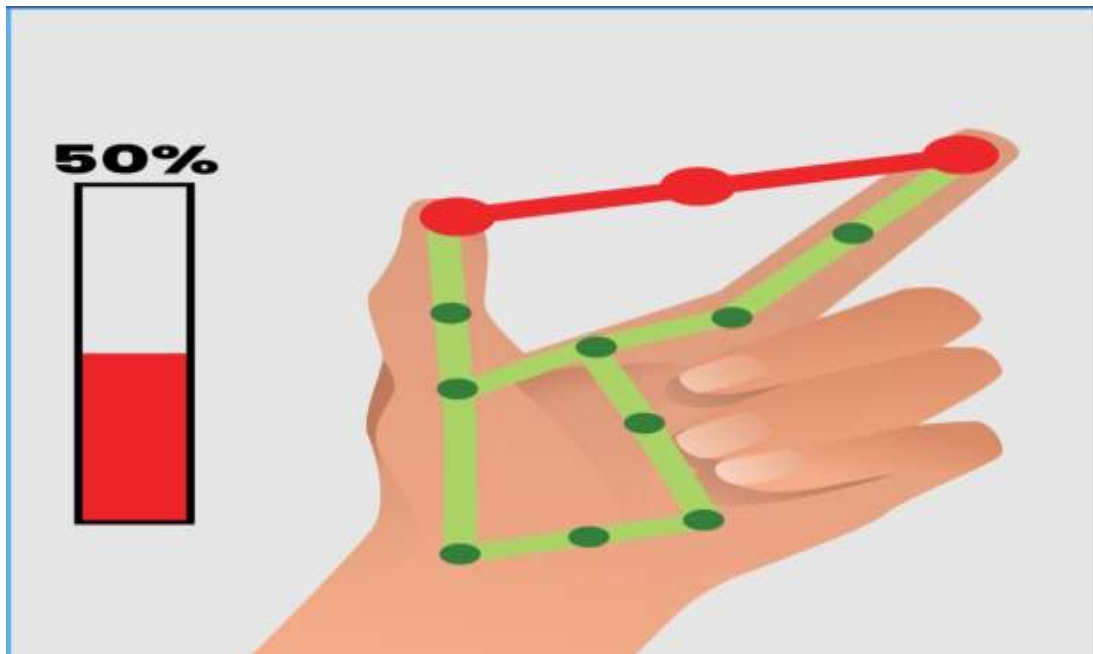
An application called IP Webcam is used for computer vision or video visualization. For now, their mobile phones have been used for object movement detection in Computer vision.



**Fig 4.12 IP link <http://192.168.0.103:8080/> based computer vision or video visualization through the Internet.**



**Fig 4.13 IP link <http://192.168.0.103:8080/> and contours area selection based on computer vision or video visualization software through the Internet.**



**Fig 4.14 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 fine-tune the models to maximize their performance for the particular job. **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++ and Python 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 the 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 them with the platform

**Schedule: 4 weeks**

Milestone: Completion of computer vision algorithms development and integration

Deployment and Testing (Week 9-12)

Task: Deploy the 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	Bulb	200
4	Wire	200
5	Breadboard	150
6	DTH11 Sensor	75
7	Gas Sensor	75
8	Relay Module	250
Total		2450

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

From there, I collected most of the material for This project.

## 5.3 Lesson Learned

While doing this project, I had to get acquainted with different sources. Here I need programming knowledge and basic electronics. And the project had to change many things to match the fantasy with 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 an 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 projects and take steps to ensure that it is deployed responsibly and sustainably.

This project is almost entirely internet dependent. 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 developed 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 the projects 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 is 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 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 this project, we developed a system that controls hand gestures using IoT devices and computer vision. A camera module, a microcontroller, and a relay module make up the system. A convolutional neural network is used to interpret images of hand gestures taken by the camera to identify the intended instruction. After that, the microcontroller instructs the relay strip to carry out the necessary activity.

With a collection of standard Hands gestures, we put the system to the test and assessed its accuracy, responsiveness, and usability.

We compared the output of the machine to the manually annotated ground truth labels to assess accuracy. 90% accuracy for the preset Hands motions was what we discovered, which we thought was adequate for real-world applications.

We timed how long it took the system to carry out the intended action after recognizing a gesture to determine response time. We discovered that the typical reaction time was measured in seconds, which we thought was appropriate for real-time applications.

We surveyed a group of users and solicited their opinions on the system to assess usability. The majority of participants said they would use the system for home automation and entertainment purposes since they found it to be simple to use and intuitive.

The outcomes show that integrating IoT devices and computer vision for gesture control is both feasible and efficient. A variety of applications, including home automation, entertainment, and accessibility, are possible thanks to the system's excellent accuracy, quick response time, and decent usability.

There are, however, several areas that could use improvement and further study. The system might be improved, for instance, to recognize a larger variety of gestures and commands and to adjust to various lighting scenarios and user preferences. The system could also benefit from more complex gesture recognition algorithms, including attention-based models or recurrent neural networks.

Overall, we advise that the system be improved and developed further to increase its functionality and commercial viability. We particularly advise looking into possibilities for user customization and personalization as well as integration with other IoT devices and platforms. The convergence of IoT and computer vision has the potential to change how we interact with technology and enhance our daily lives with continuing innovation and study.

## References

1. "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. "MATEMA6: Machine Tele-Monitoring Assistance with 6LoWPAN," 5th ICT International Student Project Conference (ICT-ISPC), pp. 49–52, 2016. R. Amornpornwiwat, P. Piyachat, V. Chawathaworncharoen, V. Visoottiviseth, and R. Takano.
3. "Smart City Implementation Models Based on IoT Technology," Advanced Science and Technology Letters, vol. 129, no. 2, 2016, pp. 209–212. Jaehak Byun, Sooyeop Kim, Jaehun Sa, Sangphil Kim, Yong-Tae Shin, and Jong-Bae Kim.
4. [HTTPS://WWW.IRJET.NET/ARCHIVES/V7/I3/IRJET-V7I3728.PDF](https://www.irjet.net/archives/V7/I3/IRJET-V7I3728.PDF)
5. [LABS.PT/DOCS/ASSESSING\\_THE\\_ESP8266\\_WIFI\\_MODULE\\_FOR\\_THE\\_IN  
TERNET\\_OF\\_THINGS/1438/VIEW.PDF](https://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#:~:TE  
XT=MOVING%20OBJECT%20DETECTION%20IS%20A,ANY%20MOVING%  
20OBJECT%20IS%20DETECTED](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\\_MOVI  
NG\\_OBJECT\\_RECOGNITION\\_USING\\_VIDEO\\_ANALYTICS](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](https://www.researchgate.net/publication/221127238_Moving_Object_Detection_Under_Free-Moving_Camera)
9. [HTTPS://ARCHIVES.PALARCH.NL/INDEX.PHP/JAE/ARTICLE/VIEW/7766/7  
295](https://archives.palarch.nl/index.php/jae/article/view/7766/7295)
10. [HTTPS://CDN-LEARN.ADAFRUIT.COM/DOWNLOADS/PDF/WELCOME-  
TO-ADAFRUIT-IO.PDF](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](https://cdn-learn.adafruit.com/downloads/pdf/adafruit-io-hub-with-the-adafruit-funhouse.pdf)
12. [HTTPS://IOTDESIGNPRO.COM/ESP8266-PROJECTS](https://iotsdesignpro.com/esp8266-projects)
13. [HTTPS://WWW.GOOGLE.COM](https://www.google.com)
14. [HTTPS://IO.ADAFRUIT.COM](https://io.adafruit.com)
15. [HTTPS://IFTTT.COM/EXPLORE](https://ifttt.com/explore)
16. [HTTPS://CREATE.ARDUINO.CC](https://create.arduino.cc)
17. [HTTPS://STORE.ROBOTICSBD.COM/](https://store.roboticsbd.com/)
18. [HTTPS://WWW.THATTUBE.COM/](https://www.thattube.com/)
19. [HTTPS://WWW.ARDUINO.CC/EN/SOFTWARE/](https://www.arduino.cc/en/software/)
20. [HTTPS://WWW.ROKOMARI.COM/BOOK](https://www.rokomari.com/book)

21. [HTTPS://LASTMINUTEENGINEERS.COM/ESP8266-NODEMCU-ARDUINO-TUTORIAL/](https://lastminuteengineers.com/esp8266-nodemcu-arduino-tutorial/)
22. [HTTPS://LASTMINUTEENGINEERS.COM/ELECTRONICS/ARDUINO-PROJECTS/](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&S\\_CLIENT=IMG&EI=KFY8Y663D7GIMGEZ6RRIDW&BIH=619&BIW=1280](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&s_client=img&ei=KFY8Y663D7GIMGEZ6RRIDW&bih=619&biw=1280)
24. [HTTPS://WWW.THATTUBE.COM/@MURTAZASWORKSHOP](https://www.thattube.com/@murtazasworkshop)
25. [HTTPS://WWW.THATTUBE.COM/@JHANKARMAHBUB](https://www.thattube.com/@jhankarmahbub)
26. [HTTPS://WWW.IRJET.NET/ARCHIVES/V6/I3/IRJET-V6I3951.PDF](https://www.irjet.net/archives/V6/I3/IRJET-V6I3951.pdf)
27. [HTTPS://ELECTROPEAK.COM/LEARN/SPEAK-TO-ARDUINO-CONTROL-ALL-PARTS-BY-GOOGLE-ASSISTANT/](https://electropeak.com/learn/speak-to-arduino-control-all-parts-by-google-assistant/)
28. [HTTPS://MEDIAPIPE.DEV/](https://mediapipe.dev/)
29. 1. "VISION BASED HAND GESTURE RECOGNITION FOR HUMAN-COMPUTER INTERACTION: A SURVEY", ARTIFICIAL INTELLIGENCE REVIEW, VOL. 43, NO. 1, PP. 1-54, 2015.
30. "REVIEW OF GESTURES RECOGNITION BASED ON VISION [J]", COMPUTER SCIENCE, VOL. 43, NO. 6A, PP. 103-108, 2016. JING-GUO YI, JIANG-HUA CHENG, AND XI-SHU KU.
31. "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/MATECCCONF/ABS/2018/91/MATECCCONF\\_EITCE2018\\_03042/MATECCCONF\\_EITCE2018\\_03042.HTML](https://conferences.org/articles/mateccconf/abs/2018/91/mateccconf_eitce2018_03042/mateccconf_eitce2018_03042.html)
33. [HTTPS://DEVELOPERS.GOOGLE.COM/MEDIAPIPE/SOLUTIONS/VISION/GESTURE\\_RECOGNIZER/PYTHON](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](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](https://cinelight.com/tungsten-light-accessories/light-dimmer-1000-watts)
36. [HTTPS://WWW.RESEARCHGATE.NET/PUBLICATION/351999190\\_AN\\_IOT-BASED SMART HOME AUTOMATION SYSTEM](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, AND WENYU LIU, "DEPTH-PROJECTION-MAP-BASED BAG OF CONTOUR FRAGMENTS FOR ROBUST HAND GESTURE RECOGNITION," IEEE TRANSACTIONS ON HUMAN-MACHINE SYSTEMS, VOL.47,
38. "GRAPH FOURIER TRANSFORM BASED DESCRIPTOR FOR GESTURE CLASSIFICATION," 2017 FOURTH INTERNATIONAL CONFERENCE ON IMAGE INFORMATION PROCESSING (ICIIP), PP. 1-6. NOJAYA SHUKLA, VIJAY KUMAR CHAKKA, A. SATYANARAYANA REDDY, AND MADAN GOPAL.
39. 40. SOUKAINA CHRAA MESBAHI, MOHAMED ADNANE MAHRAZ, JAMAL RIFFI, AND HAMID TAIRI, "HAND GESTURE RECOGNITION BASED ON CONVEXITY APPROACH AND BACKGROUND SUFFIXATION," 2018 INTERNATIONAL CONFERENCE ON INTELLIGENT SYSTEMS AND COMPUTER VISION (ISCV), PP. 1–5, 2018.
40. A. K. PATEL AND S. CHATTERJI, "COMPUTER VISION-BASED LIMESTONE ROCK-TYPE CLASSIFICATION USING PROBABILISTIC NEURAL NETWORK," GEOSCIENCE FRONTIERS, VOL. 7, NO. 1, PP. 53–60, 2016.



## APPENDIX A

### TURNITIN REPORT

*Include here the 1<sup>st</sup> page of the 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.

<b>Turnitin Originality Report</b>	
Processed on: 04-Mar-2023 09:21 +06	
ID: 2028485532	
Word Count: 15016	
Submitted: 1	
191-33-849 By Md. Hasemi Rafsanjani Shohan	
Similarity Index <b>17%</b>	<b>Similarity by Source</b> Internet Sources: 14% Publications: 6% Student Papers: 13%

## APPENDIX B

# COMPLEX ENGINEERING PROBLEM SOLVING AND ENGINEERING ACTIVITIES

*Provide that 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	<p>Low: Awareness of IOT and Computer Vision and their potential applications</p> <p>Moderate: Understanding of IoT and Computer Vision architecture, device communication, and data exchange</p> <p>High: Ability to design and implement IoT and Computer Vision solutions, troubleshoot and optimize network performance</p>
P2	Level of interaction	<ol style="list-style-type: none"> <li>1. Real-time performance vs data privacy and security</li> <li>2. Scalability vs cost</li> <li>3. Complex algorithms vs processing power</li> <li>4. Network bandwidth vs data volume</li> <li>5. Various Library addition issues</li> <li>6. Wire Connecting problem</li> <li>7. Bit error rate vs Conductor resistance</li> </ol>
P3	Innovation	<p>The Internet of Things (IoT) is a network of physical things like cars, appliances, and other household products that can connect 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, agriculture, and healthcare.</p> <p>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.</p> <p>IoT and computer vision may be integrated to provide a potent system that can analyze data in real-time and make wise judgments.</p>
P4	Consequences of society and environment	<ol style="list-style-type: none"> <li>1. Technical Complexity Solution from Control System Analysis</li> <li>2. Data Privacy and Security Solution from Digital Electronics Microprocessor Interfacing</li> </ol>

		3. Cost Management from Energy Economics 4. Standardization Solution From Power Electronics Lab 5. Design Solution From the Engineering drawing lab
P5	Familiarity	As Programming languages: C++ Python IoT frameworks: Adafruit io, AArduinoo. cc As Computer Vision libraries: Mediapipe, NumPy
P6	The extent of stakeholder involvement and conflicting requirements	<p><b>End-users:</b> As the project's main benefactors, their needs must be taken into account while determining the project's scope. They could have incompatible criteria, including needing a user-friendly interface and high levels of privacy and security.</p> <p><b>Project sponsor:</b> They give the project essential political and financial backing, but they frequently have demands that contradict with of the end users. For instance, they can put cost effect cost-effectiveness customer happiness.</p> <p><b>Technology providers:</b> They could wish to employ their proprietary technology rather of an open-source solution, which would be in opposition to the project sponsor's and end users' needs.</p> <p><b>Regulators:</b> They may have demands that contradict those of end users, such as the need to access data gathered by IoT devices for surveillance.</p> <p><b>Technical experts:</b> They could have demands that contradict those of the project sponsor and end users, such as the desire to adopt cutting-edge technology even if it might not be practical from a cost or complexity perspective.</p>
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 Interfacing 3. Cost Management from Energy Economics 4. Standardization Solution From Power Electronics Lab 5. Design Solution From the Engineering drawing lab
A5	The 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);
const int analogPin = A0; // gas

// #define USE_SERIAL SerialUSB // Serial for boards with USB serial port
#define USE_SERIAL Serial
#define outputPin D0
#define zerocross D1 // for boards with CHANGEABLE input pins

dimmerLamp dimmer(outputPin, zerocross); // initialise port for dimmer for ESP8266,
ESP32, Arduino due boards
// dimmerLamp dimmer(outputPin); // initialise 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" // Your SSID
#define WLAN_PASS "diu123456" // Your 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 your username
#define AIO_KEY "aio_suNT937Hh0BRJpgGMBTmDcBarBfQ" // Replace with your
Project Auth Key

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

// Create an ESP8266 WiFiClient class to connect to the MQTT server.
WiFiClient client;
// or... use WiFiClientSecure 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");
Adafruit_MQTT_Publish neww2 = Adafruit_MQTT_Publish(&mqtt, AIO_USERNAME
"/feeds/Gas Sensor"); // gas sensor

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(10000))) {
    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%)
}
}

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!"));
}
//delay(100);

// gas

int analogValue = check();

if (!neww2.publish(analogValue)) {
    //Serial.println(F("Failed"));
} else {
    //Serial.println(F("OK!"));
}
//delay(100);

```



```

    // e
}

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!");
}

int check()
{
    int analogValue = analogRead(analogPin); // 1023
    analogValue = map(analogValue, 0, 1023, 0, 100);
    // Serial.println(analogValue);
    return analogValue;
}

```

**Fig A1: Code for IOT 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](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 MQ2 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](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](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>

- Full project Documents and raw materials and recruitment code are linked below: <https://github.com/Rafsan12345/Final-year-project-.git>