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

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

Submitted by
Name: Md. Hasemi Rafsan Jani Shohan
ID: 191-33-849

Supervised by
Ms. Tanjum Rahi Akanto
Lecturer
Department of Electrical and Electronic Engineering

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

JANUARY, 2023

DECLARATION

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

Signature of the candidates

____

Name: …………………..

ID: ……………………..

APPROVAL

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

Signed

____

Ms. Tanjum Rahi Akanto
Lecturer
Department of Electrical and Electronic Engineering
Faculty of Engineering
Daffodil International University

ACKNOWLEDGEMENT

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

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

for engaging me in such an important research. She was always there to share his absolute expertise and valuable time.
It was his constant guidance, helpful suggestions, constructive criticism and endless patience throughout development of this project .

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

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

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

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

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

LIST OF SYMBOLS

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

ABSTRACT

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

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

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

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

Computer vision, which permits remote visual inspection of controlled areas, and Gesture Control Phase angle Converter are crucial parts of this project.
For this project, I created computer vision software. Based on an IP link, it allows for video visualization. It makes it possible to detect the movement of things at a given place around the globe. This, in my opinion, will considerably enhance robotic vision.

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

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

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

CHAPTER 1

1.1 INTRODUCTION

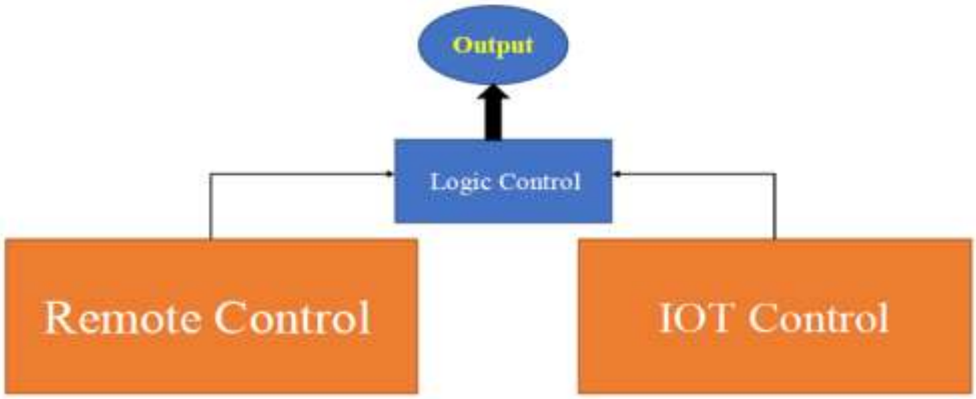


Fig 1.1 Logic Controll Block Diagram of this project

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

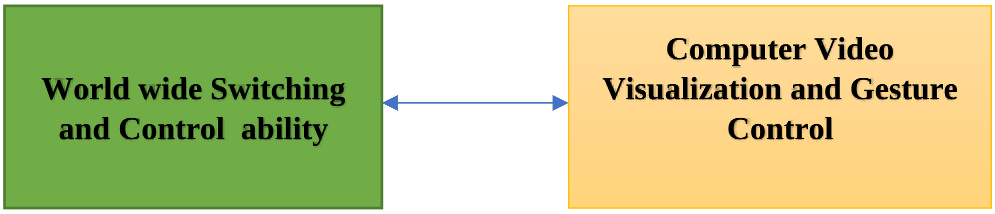
By this the analog switching method is changed and converted to IOT. It enables switching from anywhere in the world. And also connected voltage divorce key temperature sensor and gas sensor. With the help, it is possible to determine the temperature and humidity and the amount of methane gas present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

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

And I have also set up a mail system to send warning messages in different situations. The biggest advantage of This project is that it can be controlled with a system without changing the code to control it.
An important component of this project is computer vision, which enables visual inspection of controlled locations from remote locations and Phase voltage (angle) control by Finger Gesture Movement.

I have developed a computer vision software for this project. It enables video visualization based on IP link. It enables movement detection of objects at a specific location over the world .
I think robotic vision can be greatly improved with this.
It has the default set to IP Link and Control Area. Image Capture arrangements are in project
I hope that my ideas from this project will contribute to the development of updated electrical project in the near future. Above all I would like to thank my supervisors and all my associates for their efforts and their cooperation which facilitated the completion of This project .

1.2 Problem Statement and /or Proposed solution(s)



The increasing use of Internet of Things (IoT) devices in homes and businesses has led to a need for a more efficient and secure way to manage these devices. Currently, users must individually access and control each device through separate apps or web interfaces, which can be time-consuming and confusing.
The proposed solution is to develop a central management platform for IoT devices. This platform will allow users to easily access and control all of their connected devices through a single interface. The platform will also incorporate advanced security features to protect against hacking and unauthorized access. Additionally, the platform will include analytics and data visualization tools to help users better understand their usage patterns and optimize their device settings.

1.3Aims/ Objectives

To develop a central management platform for Internet of Things (IoT) devices that is user-friendly and efficient.
To improve the security of IoT devices by incorporating advanced security features into the management platform.
To provide data analytics and visualization tools to help users understand and optimize their device usage.
Objectives:

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

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

To integrate advanced security features, such as encryption and authentication, into the platform.
To test the developed platform with a group of user to gather feedback and make necessary improvements.

To deploy the platform to a small group of early adopters to gather additional feedback and make any final adjustments before a full launch.
To develop data analytics and visualization tools and integrate them into the platform.
To measure the success of the platform through key performance indicators such as user satisfaction, security, and ease of use.

Project aims , objectives and uses Expected Outcome

- Home Automation.
- Industrial Machine Controlling.
- Weather Monitoring.
- Robot's Arm and Robot Control Car Controlling.
- Smart Irrigation System Control.
- Using Lab use for IOT Smart Trainee Module Board.
- Power System And Grid Network Controlling . .
- Medical and ICU Instruments Controlling

1.3 Brief Methodology / Technologies/ Procedures

1.4 Working Principle

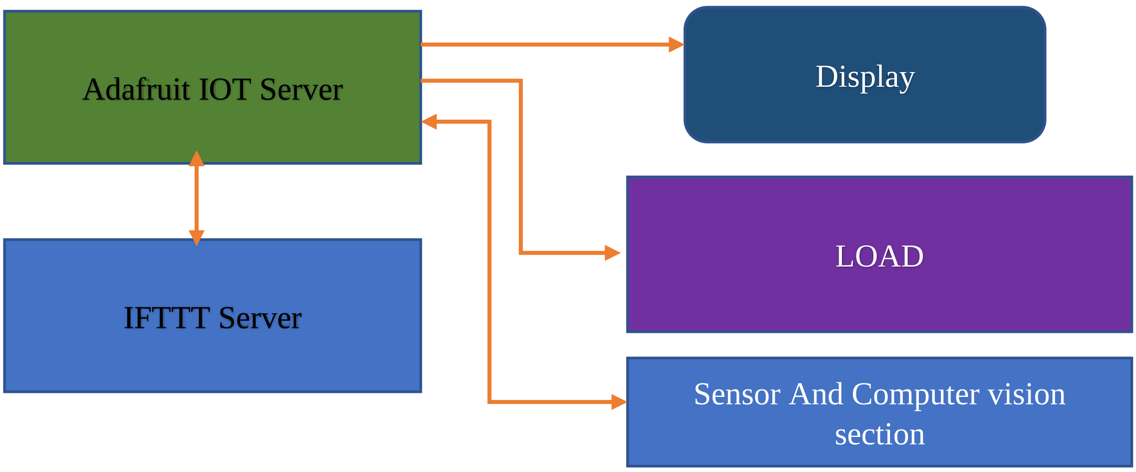


Fig 1.2 Control Block of this project

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

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

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

And used RBD Voltage Dimmer module as voltage dimmer.

Used adafruit.io website server as virtual control platform.

I used a web server called IFTTT for controlling.

And used four relay modules.

Used tv remote control system for manual control.

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

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

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

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

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

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

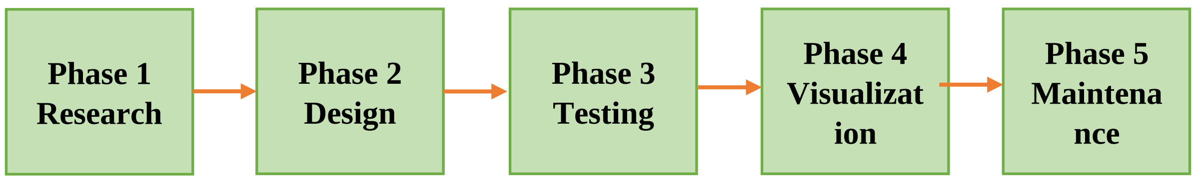
In this project it will basically detect the motion of any type of object and can detect the movement.

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

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

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

1.5 Implementation Schedule



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

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

Analyze user needs and requirements

Develop project scope and objectives

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

Design user interface for the central management platform

Develop platform architecture and integrate security features

Develop computer vision algorithms and integrate with the platform

Test platform with a small group of users for feedback

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

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

Make final adjustments based on user feedback

Prepare for full launch

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

Develop data analytics and visualization tools

Integrate tools into the platform

Test and optimize analytics and visualization features

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

Launch the platform to the public

Monitor and maintain the platform

Continuously gather user feedback and make improvements as necessary

2.1 Introduction

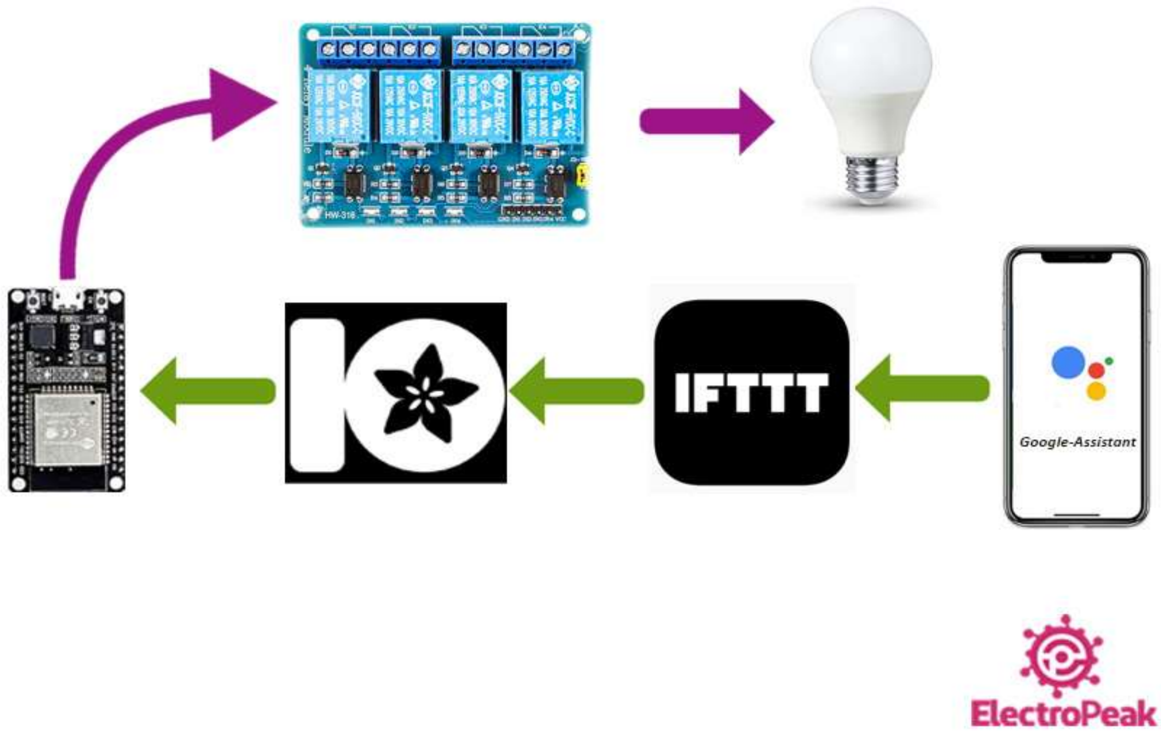


Fig 2.1 Control Steps Block[27]

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

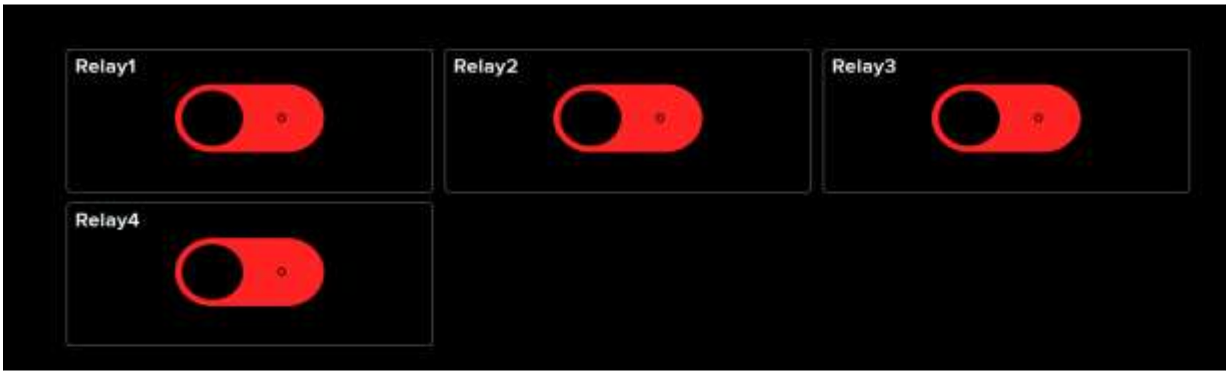


Fig 2.2 Switching Control Block

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

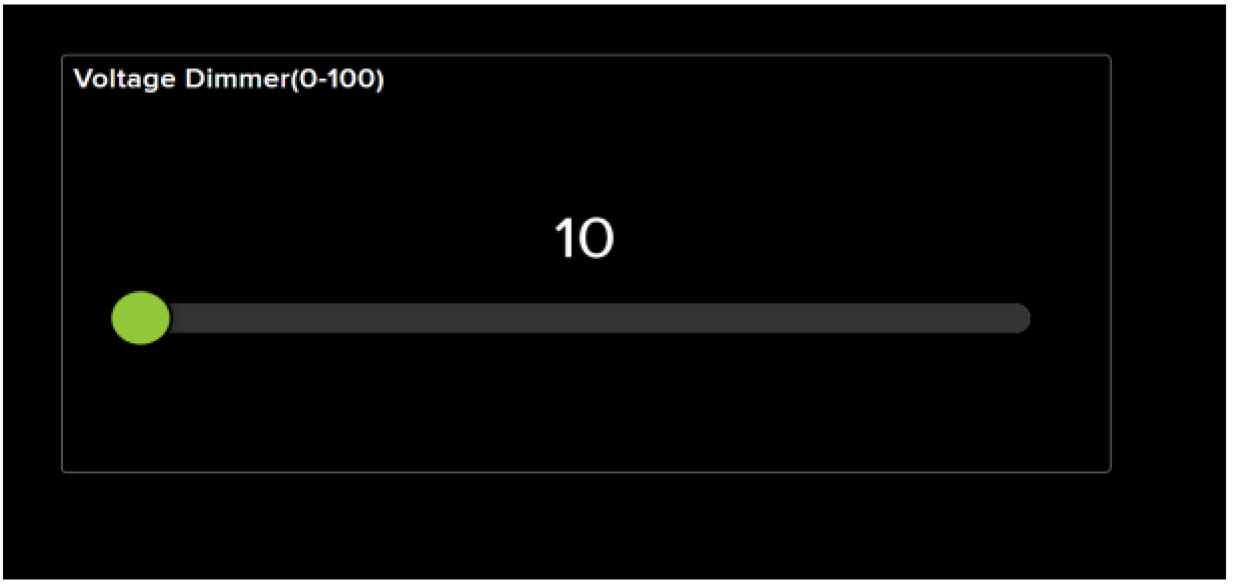


Fig 2.3 Voltage Dimmer Control Block

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

Fig 2.4 Gas Sensor Value Showing Block

This Figure shows the amount of gas sensors that can be used to monitor the amount of gas at a controlled location from a remote location.
With the help of which is the amount of methane gas or LPG gas in a particular place.

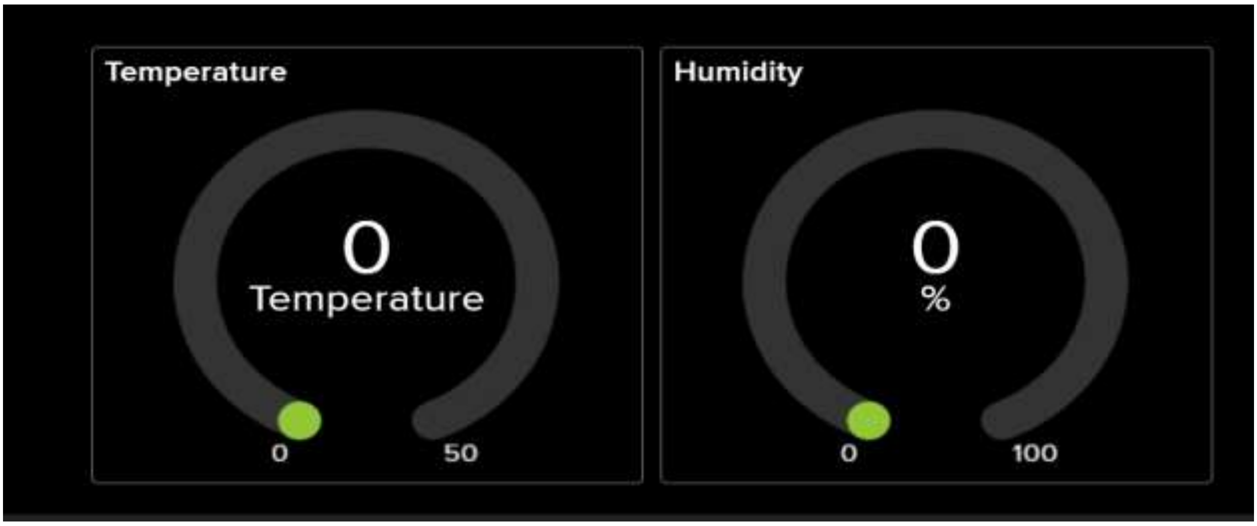


Fig 2.5   Temperature and Humidity Showing Block

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

Fig 2.6   Manual remote Control Based Load Switching

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

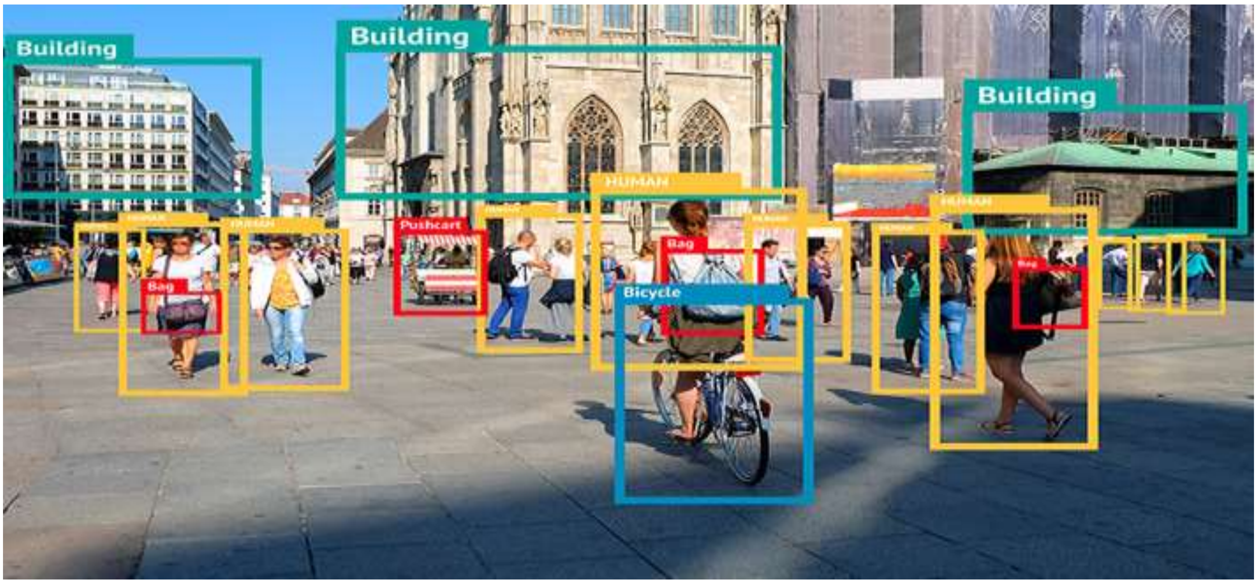


Fig 2.7   Computer Vision Object Movement Showing frame

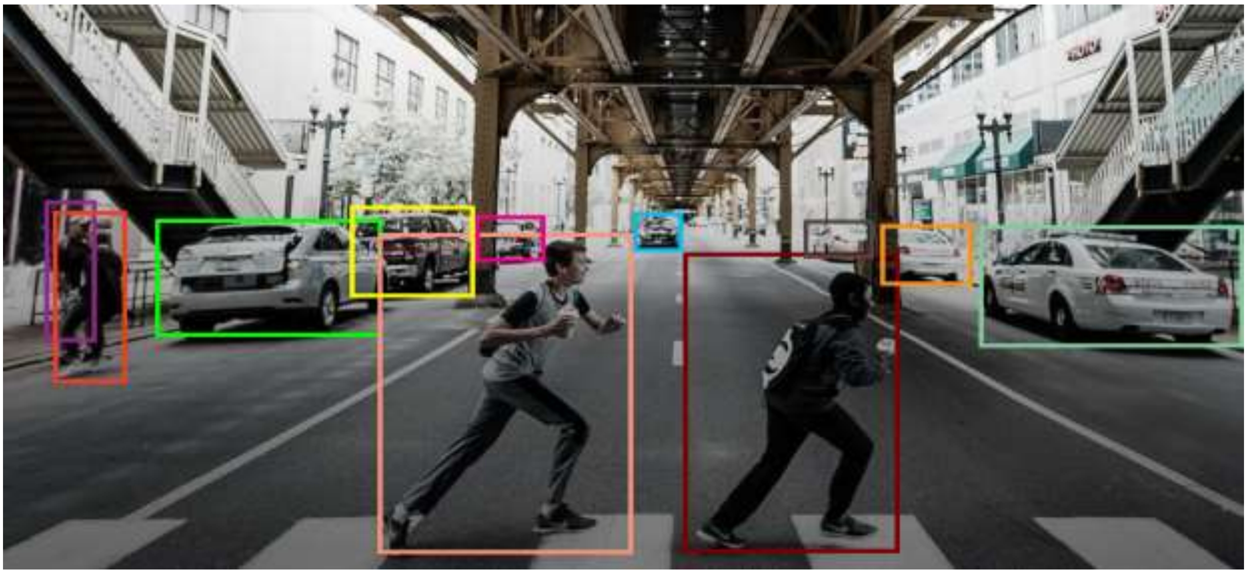


Fig 2.8   Computer Vision Object Movement Showing frame2

Computer vision is shown at in this figure of this chapter. With the help of computer vision, it is possible to show the video visualization view of my controlled space from a remote location. It can detect the movement of an object, the presence of a human face, and the presence of a human hand.
This is best for viewing convenience. [8]

Fig 2.9 Self Developed Computer Vision Sofware Object Movement initial frame

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

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

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

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

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

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

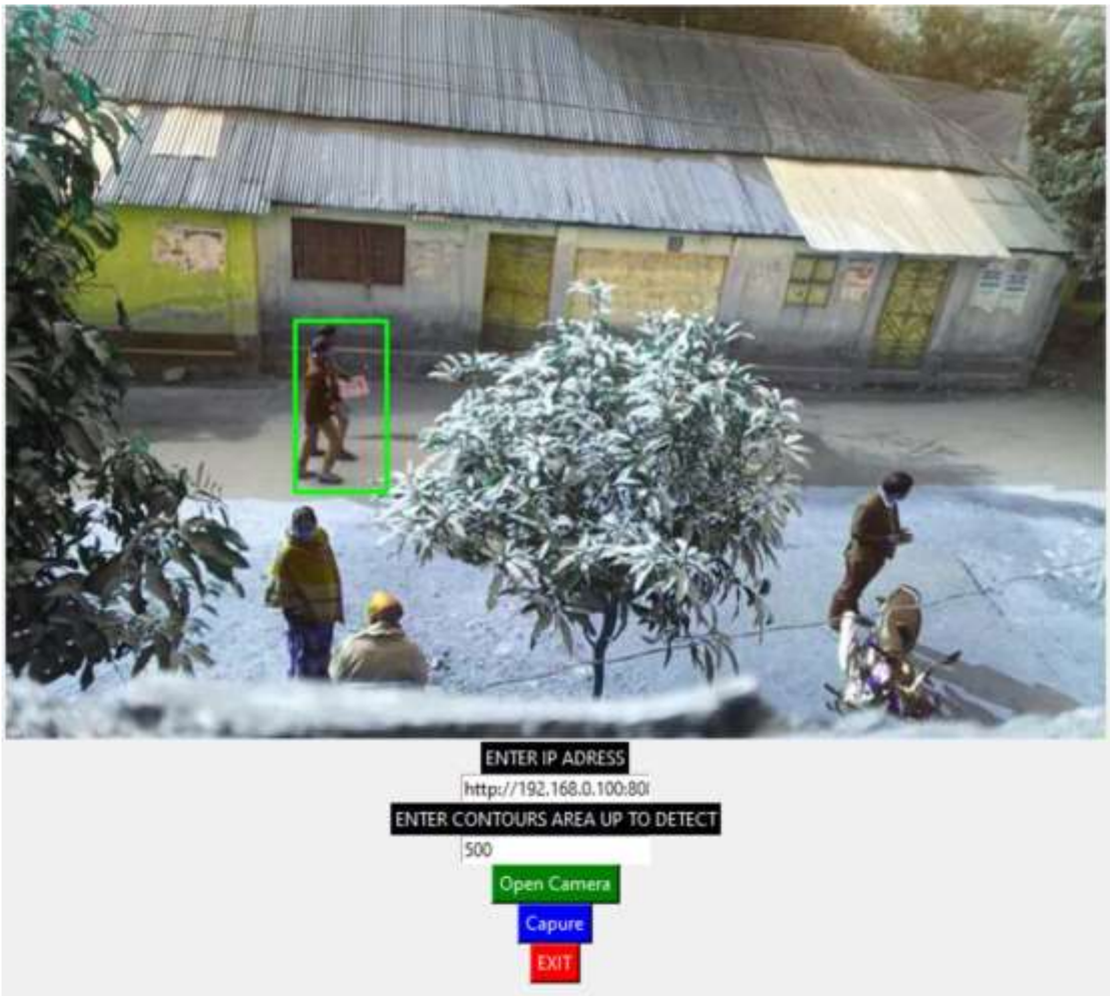


Fig 2.10 After clicking “open camera” button Computer Vision Movement frame.

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

Gesture Movement based phase angle control part:

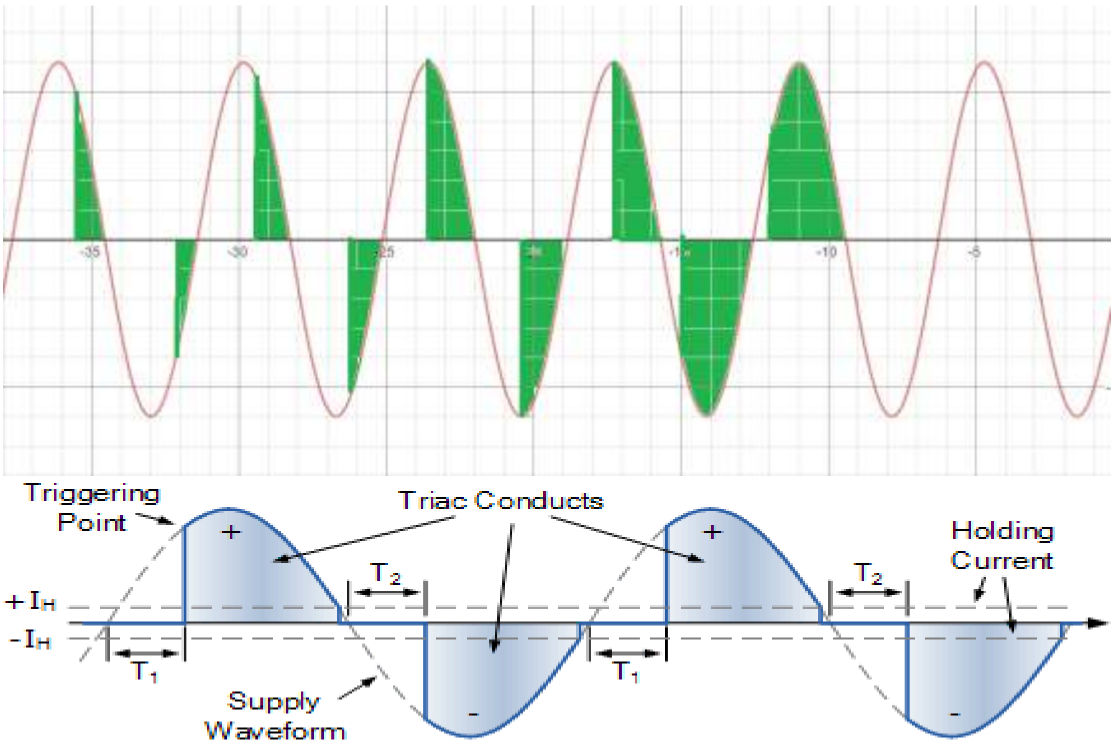


Fig 2.12 Gesture Control phase Voltage Triggering point view.[35][34]

Computers can comprehend human body language with the aid of gesture recognition. Instead of relying solely on text-based or graphical user interfaces, this promotes the development of a more powerful connection between humans and technology (GUIs). The movements of the human body are interpreted by the computer camera in this gesture recognition experiment. The computer then uses this information as input to manage apps. In order to adjust the voltage phase amplitude level, this project aims to provide an interface that dynamically captures human hand gestures.
The Python programming language now has support for massive, multi-dimensional arrays and matrices, as well as a large number of high-level mathematical functions to work on these arrays, thanks to the NumPy package. A free machine learning resource is Mediapipe. Google's open-source Mediapipe machine learning library offers encapsulation for python, js, and other languages and includes some solutions for face and gesture detection. A high-fidelity hand and finger tracking solution is MediaPipe Hands. It employs machine learning (ML) to extrapolate 21 crucial 3D hand details from a single frame. It can be used to obtain the coordinates of the hand's important spots. **[28][29][30][31][32][33][34]**

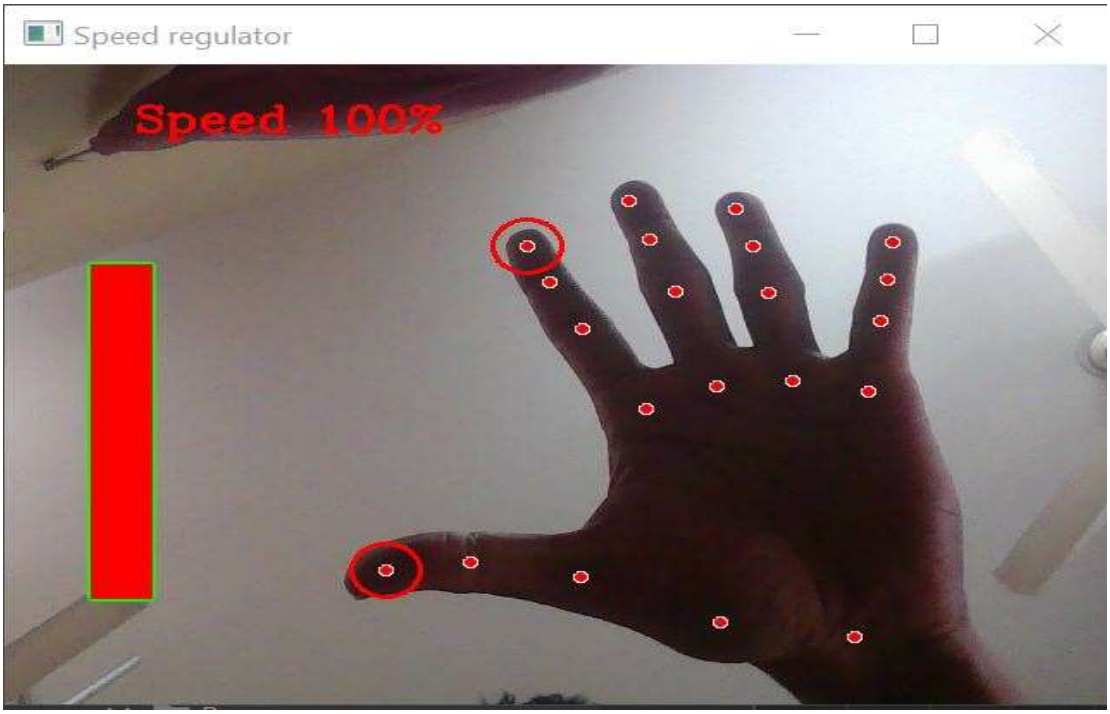


Fig 2.13 Gesture Control High Level gap between index and thumb finger .

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

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

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

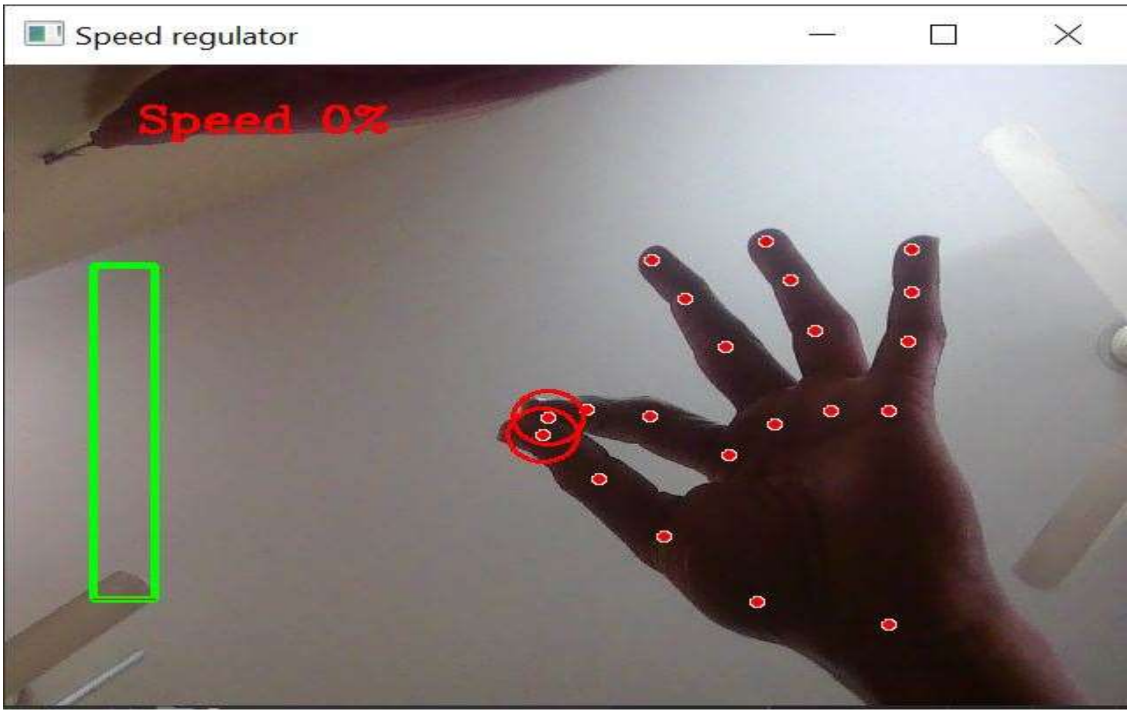


Fig 2.16 Gesture Control Zero Level gap between index and thumb finger .

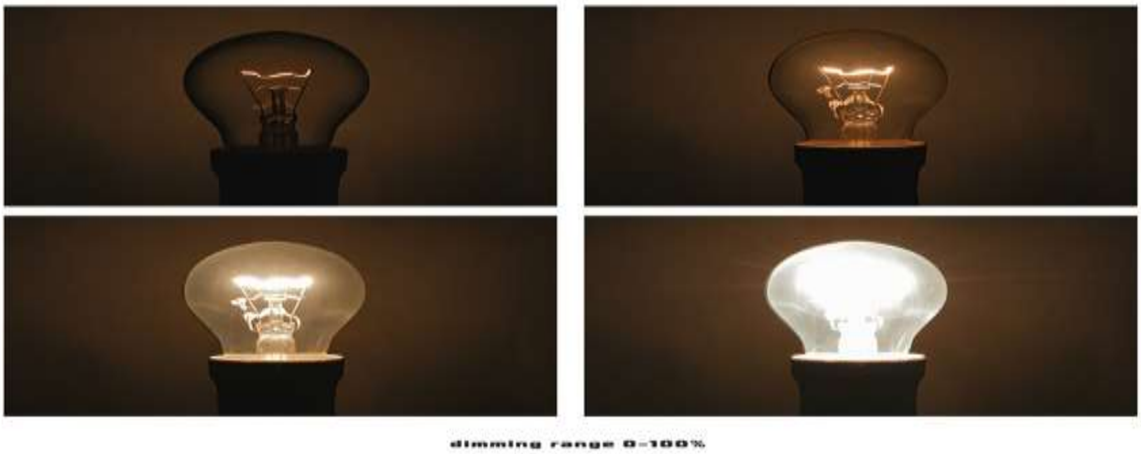


Fig 2.17 After Gesture Control Zero Level to High Level Voltage amplitude and brightness fluctuate Result.[35]

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

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

Chapter 2

LITERATURE REVIEW

2.1 INTRODUCTION

Home automation has grown significantly in popularity in recent years as daily living becomes easier as a result of the quick development of technology. Nearly everything is now automated and computerized. A system for integrating sensors, actuators, and other data sources for various home automations is suggested in this study. The system is known as qToggle and operates by utilizing the flexibility and power of an Application Programming Interface (API), which serves as the basis of a straightforward and widely used communication mechanism. The majority of the time, qToggle is utilized with sensors or actuators that have an upstream network connection and implement the Toggle API. The majority of the devices utilized by qToggle are built using Raspberry Pi boards and/or ESP8266/ESP8285 chips. A smartphone app has been created that enables users to manage a number of home gadgets and sensors. The Toggle system is user-friendly, adaptable, and can be further enhanced utilizing a variety of accessories.[[37] [37] [37] [37]

The main objective of the multidisciplinary discipline of computer vision is to comprehend the environment as closely as human perception can. Equally important as research on techniques and algorithms that enhance the capabilities of a computer vision system is research on the architecture and design of these systems. In this article, we suggest an open-source, Python-based End-to-End Computer Vision Framework with the goal of supporting researchers working in this field. Given the continuing need to add new computer vision algorithms or machine learning models for ongoing research efforts, the framework places a lot of focus on the system's flexibility and scalability.[37] [37] [37] [37]

The use of hand gestures is one of the most intuitive ways to interact with a computer, and if moving hand movements are accurately translated in real-time, there may be a wide range of applications. The author of this paper has developed and built a system that can recognize motions in front of a web camera in real time using motion history images (MHI) and feedforward neural networks. The background from collected frames is first removed using a Gaussian mixture-based background/foreground segmentation approach in order to capture moving areas in the image. The frame is then cleaned up using median filtering to reduce random noise. These processed frames are concatenated, and a cumulative motion is created by using binary thresholding with Otsu's binarization, which will find the appropriate threshold value.**[38]** [37] [37] [37] [37]

2.2Related Research/ Works Compare and Contrast

Related work for IOT project:

IoT-based Smart Switching and Controlling: The integration of IoT technology in smart switching and controlling systems has been widely researched, with a focus on improving automation, efficiency, and safety. This has led to the development of various IoT-based solutions for home automation, industrial control, and smart grid management.

Wireless Communication Technologies: The development of wireless communication technologies, such as Zigbee, Bluetooth, and Wi-Fi, has enabled the integration of IoT in smart switching and controlling systems, providing reliable and low-latency communication between devices.

Cloud Computing: Cloud computing has become a crucial component in IoT-based smart switching and controlling systems, as it provides scalable computing and storage resources for big data processing, as well as enables remote monitoring and control of devices.

Interoperability and Standardization: Interoperability and standardization are critical for the development of IoT-based smart switching and controlling systems, as they ensure seamless integration and compatibility of different devices and systems.

Security and Privacy: The integration of IoT in smart switching and controlling systems poses security and privacy risks, as it involves the exchange of sensitive information between devices. Hence, secure and privacy-preserving protocols and solutions have been widely researched to address these challenges.

Resource-Constrained Devices: The integration of IoT in smart switching and controlling systems presents challenges in terms of computational and energy constraints, particularly for resource-constrained devices such as smart sensors and wearable devices.

Artificial Intelligence and Machine Learning: 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, by enabling real-time monitoring and prediction of system behavior.

Related work for computer vision project:

Computer Vision-based Gesture Recognition: The use of computer vision techniques for gesture recognition has been widely researched, with applications in areas such as human-computer interaction, gaming, and sign language recognition.

Deep Learning Approaches: Deep learning models, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs), have been extensively used in computer vision for gesture recognition, due to their ability to learn from data and improve performance over time.

Hand and Body Pose Estimation: Hand and body pose estimation algorithms are critical components of computer vision-based gesture recognition, as they enable the estimation of the position and orientation of the hands and body, which are used to infer the intended gesture.

Multi-modal Input: The integration of multi-modal inputs, such as depth data from RGB-D cameras, accelerometer and gyroscope data from wearable devices, and audio data, has been researched to improve the accuracy and robustness of computer vision-based gesture recognition.

Real-time Systems: Computer vision-based gesture recognition systems need to operate in real-time to be useful in practical applications, and real-time performance has been a major focus of research in this field.

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

Resource-Constrained Devices: The integration of computer vision-based gesture recognition in resource-constrained devices, such as wearable devices and mobile phones, presents challenges in terms of computational and energy constraints, and has been a major focus of research in this field.

2.4 Summary

The integration of the Internet of Things (IoT) and computer vision technologies has the potential to revolutionize various domains, including home automation, smart health, and gaming. In the context of IOT-based switching and controlling modules, computer vision can be used to enable real-time monitoring and control of physical devices in a more user-friendly way, through the recognition of gestures, facial expressions, or body poses. On the other hand, in computer vision-based gesture recognition projects, IoT devices such as wearable devices and mobile phones can provide multi-modal input, such as depth data, accelerometer data, and audio data, to improve the accuracy and robustness of gesture recognition algorithms. The combination of IoT and computer vision technologies presents both opportunities and challenges, including real-time performance, computational and energy constraints, and data privacy and security.

Chapter 3

MATERIALS AND METHODS [OR PROJECT/SYSTEM DESIGN] [OR DESIGN PROCEDURE]

3.1 Introduction

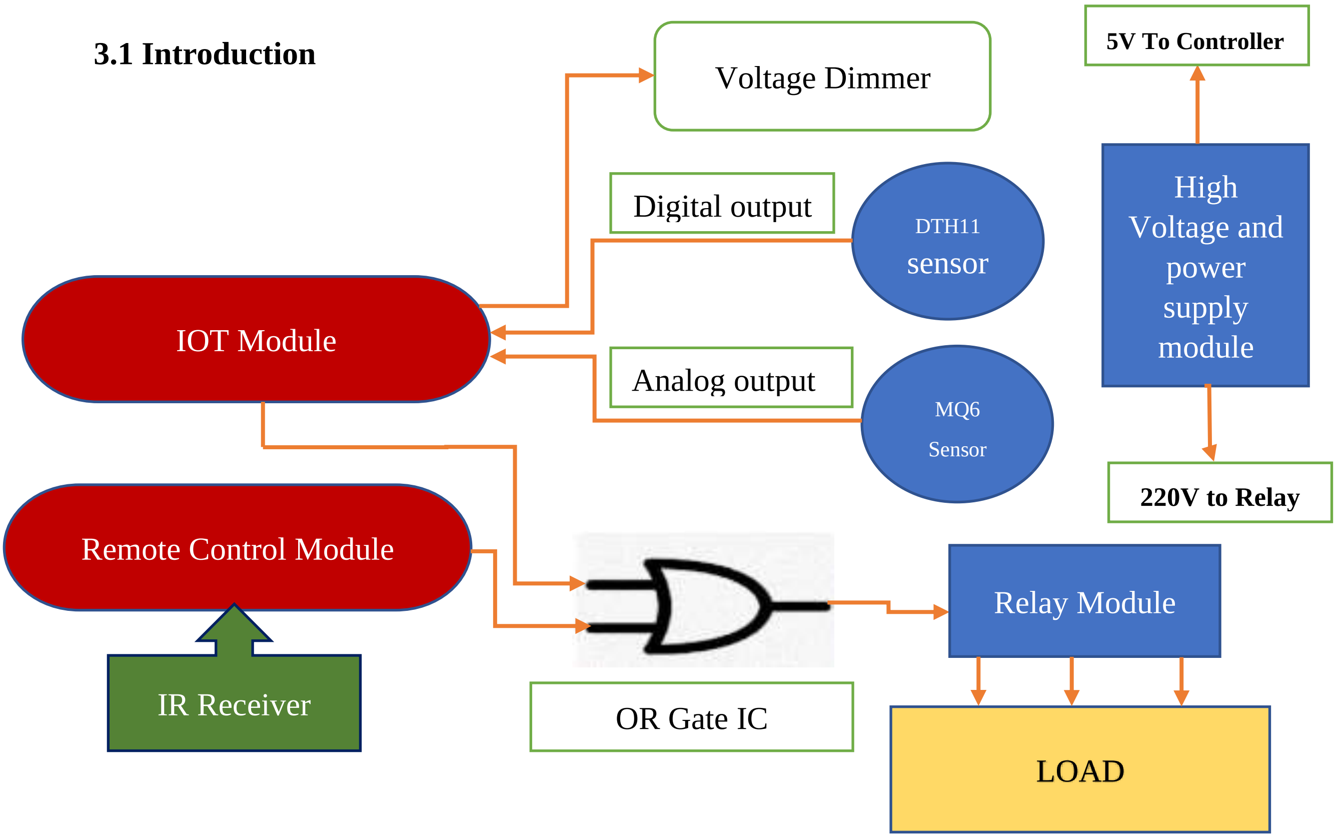


Fig 3.1 IOT and Manual Control structure diagram of this project

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

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

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

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

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

Design Procedure Introduction for Computer Vision Project:

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

3.2 Methods and Materials or System Design and Components

Methods:
Identifying the problem or need that the IoT system will address.
Researching and selecting the appropriate hardware and software components for the system.
Designing the architecture of the system, including the communication protocols, data flow, and security measures.
Implementing the system, including programming the devices and integrating them with the network.
Testing the system to ensure proper functionality and performance.
Deploying the system in the target environment and monitoring it for any issues.
Materials:
IoT devices such as sensors, actuators, and microcontrollers.
Networking equipment such as routers and gateways.
Cloud-based platforms or servers for data storage and processing.
Power supply and backup batteries.
Cabling and wiring.
Development tools such as programming languages, software development kits (SDKs), and integrated development environments (IDEs).
Computer Vision Project:

System Design:
Identifying the problem or need that the computer vision system will address.
Researching and selecting the appropriate algorithms and models for the system.
Designing the architecture of the system, including the data flow, processing pipeline, and interfaces.

Collecting and preprocessing the data, such as images and videos, that the system will use to train and test its models.
Training and testing the models to ensure proper functionality and performance.
Integrating the models into the final system and testing it in real-world scenarios.

Components:
Camera or other image capture device
Image processing and analysis software
Algorithms such as object detection, image classification, and facial recognition
Deep learning frameworks such as TensorFlow and PyTorch
Hardware such as GPUs for training and inference
Datasets for training and testing the models.

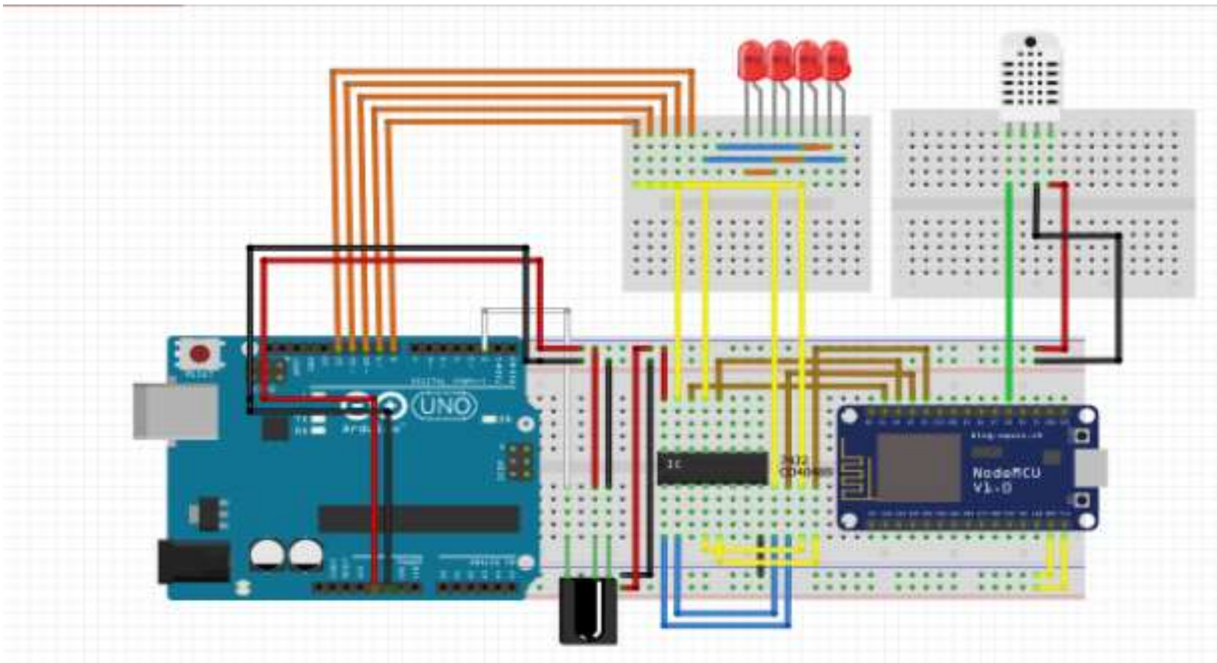
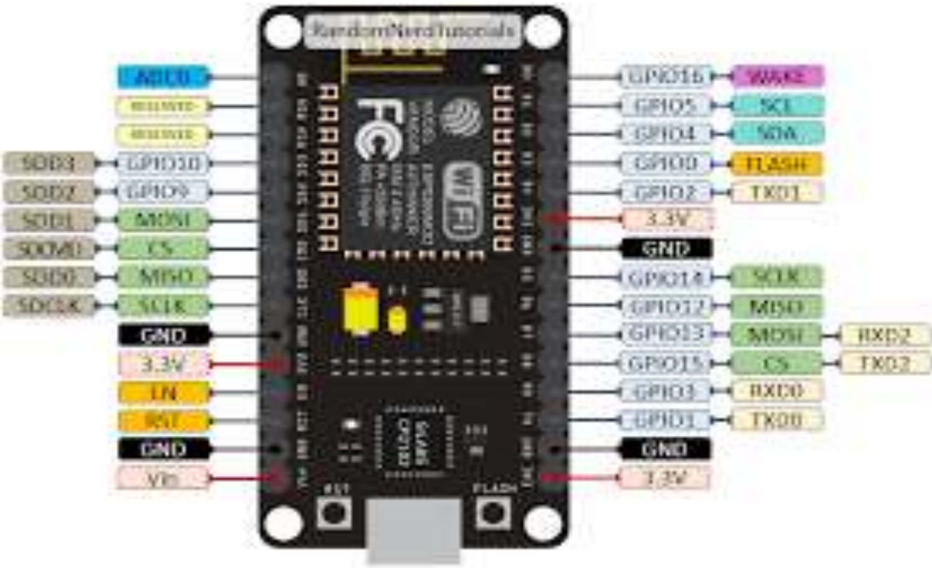


Fig 3.2 Component visiual Circuit diagram of this project

Arduino uno and Node MCU ESP8266 are used as microcontrollers in this project. Arduino Uno is used as a simple microcontroller and Node MCU ESP8266 is used to take advantage of WiFi connectivity. IR receiver sensor is used as remote control receiver. Star-brand remote is used for total control.



.**Fig 3.3 Node mcu esp8266**

The ESP8266 is a low-cost WiFi microcontroller that is often used in Internet of Things (IoT) projects. It is a System on a Chip (SoC) that includes a microcontroller, WiFi transceiver, and several other components on a single chip. The ESP8266 is available in several different versions, including the NodeMCU, which is a development board that includes the ESP8266 and a USB-to-serial converter. The ESP8266 can be programmed using the Arduino IDE or other programming environments. It is often used to connect devices to the Internet and to send and receive data from cloud services. It is also commonly used as a WiFi client to connect to an existing network, or as a WiFi access point to create a new network. [1] [D2]

Fig 3.4 IR Reciver

An infrared (IR) receiver is a device that is used to detect infrared radiation and convert it into an electrical signal. It consists of a photodiode or other type of IR-sensitive semiconductor device and associated circuitry that is designed to detect and amplify the IR signal. IR receivers are commonly used in a variety of applications, including remote controls, burglar alarms, and other devices that use IR signals for communication. They are often used in combination with IR transmitters, which are used to transmit the IR signals. IR receivers are available in a variety of packages and configurations, including surface mount devices (SMDs), through-hole devices, and modules with built-in amplifiers and other components. They can be interfaced with microcontrollers and other electronic devices using a variety of methods, including digital and analog inputs. [D6]

Fig 3.4 IR transmitter Star universal brand remote.

An infrared (IR) transmitter is a device that is used to transmit IR signals. It consists of a light-emitting diode (LED) or other type of IR-emitting device and associated circuitry that is designed to generate and modulate the IR signal. IR transmitters are commonly used in a variety of applications, including remote controls, burglar alarms, and other devices that use IR signals for communication. They are often used in combination with IR receivers, which are used to detect the IR signals. A TV remote control is a common example of a device that uses an IR transmitter. The remote control sends IR signals to the TV, which are detected by an IR receiver on the TV. The signals contain information such as the channel number or the Voltage amplitude level, which the TV uses to control its operation. [D7]

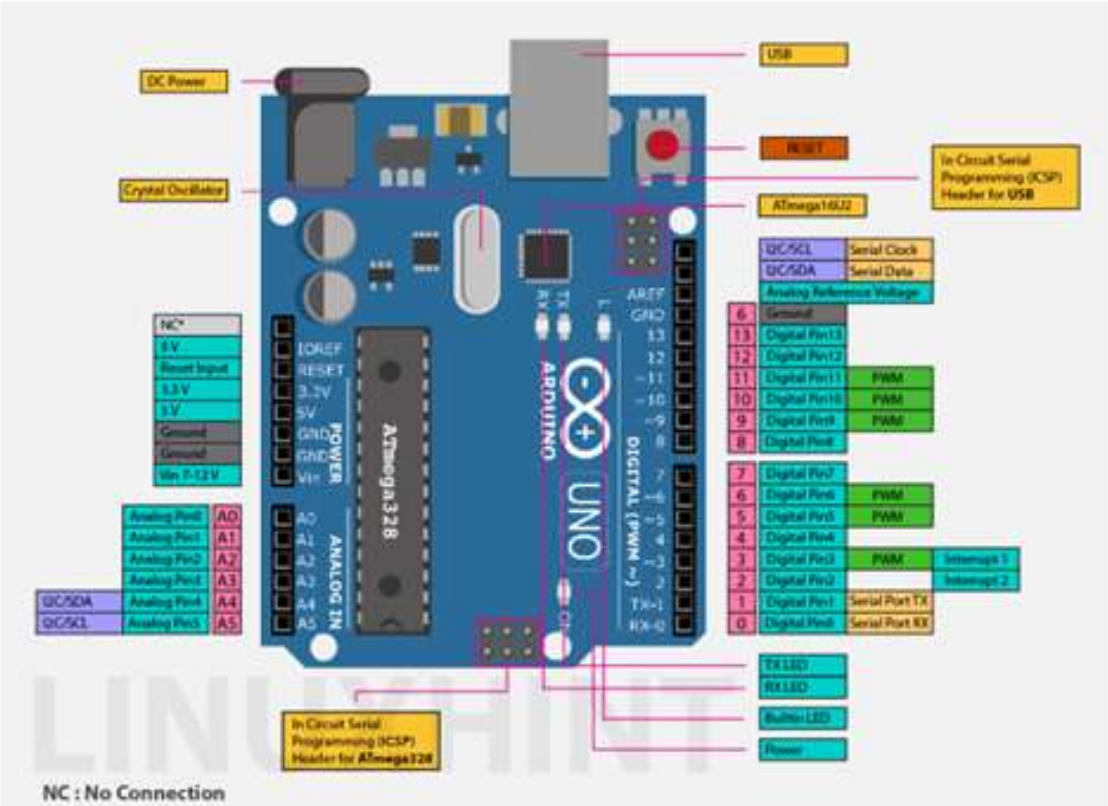


Fig 3.5 Arduino uno r3

Arduino is an open-source hardware platform that consists of a series of microcontroller boards and a software development environment (IDE) for writing and uploading code to the boards. The Arduino Uno is a popular version of the Arduino platform and is based on the ATmega328 microcontroller. The Arduino Uno has 14 digital input/output pins, 6 analog input pins, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It can be powered by a USB connection or by an external power supply. The Arduino Uno can be programmed using the Arduino IDE, which is available for Windows, Mac, and Linux. The Arduino IDE uses a version of C++ and includes a library of pre-written code called "sketches" that can be modified and uploaded to the Arduino Uno. [D1]

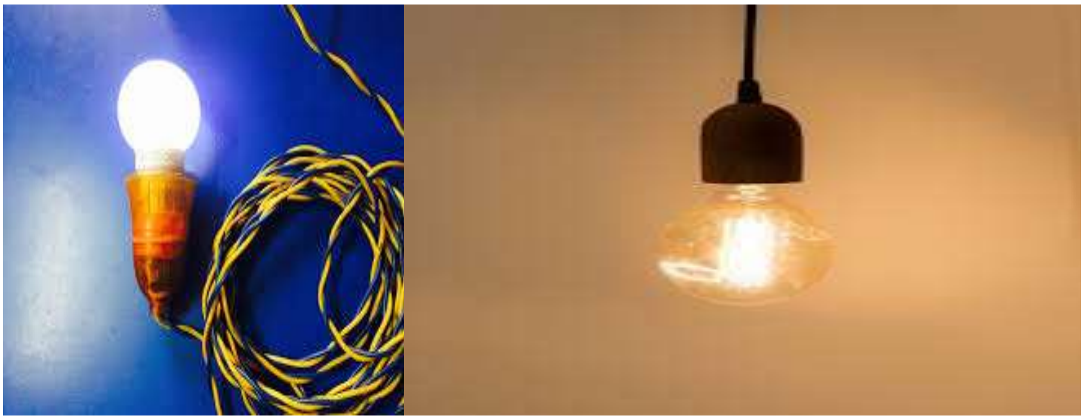


Fig 3.6 Bulb as an Load

Four tungstens were used as loads. Using tungstens balls for ease of display. 220 volt supply provided for operating . A 220V tungsten bulb is a type of incandescent light bulb that operates at a voltage of 220V and uses a tungsten filament as the source of light. Tungsten bulbs are widely used in a variety of applications, including residential and commercial lighting, due to their low cost and good performance. Tungsten bulbs work by heating a thin tungsten filament to a high temperature using an electrical current. The filament then emits light as a result of its high temperature. The intensity of the light produced by a tungsten bulb is directly proportional to the filament temperature, so increasing the filament temperature will increase the brightness of the bulb.

Fig 3.7 RBD Voltage Dimmer

Voltage dimmers are used to control the voltage from a remote location as Voltage dimmer. There is a field ready detector and pwm modulation system. It is mainly composed of zero crossing detector circuit and triac based drive circuits. An RBD voltage dimmer is a type of electronic dimmer that uses a phase-cut technique to control the brightness of a lighting load. It works by switching the power to the load on and off at a high frequency, and by adjusting the ratio of on-time to off-time, the average voltage to the load can be controlled. This in turn allows the brightness of the load to be controlled. RBD dimmers are often used in residential and commercial lighting applications, and are available in a range of different form factors including wall-mounted dimmers, in-line dimmers, and dimmer switches. They are typically compatible with a wide range of lighting loads, including incandescent, halogen, and some types of LED lights [D9]

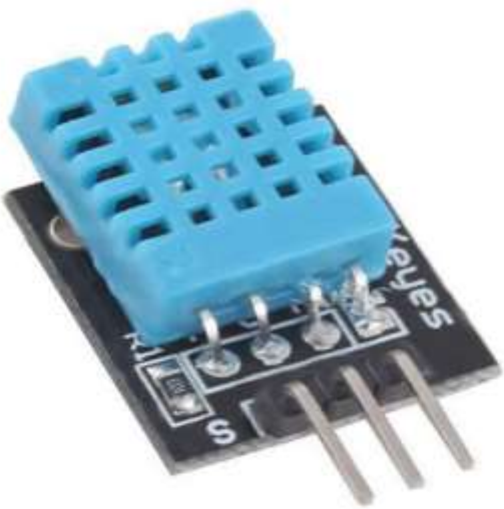


Fig 3.8 DTH11 Sensor

Dth11 sensor is used for temperature and humidity sensing. Basically operates on five volts. Informs the system by providing digital output.
The DHT11 is a low-cost humidity and temperature sensor that is often used in DIY projects. It is available as a standalone sensor or as a module that includes a microcontroller and a 4-pin connector. The DHT11 measures both humidity and temperature and provides a digital output through the 4-pin connector. It has a range of 20-80% relative humidity (RH) and a temperature range of 0-50°C. It has an accuracy of +/-5% RH and +/-2°C. The DHT11 can be interfaced with a microcontroller or a single-board computer such as an Arduino or Raspberry Pi using a 4-pin connector or a set of wires. The data can then be processed and used to control devices or to display the humidity and temperature on a display or online. [D3]

Fig 3.9 The MQ6 Gas Sensor

The MQ6 sensor used as gas sensing. Basically operates on five volts. Informs the system by providing analog output. And there's a variable nobe system on the increase or decrease sensitivity.
The MQ6 is a gas sensor that is commonly used to detect the presence of flammable gases such as propane, butane, and methane. It is available as a standalone sensor or as a module that includes a microcontroller and a 4-pin connector. The MQ6 sensor operates by measuring the electrical resistance of a metal oxide semiconductor (MOS) element, which changes in response to the presence of gas molecules. The sensor outputs an analog voltage that is proportional to the concentration of gas in the air. The MQ6 sensor can be interfaced with a microcontroller or a single-board computer such as an Arduino or Raspberry Pi using a 4-pin connector or a set of wires. The data can then be processed and used to control devices or to display the gas concentration on a display or online. [D4]

Fig 3.10 5v four channel Relay Module

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

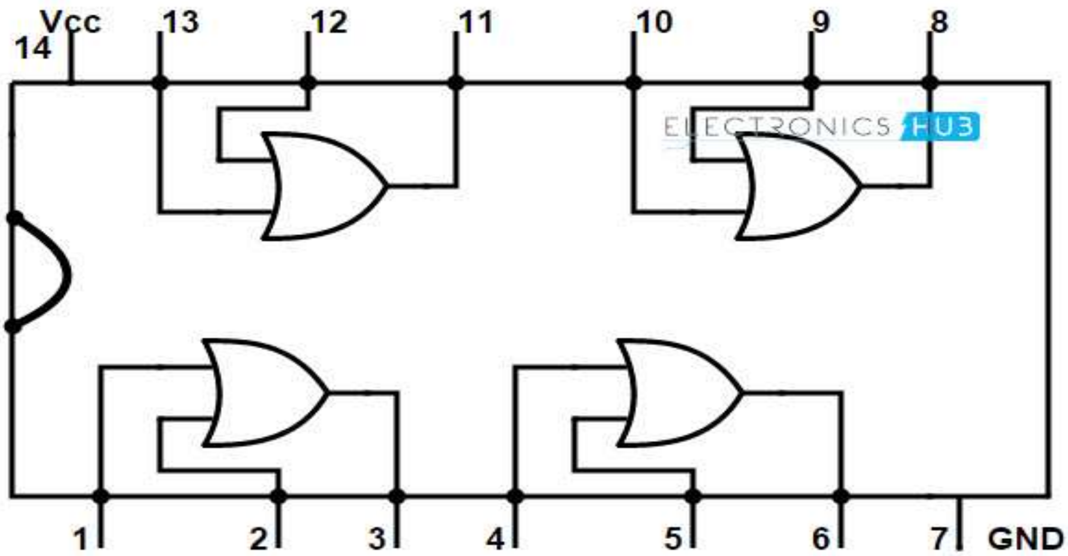


Fig 3.11 Quard OR gate IC 7432

74322 is used as the or gate IC. This Orgate will conduct its operations between remote control and IOT control . The 7432 is a quad 2-input OR gate. It is a digital logic gate that performs an OR operation on two binary inputs. An OR operation returns a 1 if either or both of the inputs are 1, and a 0 if both inputs are 0. The 7432 has four independent 2-input OR gates, each with its own set of inputs and outputs. It is available in a variety of package types, including DIP, SOIC, and SOP. The 7432 can be used in a variety of digital circuits, including logic systems, computer systems, and other electronic systems. It is often used in combination with other logic gates to create more complex circuits. [D8]

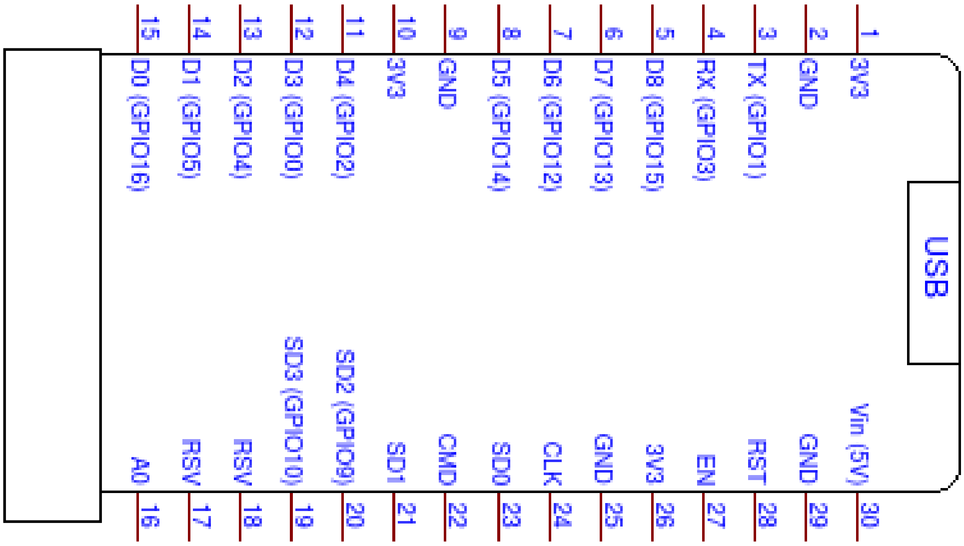


Fig 3.12 Scametric arrangement for Node mcu esp8266

Nodemcu esp8266 is used as microcontroller in IoT control.This is a wifi development board
D0 and D1 pin are used as the output and zero cross detector pins of voltage Dimmer Module .
D7 Pin is used to receive input from the temperature sensor.
Pins number D5,D6,D2,D8 are used for relay controlling. [D2]

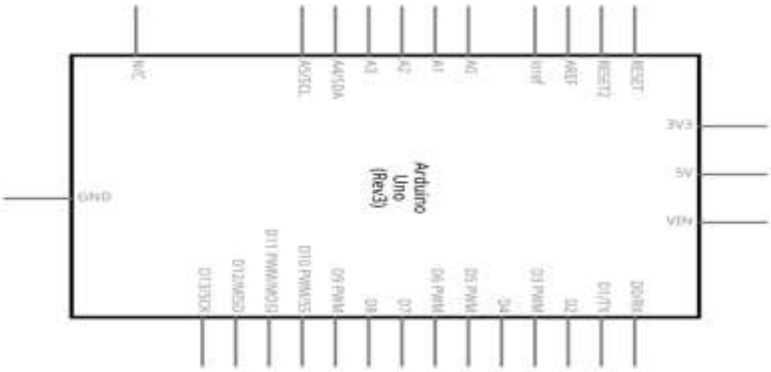


Fig 3.13 Scametric arrangement of arduino uno for manual control

Arduino Uno is used as microcontroller for manual control.Connected to it is the remote control section.
With microcontroller digital 2 number pin is used as receiver pin from ir sensor . Digital Pin 8,10,11,12 is used for IR remote control based relay controlling. [D1]

3.3 Design Specifications. Standards and Constraints

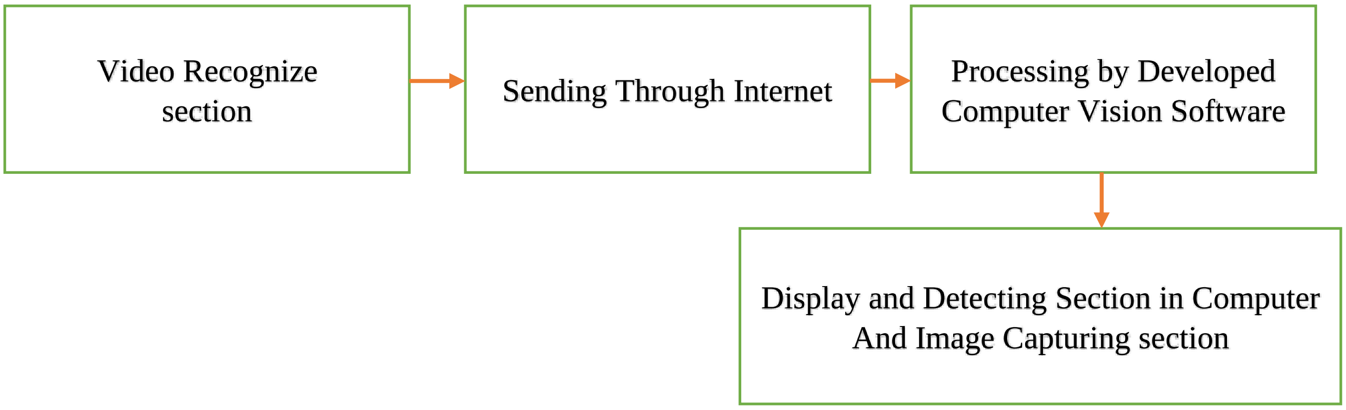


fig 3.14 Block diagram of Computer vision processing.

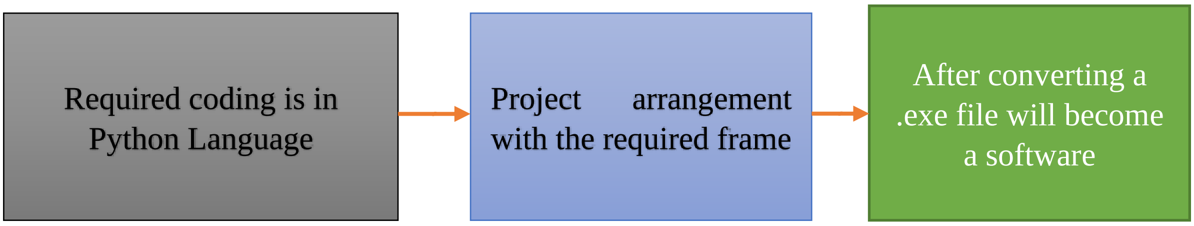
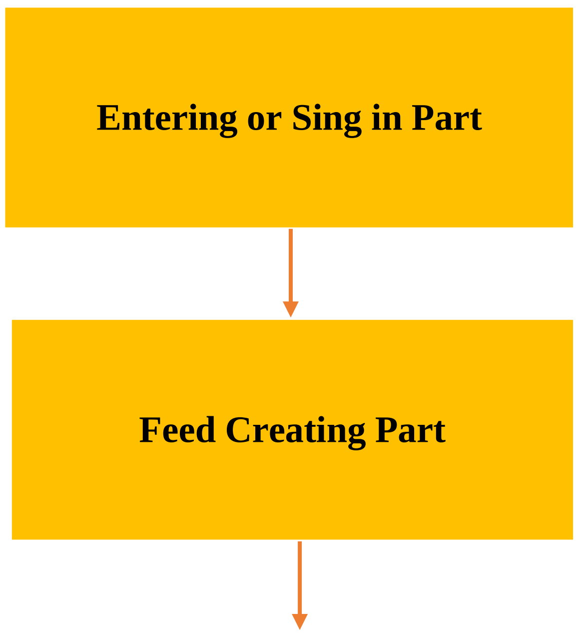


fig 3.15 Developing Block step For Computer vision Software

3.4 Simulation/Experimental Setup

Standards:

The system must comply with relevant industry and regulatory standards for data security and privacy.
The system must comply with relevant standards for image and video processing such as OpenCV.
The system must comply with relevant standards for deep learning frameworks such as TensorFlow and PyTorch.
Constraints:
The system must operate within the constraints of the available budget and resources.
The system must operate within the constraints of the available data and the quality of that data.
The system must operate within the constraints of the available computational resources and power.
The system must operate within the constraints of the legal and ethical issues related to the data processing and usage. [3] [6]



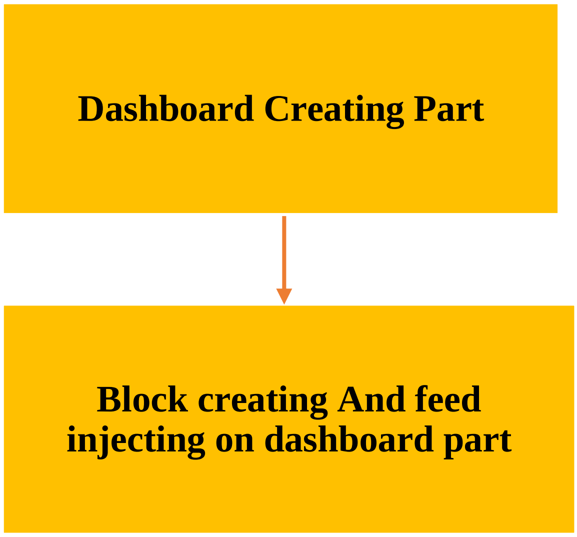


fig 3.16 IOT platform Creating block step on adfruit server .

fig 3.17 IOT platform Creating step 1

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

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

fig 3.18 IOT platform Creating step 2

After completion of signin work. I will make feeds. With the help of this i will send data here from different places and perform display tasks on the sent write.
This will be known as the second part of theIOT block creatinine. Feeds are the core of Adafruit IO. They hold both the data that uploaded and meta-data about the data that sensors push to Adafruit IO. For example, the date and time when it was uploaded. Or, the GPS coordinates where the data came from. Want to share thatr data feed with the world, a group of people, or just a friend? Adafruit IO has integrated feed sharing with fine-tuned privacy controls to allow (and restrict) access to data. [10]

fig 3.19 IOT platform Creating step 3

The third part of the project will be known as Creating Dashboards in service. Through the dashboard, my returned data can be displayed floating and I will have a system of virtual input and output. Dashboards are a feature integrated into Adafruit IO which allow that to chart, graph, gauge, log, and visualize thatr data. it can view thatr dashboards from anywhere in the world. [11]

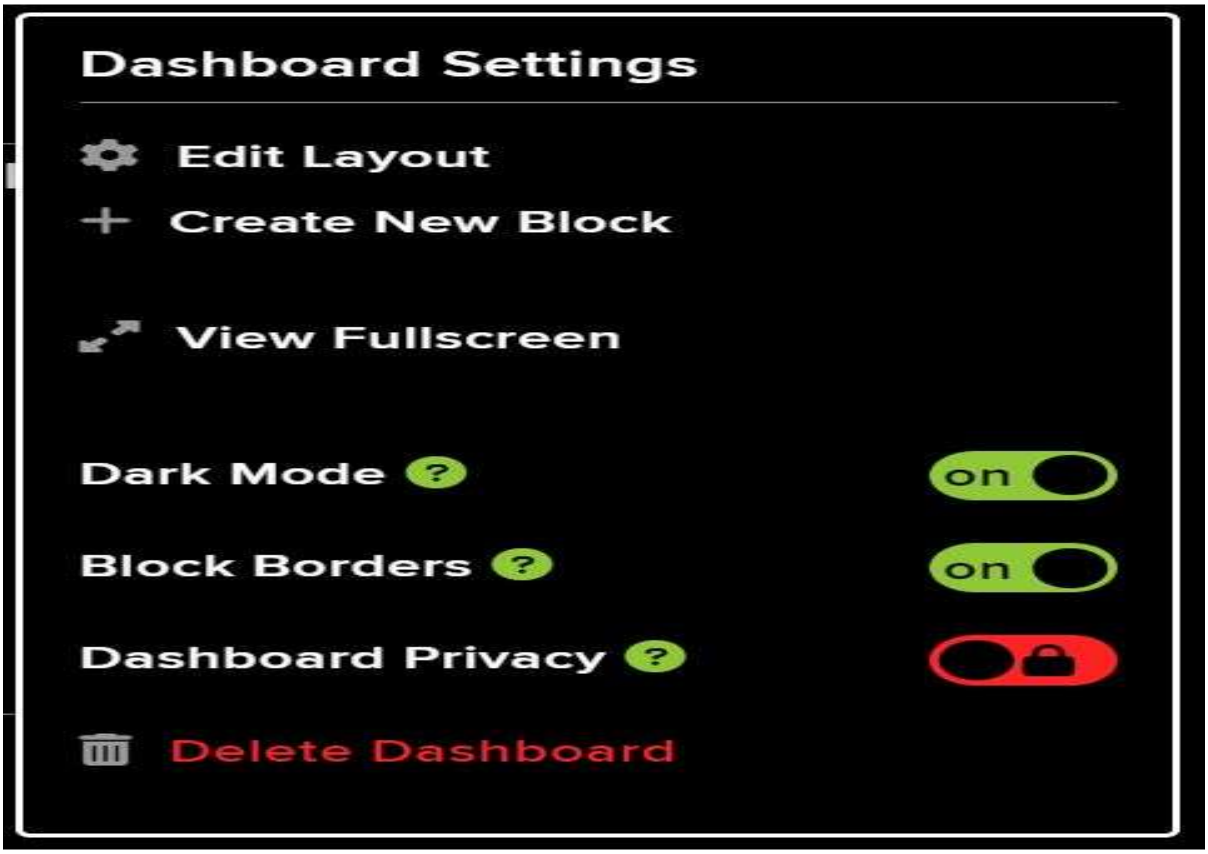


fig 3.20 IOT platform Creating step 3

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

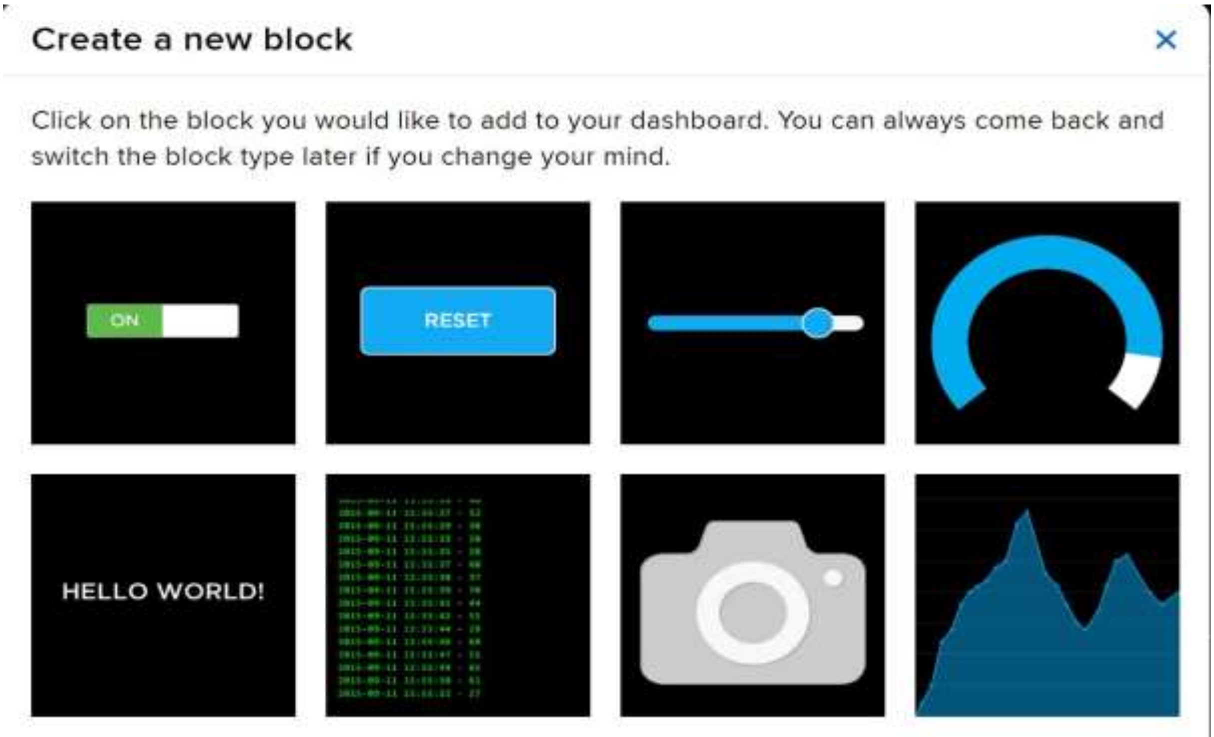


fig 3.21 IOT platform Creating step 4

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

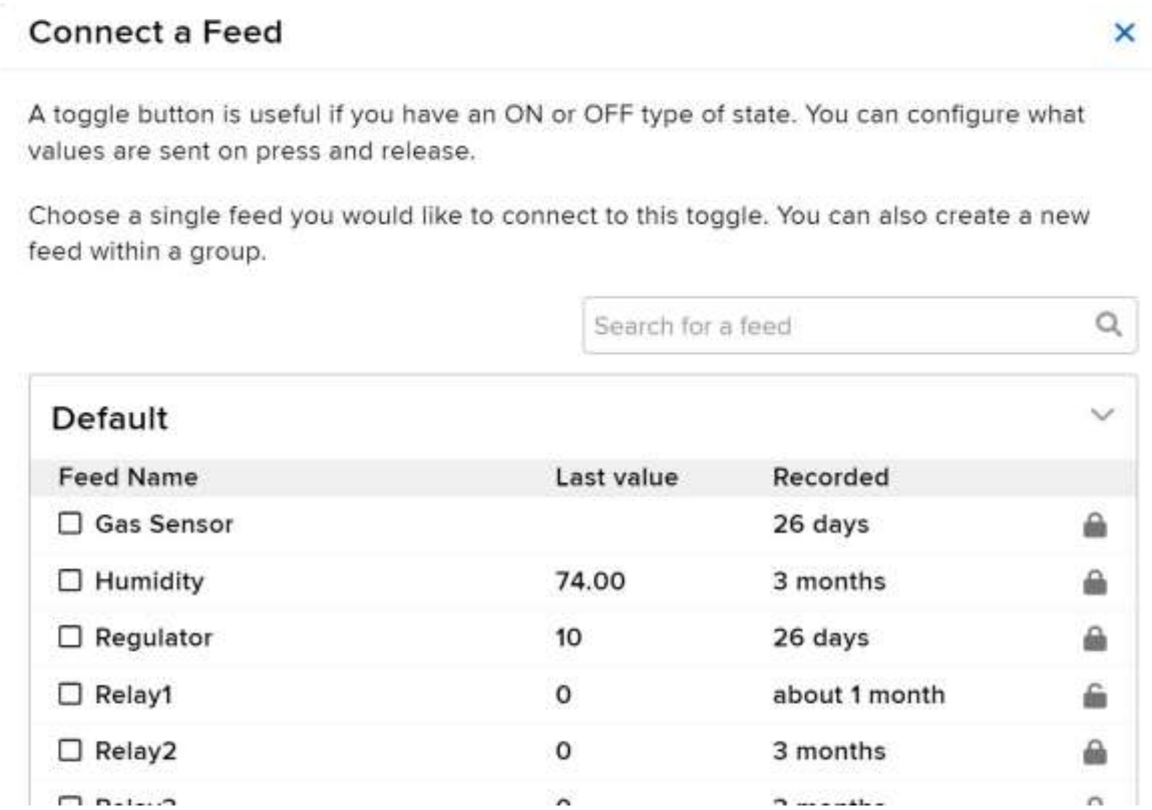


fig 3.22 IOT platform Creating step 5

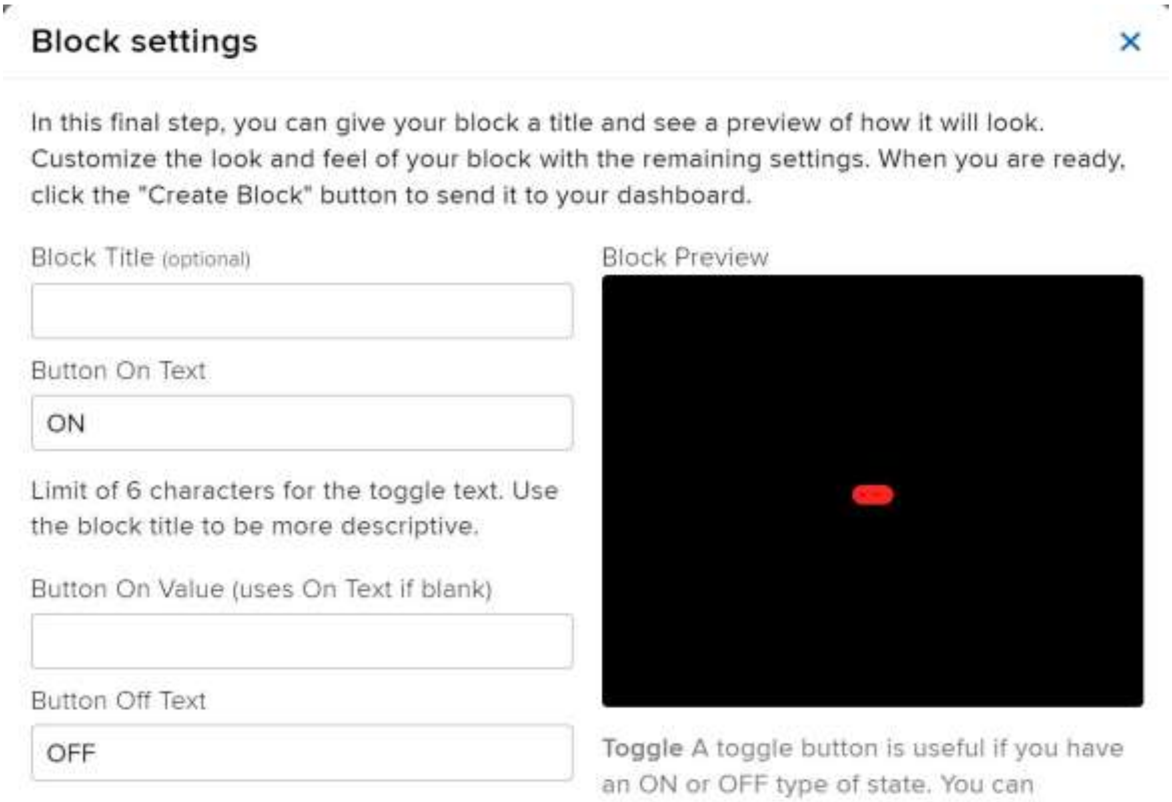


fig 3.23 IOT platform Creating step 6

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

Gesture movement based Voltage Amplitude Controller Making Steps:

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

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

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

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

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

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

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

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

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

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

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

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

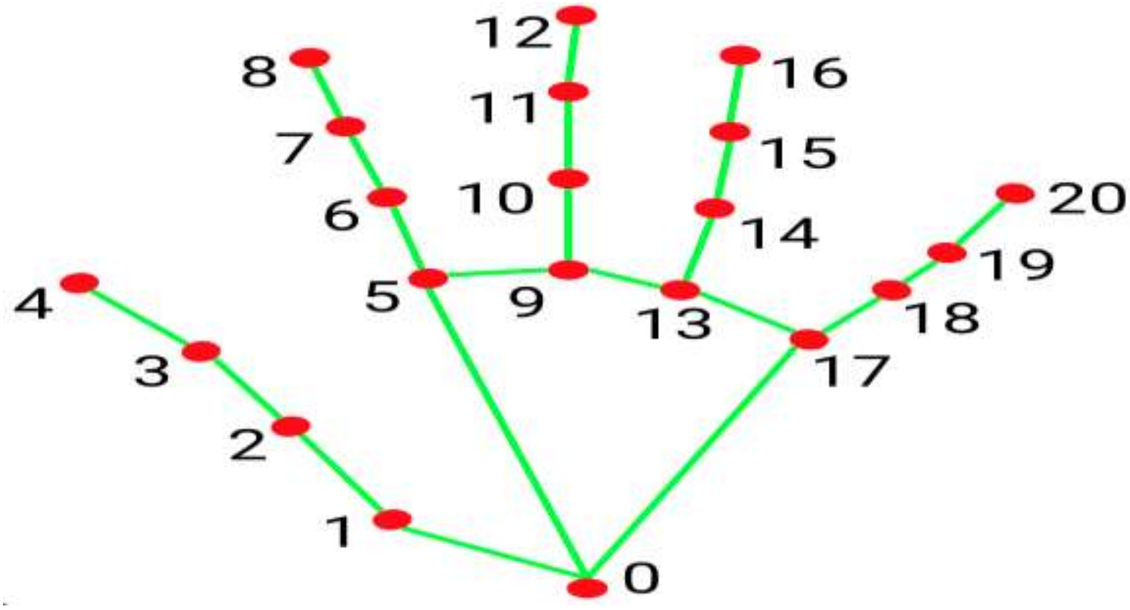


Fig 3.24 21 Hands Landmarks point [35]

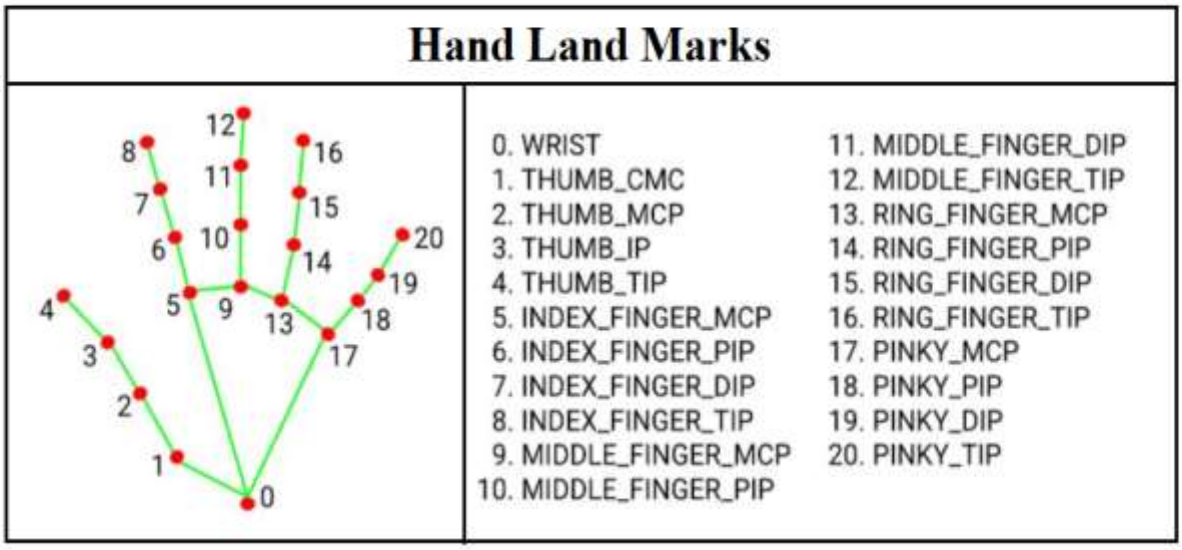


Fig 3.21 25 Hands Landmarks point Specification [35]

Customize the pipeline: I can alter the configuration file for the hand detection pipeline or provide Ir own code to handle the output. For instance, I may control a user interface or work with 3D objects using the hand detection information.
The shape and location of the hand can be described using hand landmarks, which are certain places on the hand like the fingertips, knuckles, and wrist. The hand bounding boxes produced by the hand detection model are used in MediaPipe to estimate hand landmarks.
A machine learning model that was trained on a sizable dataset of hand photographs is used to calculate the hand landmarks. The image plane's landmarks' 2D coordinates are

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

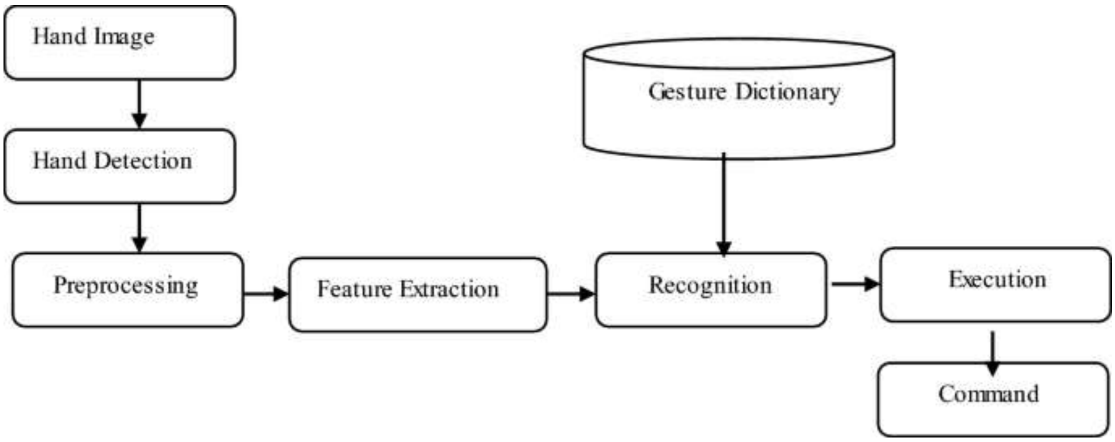


Fig 3.26 Gesture Control Steps 1 [35] [30] [32]

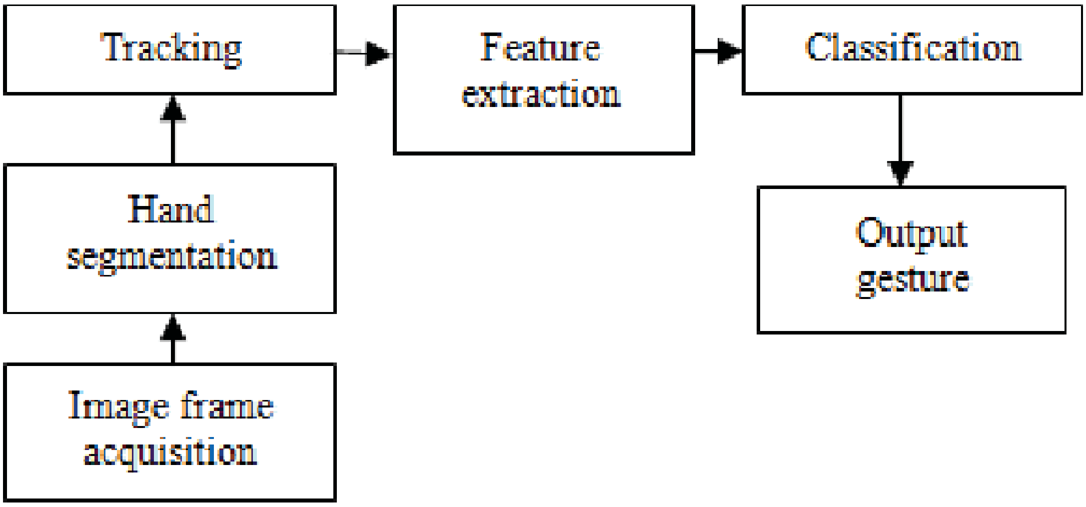


Fig 3.27 Gesture Control Steps 2 [35] [30] [32]

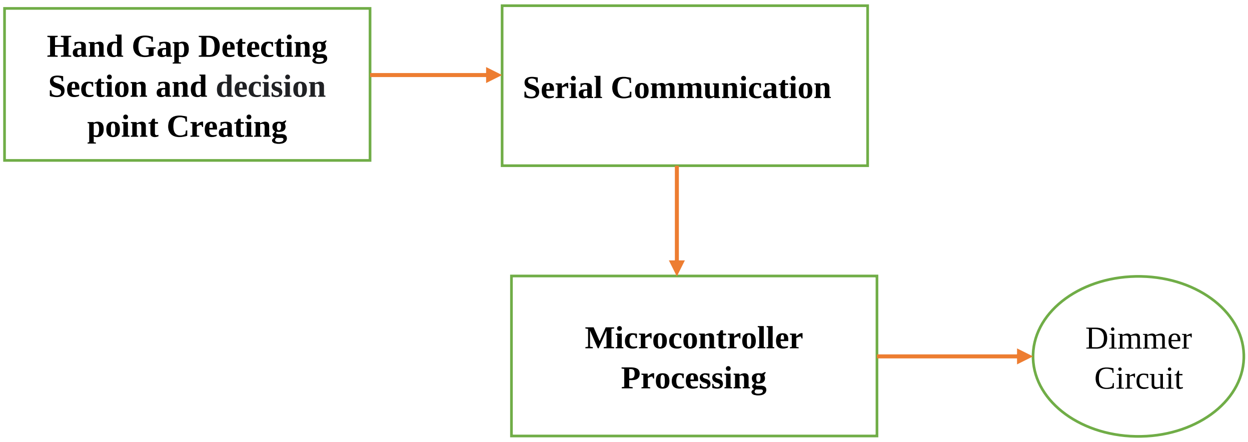


Fig 3.28 Block steps of Gesture Control Development

[35]

Chapter 4

RESULTS AND DISCUSSIONS

4.1 Results

Improved monitoring and control of physical devices and systems, such as temperature and humidity in a warehouse.

Increased efficiency and automation of processes, such as lighting and HVAC systems in a building.

Increased data collection and analysis, which can be used to make data-driven decisions and improve overall performance.

Increased connectivity and communication between devices, systems, and people.

Improved security and reliability of the system.

Improved perception and understanding of visual data, such as images and videos, by the system.

Increased accuracy and speed of tasks such as object detection, image classification, and facial recognition

Increased automation and efficiency of processes that rely on visual data analysis.

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

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

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

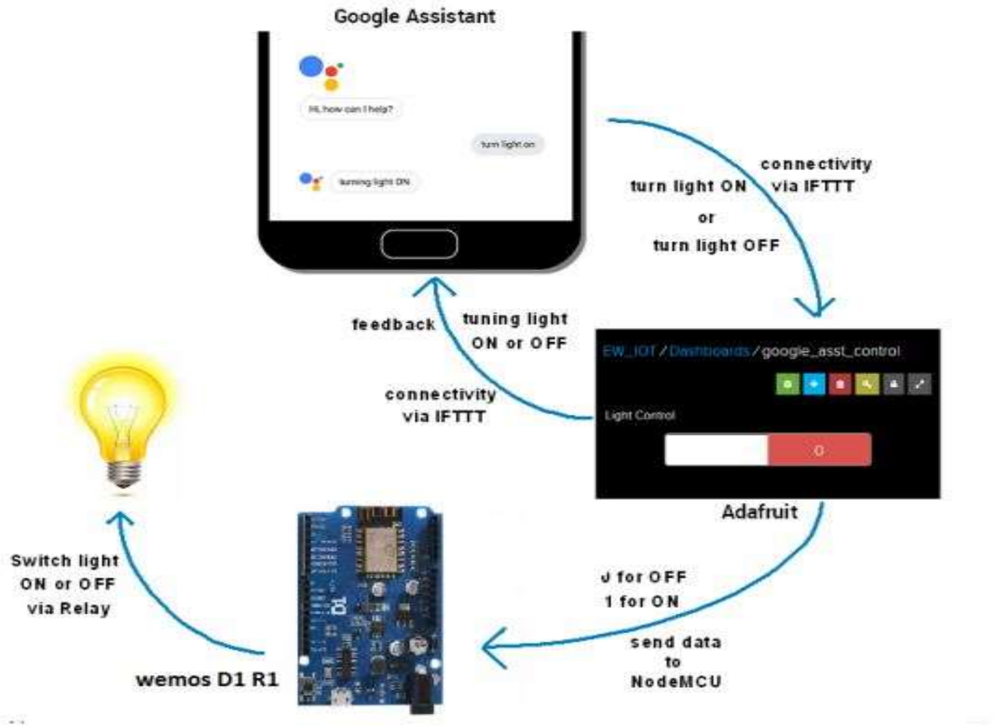


Fig2:Flow Diagram

Fig: Flow Diagram [26]

Fig 4.1 IFTTT Service and Appleates1

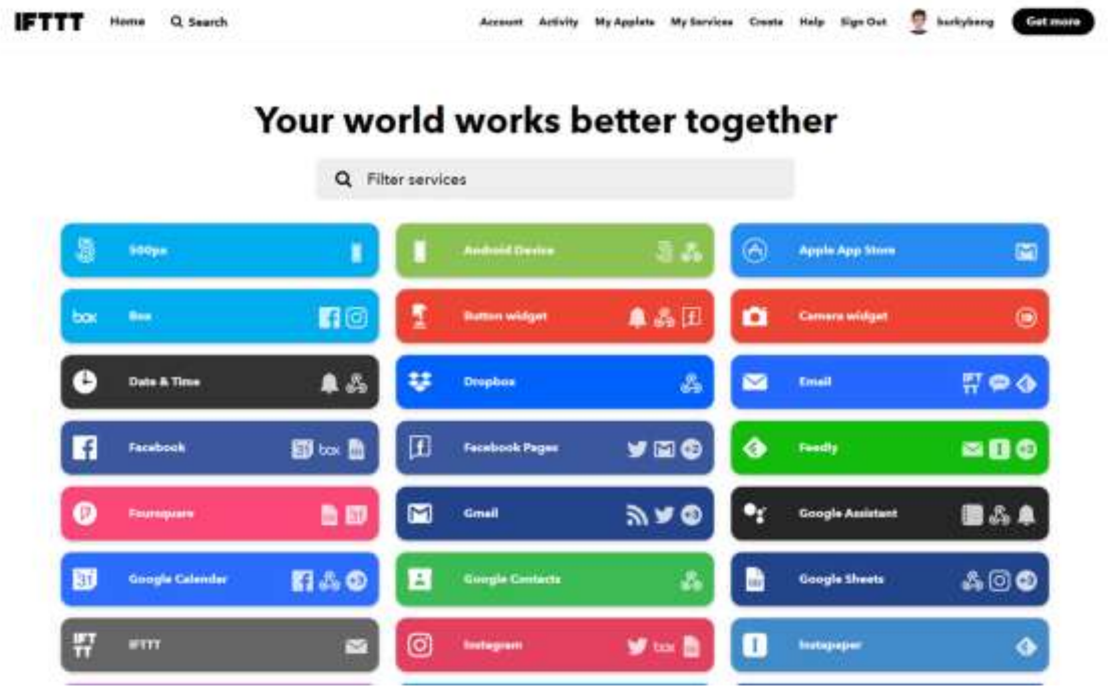


Fig 4.2 IFTTT Service and Appleates2

The core controlling of the IOT project part is a web server called IFTTT. By using this server. Multidimensional control is possible by using only this Server without changing the original source code. Amazon , Alexa, Android Device, Android ,SMS, Discord, Dropbox, RSS Feed, Fitbit, Gmail. It is possible to do various other things along with these. [26]

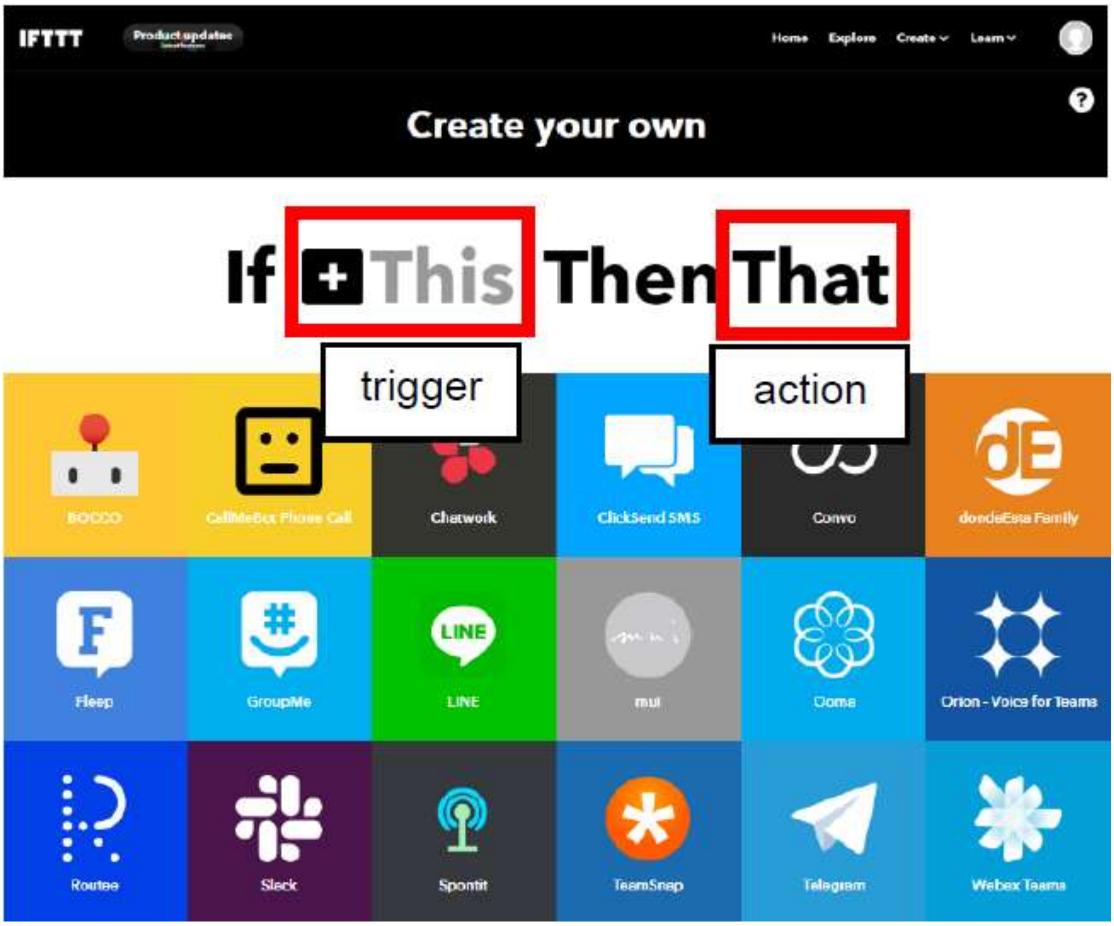


Fig 4.3 IFTTT Service and Appleates3

With this, a different web server can set a control based on a data point and take action based on it. It has more than 1000 services and applications. How to connect this IFTTT server to This project so that all types of controlling can be done very easily. Especially the data collection work can be done very well. For example can be said. Controlling based on any type of data can send alert messages to the user through mail. A lot more can be done like this project. **[26]**

Fig 4.4 Self-developed apps for use in projects

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

Fig 4.5 Computer vision or video visualization protocols

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

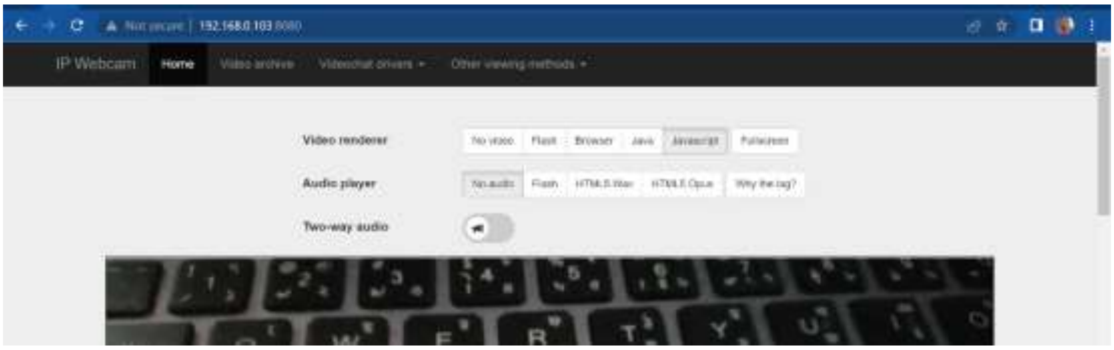


Fig 4.6 IP link http://192.168.0.103:8080/ based computer vision or video visualization through Internet.

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

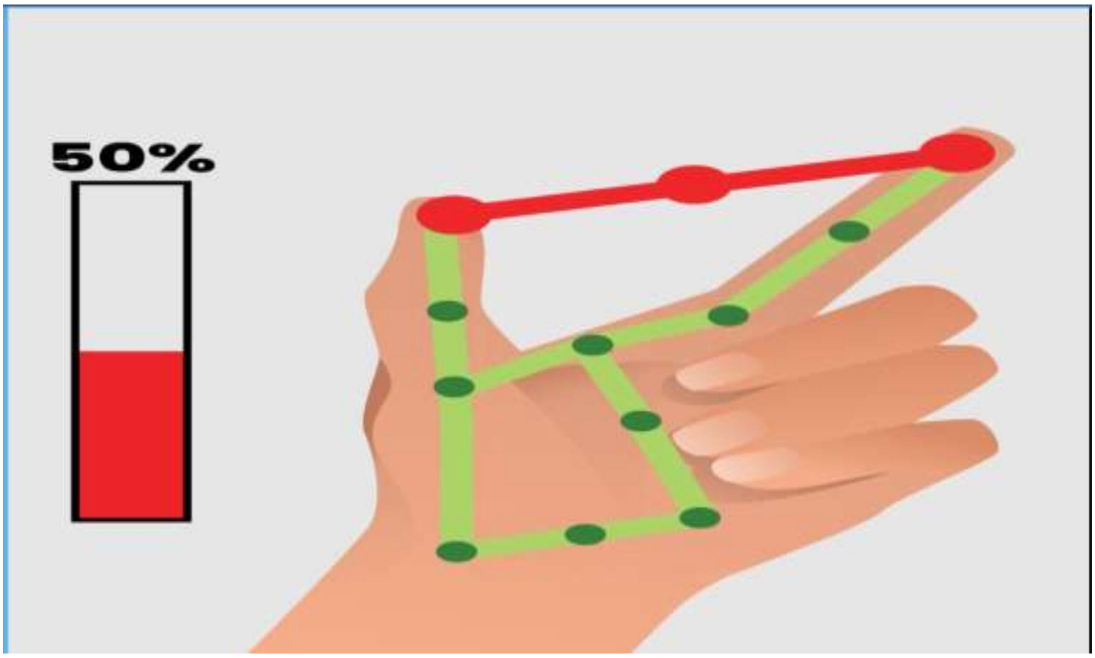


Fig 4.8 Index and thumb Finger Gesture Movement based phase angle control part .

Discussions for IoT (Internet of Things) Project:

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

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

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

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

Discussions for Computer Vision Project:

Data Quality and Quantity: The quality and quantity of data used to train and test computer vision models are crucial factors in determining the performance of the system. Ensuring that the data is accurate, diverse and sufficient is crucial to ensure the system's ability to generalize and adapt to new scenarios.

Algorithm Selection and Fine-tuning: The choice of algorithms and models used in a computer vision system is critical to its performance. It is important to carefully evaluate the capabilities and limitations of different approaches and fine-tune the models to optimize their performance for the specific task at hand.

Bias and Fairness: Computer vision systems are prone to biases if not designed and trained carefully. These biases can lead to errors and inaccuracies in the system's performance, particularly when analyzing images and videos of certain groups of people. It is important to consider these issues and take steps to mitigate any potential biases in the system.

Real-world deployment and maintenance: The deployment of computer vision systems in real-world scenarios requires a robust and reliable infrastructure to ensure that the system is able to function in different environments, handle failures and adapt to changing conditions. It also requires maintenance, monitoring and updating the system to ensure its performance and security. **[9] [7] [6]**

CHAPTER 5

PROJECT MANAGEMENT

5.1 Task, Schedule and Milestones

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

Research and Analysis (Week 1-4)

Task: Research current IoT and computer vision solutions

Schedule: 4 weeks

Milestone: Completion of research and analysis report

Design and Development (Week 5-12)

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

Schedule: 8 weeks

Milestone: Completion of the design and development of the platform

Computer Vision Algorithms Development (Week 5-8)

Task: Develop computer vision algorithms and integrate with the platform

Schedule: 4 weeks

Milestone: Completion of computer vision algorithms development and integration

Deployment and Testing (Week 9-12)

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

Schedule: 4 weeks

Milestone: Completion of the deployment and testing phase

Data Analytics and Visualization (Week 13-16)

Task: Develop data analytics and visualization tools

Schedule: 4 weeks

Milestone: Completion of data analytics and visualization tools

Launch and Maintenance (Week 17-20)

Task: Launch the platform to the public

Schedule: 4 weeks

Milestone: Completion of the launch and maintenance phase.

5.2 Resources and Cost Management

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

There are several online based robotics shops in Bangladesh of which Robotics BD is one of them.
From these I collected most of the material in This project .

5.3 Lesson Learned

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

CHAPTER 6

IMPACT ASSESSMENT OF THE PROJECT

6.1 ECONOMICAL, SOCIETAL AND GLOBAL IMPACT

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

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

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

6.2 Environmental and Ethical Issues

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

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

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

Ethical Issues:

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

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

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

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

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

6.3 Utilization of Existing Standards

Utilization of Existing Standards:

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

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

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

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

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

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

CHAPTER 7

CONCLUSIONS AND RECOMMENDATIONS

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

Our key findings were learn IOT and apply knowledge apply of electronics.
These results suggest that future applications.
Overall, this project has demonstrated the importance of this type arrangement.

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

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

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

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

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

REFERENCES

1. HTTPS://WWW.RESEARCHGATE.NET/PUBLICATION/337656615_INTERNET_OF_THINGS_AND_NODEMCU_A_REVIEW_OF_USE_OF_NODEMCU_ESP8266_IN_IOT_PRODUCTS
2. HTTPS://IEEEXPLORE.IEEE.ORG/DOCUMENT/8502562
3. HTTPS://IEEEXPLORE.IEEE.ORG/DOCUMENT/8385294
4. HTTPS://WWW.IRJET.NET/ARCHIVES/V7/I3/IRJET-V7I3728.PDF
5. LABS.PT/DOCS/ASSESSING_THE_ESP8266_WIFI_MODULE_FOR_THE_INTERNET_OF_THINGS/1438/VIEW.PDF
6. HTTPS://EN.WIKIPEDIA.ORG/WIKI/MOVING_OBJECT_DETECTION#:~:TEXT=MOVING%20OBJECT%20DETECTION%20IS%20A,ANY%20MOVING%20OBJECT%20IS%20DETECTED.
7. HTTPS://WWW.RESEARCHGATE.NET/PUBLICATION/318598502_A_MOVING_OBJECT_RECOGNITION_USING_VIDEO_ANALYTICS
8. HTTPS://WWW.RESEARCHGATE.NET/PUBLICATION/221127238_MOVING_OBJECT_DETECTION_UNDER_FREE-MOVING_CAMERA
9. HTTPS://ARCHIVES.PALARCH.NL/INDEX.PHP/JAE/ARTICLE/VIEW/7766/7295
10. HTTPS://CDN-LEARN.ADAFRUIT.COM/DOWNLOADS/PDF/WELCOME-TO-ADAFRUIT-IO.PDF
11. HTTPS://CDN-LEARN.ADAFRUIT.COM/DOWNLOADS/PDF/ADAFRUIT-IO-HUB-WITH-THE-ADAFRUIT-FUNHOUSE.PDF
12. HTTPS://IOTDESIGNPRO.COM/ESP8266-PROJECTS
13. HTTPS://WWW.GOOGLE.COM
14. HTTPS://IO.ADAFRUIT.COM
15. HTTPS://IFTTT.COM/EXPLORE
16. HTTPS://CREATE.ARDUINO.CC
17. HTTPS://STORE.ROBOTICSBD.COM/
18. HTTPS://WWW.THATTUBE.COM/
19. HTTPS://WWW.ARDUINO.CC/EN/SOFTWARE/
20. HTTPS://WWW.ROKOMARI.COM/BOOK
21. HTTPS://LASTMINUTEENGINEERS.COM/ESP8266-NODEMCU-ARDUINO-TUTORIAL/
22. HTTPS://LASTMINUTEENGINEERS.COM/ELECTRONICS/ARDUINO-PROJECTS/
23. HTTPS://WWW.GOOGLE.COM/SEARCH?Q=NODE+MCU+ESP8266+PIN+DIAGRAM+D12345&TBM=ISCH&VED=2AHUKEWJU77H1PRZ8AHUXHOYKHRM1DVKQ2-CCEGQIABAA&OQ=NODE+MCU+ESP8266+PIN+DIAGRAM+D12345&GS_LCP=CGNPBWCQAZOECAAQQZOECAAQHLDMA1JBHWDXIMGACAB4AIABIQGIAF0HKGEDMC44MAEAOAEBQGELZ3DZLXDPEI1PBWFAAQE&SCLIENT=IMG&EI=KFY8Y663D7GIMGEZ6RRIDW&BIH=619&BIW=1280
24. HTTPS://WWW.THATTUBE.COM/@MURTAZASWORKSHOP
25. HTTPS://WWW.THATTUBE.COM/@JHANKARMAHBUB
26. HTTPS://WWW.IRJET.NET/ARCHIVES/V6/I3/IRJET-V6I3951.PDF

27. HTTPS://ELECTROPEAK.COM/LEARN/SPEAK-TO-ARDUINO-CONTROL-ALL-PARTS-BY-GOOGLE-ASSISTANT/
28. HTTPS://MEDIAPIPE.DEV/
29. **HTTPS://WWW.RESEARCHGATE.NET/PUBLICATION/329039460_RESEARCH_ON_GESTURE_RECOGNITION_METHOD_BASED_ON_COMPUTER_VISIO**
30. **HTTPS://IEEEXPLORE.IEEE.ORG/DOCUMENT/8634348**
31. **HTTPS://IEEEXPLORE.IEEE.ORG/DOCUMENT/9622242**
32. **CONFERENCES.ORG/ARTICLES/MATECCONF/ABS/2018/91/MATECCONF_EITCE2018_03042/MATECCONF_EITCE2018_03042.HTML**
33. **HTTPS://DEVELOPERS.GOOGLE.COM/MEDIAPIPE/SOLUTIONS/VISION/GESTURE_RECOGNIZER/PYTHON**
34. **HTTPS://WWW.HACKSTER.IO/AS4527/VOLUME-CONTROL-USING-HAND-GESTURE-USING-PYTHON-AND-OPENCV-7AAB9F**
35. HTTPS://CINELIGHT.COM/TUNGSTEN-LIGHT-ACCESSORIES/LIGHT-DIMMER-1000-WATTS
36. HTTPS://WWW.RESEARCHGATE.NET/PUBLICATION/351999190_AN_IOT-BASED_SMART_HOME_AUTOMATION_SYSTEM
37. HTTPS://IEEEXPLORE.IEEE.ORG/DOCUMENT/9301078
38. HTTPS://WWW.RESEARCHGATE.NET/PUBLICATION/324417835_DYNAMIC_HAND_GESTURE_RECOGNITION_USING_COMPUTER_VISION_AND_NEURAL_NETWORKS
39.

APPENDIX A

TURNITIN REPORT

Include here the 1st page of Turnitin Report

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

APPENDIX B

COMPLEX ENGINEERING PROBLEM SOLVING AND ENGINEERING ACTIVITIES

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

Complex Engineering Problems (P) Solving		
	Attributes	Statement from students
P1	Range of resources	
P2	Level of interaction	
P3	Innovation	
P4	Consequences of society and environment	
P5	Familiarity	
P6	Extent of stakeholder involvement and conflicting requirements	
P7	Interdependence	

Provide thatr 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	
A2	Range of conflicting requirements	
A3	Depth of analysis required	
A4	Familiarity of issues	
A5	Extent of applicable codes	

APPENDIX C

PROGRAM CODE

```
#include <RBDdimmer.h> //
#include "DHT.h"
DHT dht2(D7,DHT11);
//#define USE_SERIAL  SerialUSB //Serial for boards whith USB serial port
#define USE_SERIAL  Serial
#define outputPin  D0
#define zerocross  D1 // for boards with CHANGEBLE input pins

dimmerLamp dimmer(outputPin, zerocross); //initialase port for dimmer for ESP8266, ESP32, Arduino due boards
//dimmerLamp dimmer(outputPin); //initialase port for dimmer for MEGA, Leonardo, UNO, Arduino M0, Arduino Zero
int outVal = 0;
#include <ESP8266WiFi.h>
#include "Adafruit_MQTT.h"
#include "Adafruit_MQTT_Client.h"
#define Relay1            D5 //
#define Relay2            D6
#define Relay3            D2
#define Relay4            D8
#define WLAN_SSID       "Rafsan"            // Thatr SSID
#define WLAN_PASS       "diu123456"        // Thatr password

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

#define AIO_SERVER      "io.adafruit.com"
#define AIO_SERVERPORT  1883                   // use 8883 for SSL
#define AIO_USERNAME    "RAFSAN"            // Replace it with thatr username
#define AIO_KEY         "aio_lyKx644pDl7wnblQpS0VdG9Y0auL"  // Replace with thatr Project Auth Key

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

// Create an ESP8266 WiFiClient class to connect to the MQTT server.
WiFiClient client;
// or... use WiFiFlientSecure for SSL
//WiFiClientSecure client;
// Setup the MQTT client class by passing in the WiFi client and MQTT server and login details.
Adafruit_MQTT_Client mqtt(&client, AIO_SERVER, AIO_SERVERPORT, AIO_USERNAME, AIO_KEY);

/****************************** Feeds ***************************************/
```

```
// Setup a feed called 'onoff' for subscribing to changes.
Adafruit_MQTT_Subscribe Light1 = Adafruit_MQTT_Subscribe(&mqtt, AIO_USERNAME"/feeds/Relay1"); // FeedName
Adafruit_MQTT_Subscribe Light2 = Adafruit_MQTT_Subscribe(&mqtt, AIO_USERNAME "/feeds/Relay2");
Adafruit_MQTT_Subscribe Light3 = Adafruit_MQTT_Subscribe(&mqtt, AIO_USERNAME "/feeds/Relay3");
Adafruit_MQTT_Subscribe Light4 = Adafruit_MQTT_Subscribe(&mqtt, AIO_USERNAME "/feeds/Relay4");
Adafruit_MQTT_Subscribe regulat = Adafruit_MQTT_Subscribe(&mqtt, AIO_USERNAME "/feeds/Regulator"); //new
Adafruit_MQTT_Publish photocell = Adafruit_MQTT_Publish(&mqtt, AIO_USERNAME "/feeds/Temperature");
Adafruit_MQTT_Publish neww = Adafruit_MQTT_Publish(&mqtt, AIO_USERNAME "/feeds/Humidity");

void MQTT_connect();

void setup() {
Serial.begin(115200);

pinMode(Relay1, OUTPUT);
pinMode(Relay2, OUTPUT);
pinMode(Relay3, OUTPUT);
pinMode(Relay4, OUTPUT);
dimmer.begin(NORMAL_MODE, ON); //dimmer initialisation: name.begin(MODE, STATE)
//pinMode(sensor,INPUT);
digitalWrite(D6,LOW);
digitalWrite(D5,LOW);
digitalWrite(D2,LOW);
digitalWrite(D8,LOW);
// Connect to WiFi access point.
Serial.println(); Serial.println();
Serial.print("Connecting to ");
Serial.println(WLAN_SSID);
WiFi.begin(WLAN_SSID, WLAN_PASS);
while (WiFi.status() != WL_CONNECTED) {
delay(500);
Serial.print(".");
}
Serial.println();

Serial.println("WiFi connected");
Serial.println("IP address: ");
Serial.println(WiFi.localIP());
// Setup MQTT subscription for onoff feed.
mqtt.subscribe(&Light1);
mqtt.subscribe(&Light3);
mqtt.subscribe(&Light2);
```

```
mqtt.subscribe(&Light4);
mqtt.subscribe(&regulat); //n
//mqtt.Publish(&neww); //n
}

void loop() {

MQTT_connect();

Adafruit_MQTT_Subscribe *subscription;
while ((subscription = mqtt.readSubscription(20000))) {
if (subscription == &Light1) {
Serial.print(F("Got: "));
Serial.println((char *)Light1.lastread);
int Light1_State = atoi((char *)Light1.lastread);
digitalWrite(Relay1, Light1_State);

}
if (subscription == &Light2) {
Serial.print(F("Got: "));
Serial.println((char *)Light2.lastread);
int Light2_State = atoi((char *)Light2.lastread);
digitalWrite(Relay2, Light2_State);
}
if (subscription == &Light3) {
Serial.print(F("Got: "));
Serial.println((char *)Light3.lastread);
int Light3_State = atoi((char *)Light3.lastread);
digitalWrite(Relay3, Light3_State);
}
if (subscription == &Light4) {
Serial.print(F("Got: "));
Serial.println((char *)Light4.lastread);
int Light4_State = atoi((char *)Light4.lastread);
digitalWrite(Relay4, Light4_State);

}

if (subscription == &regulat) {
Serial.print(F("Got: "));
Serial.println((char *)regulat.lastread);
int regulat_State = atoi((char *)regulat.lastread);
outVal = regulat_State;
dimmer.setPower(outVal); // name.setPower(0%-100%)

}

}
//int force_value = analogRead(sensor); // readddd
```

```
float temperature  = dht2.readTemperature( );/* Get humidity value */ //n
if (! photocell.publish(temperature)) {
//Serial.println(F("Failed"));
} else {
//Serial.println(F("OK!"));
}
//delay(100);
float humidity  =     dht2.readHumidity();/* Get temperature value */ //n
if (! neww.publish(humidity)) {
//Serial.println(F("Failed"));
} else {
//Serial.println(F("OK!"));
}
}

void MQTT_connect() {
int8_t ret;

// Stop if already connected.
if (mqtt.connected()) {
return;
}

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

uint8_t retries = 3;

while ((ret = mqtt.connect()) != 0) { // connect will return 0 for connected
Serial.println(mqtt.connectErrorString(ret));
Serial.println("Retrying MQTT connection in 5 seconds...");
mqtt.disconnect();
delay(5000);  // wait 5 seconds
retries--;
if (retries == 0) {
// basically die and wait for WDT to reset me
while (1);
}
}
Serial.println("MQTT Connected!");

}
```

Fig A1: Code for IOT Control

```
#include <IRremote.h>
```

```
int RECV_PIN = 2;
IRrecv irrecv(RECV_PIN);
decode_results results;
int c = 0 ;
int state1 = 1;
int state2 = 1;
int state3 = 1;
int state4 = 1;
int state5 = 1;
int led1 = 8;
int led3 = 10;
int led4 = 11;
int led5 = 12;
void setup()
{
  Serial.begin(9600);
  irrecv.enableIRIn(); // Start the receiver
  pinMode(led1, OUTPUT);
  pinMode(led3, OUTPUT);
  pinMode(led4, OUTPUT);
  pinMode(led5, OUTPUT);
}

void loop() {
  //Serial.println(c);
  //38863BD2 power
  //38863BE0 1
  //38863BD0 2
  //38863BF0 3
  //38863BC8 4
  //38863BF2 up
  //38863BFA dwn

  if (irrecv.decode(&results)) {
    Serial.println(results.value, HEX);
    irrecv.resume(); // Receive the next value

    // all on off
    if (results.value == 0x38863BD2)
    {
      digitalWrite(led1, state2);
      digitalWrite(led5, state2);
      digitalWrite(led3, state2);
      digitalWrite(led4, state2);
      state2 = !state2;
    }
    // led 1
    else if (results.value == 0x38863BE0) // light on off perpose
    {
      digitalWrite(led1, state1);
```

```
      state1 = !state1;
    }
    // led 3
    else if (results.value == 0x38863BD0)
    {
      digitalWrite(led3, state2);
      state2 = !state2;
    }
    // led4
    else if (results.value == 0x38863BF0) // light on off perpose
    {
      digitalWrite(led4, state4);
      state4 = !state4;
    }
    // led 5
    else if (results.value == 0x38863BC8) // light on off perpose
    {
      digitalWrite(led5, state5);
      state5 = !state5;
    }
    /// regulator
    else if (results.value == 0x38863BF2) // up
    {
      c++;
    }
    else if (results.value == 0x38863BFA) // down
    {
      c--;
    }
    else
    {
      //none
    }
  }
  else
  {
    //none
  }
}
```

Fig A2: Code for Manual Remote Control

```
import cv2
import winsound
import urllib.request as ur
import numpy as np
import cv2
from PIL import Image
import time
url = 'http://192.168.0.101:8080/video'
cam = cv2.VideoCapture(url)
while cam.isOpened():
    ret, frame1 = cam.read()
    ret, frame2 = cam.read()
    diff = cv2.absdiff(frame1, frame2)
    gray = cv2.cvtColor(diff, cv2.COLOR_RGB2GRAY)
    blur = cv2.GaussianBlur(gray, (5, 5), 0)
    _, thresh = cv2.threshold(blur, 20, 255, cv2.THRESH_BINARY)
    dilated = cv2.dilate(thresh, None, iterations=3)
    contours, _ = cv2.findContours(dilated, cv2.RETR_TREE,
cv2.CHAIN_APPROX_SIMPLE)
    # cv2.drawContours(frame1, contours, -1, (0, 255, 0), 2)
    for c in contours:
        if cv2.contourArea(c) < 5000:
            continue
        x, y, w, h = cv2.boundingRect(c)
        cv2.rectangle(frame1, (x, y), (x+w, y+h), (0, 255, 0), 2)
        #winsound.PlaySound('alert.wav', winsound.SND_ASYNC)
    if cv2.waitKey(10) == ord('q'):
        break
    cv2.imshow('Granny Cam', frame1)
```

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

```
import datetime
from tkinter import *
import cv2
from PIL import Image, ImageTk
import urllib.request as ur

# Create a GUI app
app = Tk()
app.title("Smart Security Camara and Computer vision ")
# Bind the app with Escape keyboard to
# quit app whenever pressed
app.bind('<Escape>', lambda e: app.quit())
# Create a label and display it on app
label_widget = Label(app)
label_widget.pack()
# Create a function to open camera and
# display it in the label_widget on app
def capture():
  time_stamp = datetime.datetime.now().strftime('%Y-%m-%d-%H-%M-%S')
  file_name = f'Capture-{time_stamp}.png'
  cv2.imwrite(file_name, frame1)
def open_camera():
    global cam,ret,url,width,frame1,frame2,height
    url = str(entry.get())
    area1 = int(entry1.get())
    cam = cv2.VideoCapture(url)
    # Declare the width and height in variables
    width, height = 300, 300
    # Set the width and height
```

```
    cam.set(cv2.CAP_PROP_FRAME_WIDTH, width)
    cam.set(cv2.CAP_PROP_FRAME_HEIGHT, height)

    ret, frame1 = cam.read()

    ret, frame2 = cam.read()

    diff = cv2.absdiff(frame1, frame2)
    gray = cv2.cvtColor(diff, cv2.COLOR_RGB2GRAY)
    blur = cv2.GaussianBlur(gray, (5, 5), 0)
    _, thresh = cv2.threshold(blur, 20, 255, cv2.THRESH_BINARY)
    dilated = cv2.dilate(thresh, None, iterations=3)
    contours, _ = cv2.findContours(dilated, cv2.RETR_TREE,
cv2.CHAIN_APPROX_SIMPLE)
    # cv2.drawContours(frame1, contours, -1, (0, 255, 0), 2)
    for c in contours:
        if cv2.contourArea(c) < area1:
            continue
        x, y, w, h = cv2.boundingRect(c)
        cv2.rectangle(frame1, (x, y), (x + w, y + h), (0, 255, 0), 2)
#imshow like
    captured_image = Image.fromarray( frame1 )
    # Convert captured image to photoimage
    photo_image = ImageTk.PhotoImage(image=captured_image)
    # Displaying photoimage in the label
    label_widget.photo_image = photo_image
    # Configure image in the label
    label_widget.configure(image=photo_image)
    # Repeat the same process after every 10 seconds
    label_widget.after(10, open_camera)
```

```
# Create a button to open the camera in GUI app
entry = Entry(app)
entry1 = Entry(app)
label1 = Label( app, text= "ENTER IP ADRESS",bg= "black", fg= "white")
label2 = Label( app, text= "ENTER CONTOURS AREA UP TO DETECT",bg= "black", fg= "white")
entry.insert(0, "http://192.168.0.100:8080/video")  # Set the default value to "10"
entry1.insert(0, "500")  # Set the default contors detect area value to "500"
label1.pack()
entry.pack()
label2.pack()
entry1.pack()
button1 = Button(app, text="Open Camera", command=open_camera,bg= "green", fg= "white")
button1.pack()
button2 = Button(app, text="Capure", command = capture,bg= "blue", fg= "white")
button3 = Button(app, text="EXIT", command = app.quit,bg= "red", fg= "white")
button2.pack()
button3.pack()
# Create an infinite loop for displaying app on screen
app.mainloop()
```

Fig A4: Code For Computer vision Software

```
import cv2
import mediapipe as mp
import numpy as np
import pyautogui
import serial
import time
arduino = serial.Serial('COM6', 9600)
time.sleep(2)
cap = cv2.VideoCapture(0)
#cap.set(3,1000)
#cap.set(4,1000)
hand_detector = mp.solutions.hands.Hands()
drawing_utils = mp.solutions.drawing_utils
screen_width, screen_height = pyautogui.size()
index_y = 0
value2 = 0

def send(val):

    if val == 1:
        arduino.write(b'1')

    elif val == 2 :
        arduino.write(b'2')

    elif val == 0 :
        arduino.write(b'0')

    elif val == 3:
        arduino.write(b'3')

    elif val == 4:
        arduino.write(b'4')

    elif val == 5:
        arduino.write(b'5')

    else:
        pass




while True:
    _, frame = cap.read()
    frame = cv2.flip(frame, 1)
    frame_height, frame_width, _ = frame.shape
    rgb_frame = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
```

```
output = hand_detector.process(rgb_frame)
hands = output.multi_hand_landmarks
if hands:
    for hand in hands:
        drawing_utils.draw_landmarks(frame, hand)
        landmarks = hand.landmark
        for id, landmark in enumerate(landmarks):
            x = int(landmark.x*frame_width)
            y = int(landmark.y*frame_height)
            th = 2
            if id == 8:
                cv2.circle(img=frame, center=(x,y), radius=20, color=(0, 0, 255),thickness=2)
                index_x = screen_width/frame_width*x
                index_y = screen_height/frame_height*y

            if id == 4:
                cv2.circle(img=frame, center=(x,y), radius=20, color=(0, 0, 255),thickness=2)
                thumb_x = screen_width/frame_width*x
                thumb_y = screen_height/frame_height*y
                #print('outside', abs(index_y - thumb_y))
                v = int(index_y - thumb_y)
                v = abs(v)
                value = np.interp(v,[50,300],[0,10])
                value2 = np.interp(value, [0, 10], [400, 150])
                #v = min(v,100)
                ccc = np.interp(value,[0,10],[0,5])
                print(int(ccc))
                send(int(ccc))


                cv2.putText(frame, f"Speed {int(value*10)}%", (75, 50),
cv2.FONT_HERSHEY_COMPLEX, 0.9, (0, 0, 255), 2)
                cv2.rectangle(frame, (50, 150), (85, 400), (0, 255, 0), 3)
                cv2.rectangle(frame, (50,int(value2)), (85, 400), (0, 0, 255), cv2.FILLED)
                #print(int(value))

    cv2.imshow('Speed regulator', frame)
    cv2.waitKey(1)
```

Fig A5: Code For Gesture Control Voltage Dimmer

APPENDIX D

DATASHEET OF COMPONENTS

D1 Arduino uno r3

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

D2 Node MCU ESP8266

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

D3 DTH11 Sensor

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

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

D5 Relay Module

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

D6 IR Sensor

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

D7 IR Transmitter

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

D8 7432 IC

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

D9 RBD Dimmer

https://github.com/RobotDynOfficial/RBDDimmer