Design of an Energy Efficient Intelligent Classroom using Image processed Occupant Detection Model

by

Reduan Ahmed Tanim (160021110)
Shobnom Mustari Prottasha (160021120)
Papia Tanjia (160021132)

BACHELOR OF SCIENCE IN ELECTRICAL AND ELECTRONIC ENGINEERING

Department of Electrical and Electronic Engineering
Islamic University of Technology (IUT)
Board Bazar, Gazipur-1704, Bangladesh.
February, 2021.

CERTIFICATE OF APPROVAL

The thesis titled, "**Design of an Energy Efficient Intelligent Classroom using Image processed Occupant Detection Model** " accepted as partial fulfillment of the requirement for the Degree of BACHELOR OF SCIENCE INELECTRICAL AND ELECTRONIC ENGINEERING of Islamic University of Technology (IUT).

Approved By:

Dr. Khondokar Habibul Kabir
(Supervisor)
Professor,
Electrical and Electronic Engineering Department,
Islamic University of Technology (IUT), Gazipur.

Declaration of Candidate

It is hereby declared that this thesis report is only submitted to the Electrical Engineering Department. Any part of it has not been submitted elsewhere for the award of any Degree or Diploma.

Tanim

Reduan Ahmed Tanim
Student Id.: 160021110

Shobnom

Shobnom Mustari Prottasha
Student Id.: 160021120

Papia Tanjia
Student Id.: 160021132

Table of Contents

List of Figures

List of Tables

List of Abbreviations

MOG2	Mixture Of Gaussian 2
OPENCV	Open Source Computer Vision Library
GPIO	General Purpose Input/Output
MCU	MicroController Unit
USB	Universal Serial Bus
IP	Internet Protocol
TCP	Transmission Control Protocol
SAD	Sum of Absolute Difference
CT	Current Transformer
AC	Alternating Current
DC	Direct Current
SD card	Secure Digital card
PSU	Power Supply Unit
DSI	Display Serial Interface
CSI	Camera Serial Interface
HDMI	High Definition Multimedia Interface
MIPI	Mobile Industry Processor Interface
ES	Embedded System
KDE	K Desktop Environment
GPU	Graphics Processing Unit
ARM	Advanced RISC Machines
SDRAM	Synchronous Dynamic Random-Access Memory
BLE	Bluetooth Low Energy
CCD	Charge-Coupled Device
HAT	Host Access Transformation
PoE	Power over Ethernet
NOOBS	New Out of Box Software
OS	Operating System
ISO	International Organization for Standardization
IoT	Internet of Things
Tk	Taka
KWH	Kilo Watt Hour
CCTV	Close Circuit Tele-Vision
RFID	Radio-frequency Identification
SDK	Software Development Kit
AI	Artificial Intelligence
RGB	Red Green Blue

Acknowledgment

We have our wholehearted admiration for the support, assistance, and guidance that many people have showered us with throughout our undergraduate effort on countless occasions. We would like to have the honor to express our heartfelt gratitude to our beloved advisor, our guidance Dr. Khondokar Habibul Kabir sir in the first place for his continuous assistance, inspiration, and invaluable suggestions from time to time. Without his support, our relentless work towards our research would not have resulted in success. We would also like to thank the other members of our thesis committee for their contemplative advice. We also wish to express our appreciation to our family members for the support and love throughout our whole life. Above all, our glorification to ALLAH subhanahu wa ta'la, THE ALMIGHTY would always fall short for giving us strength and perseverance in the process of this accomplishment.

Abstract

Electrical energy saving is undeniably the crying need in our modern world since energy consumption is increasing every day but production is not increasing at a proportional rate. Without paying much attention, we use a lot of energy every day which causes wastage of a large amount of energy. Sometimes the electrical appliances are left turned on due to negligence of the user and the appliances are used unnecessarily even though no user is present in the room. This is one of the major reasons behind this huge amount of electrical energy wastage. Our dissertation describes the development of a system that may assist in reducing the extravagance of energy resources. The main purpose of our thesis is to save energy that would otherwise be wasted and also automate the system so that it does not need to rely on human intervention. We develop a smart system that will control the electrical appliances through occupant detection using image processing and act accordingly. To capture the real-time image, the Pi camera is embedded with a Raspberry Pi 4.0 Model B for processing purposes. Python 2.7 was used to write the code for the relevant algorithm. We use different occupant detection models and compare them to get the best result. We have implemented the project considering a classroom in our university. Here a classroom is divided into several cells. Whenever a movement is detected in a particular cell then the appliances are switched on. The system is an IoT-based smart classroom using Image Processing. Our paper demonstrates the detection of occupants and it is used as a parameter for the system to decide whether to switch on/off the appliances.

Chapter 1

Introduction

Electrical energy demand is increasing day by day. Electricity is an essential part of modern life. At present people are fully dependent on electricity. But more electrical energy consumption causes a huge amount of electric bill. Most often people waste energy due to their negligence. Because of the busy schedule existing in present, frequently most of us neglect to turn off lights and fans, which causes energy consumption without any reason [1]. This energy consumption without any reason should be reduced.

1.1 Problem Statement

The lights and fans are sometimes unnecessarily used as well when the students leave the classroom switching on the lights and fans and neglect to turn them off. Automation can help in accomplishing full control over the electric equipment surrounding us at our home or workplace [2].

IoT can help in building an intelligent framework that could comprehend our necessities and requirements in such a manner without even the need to order them. IoT can bring notable benefits to the energy savings [3]. The classroom automation system can be developed by IoT that can potentially scan every person passing in front of it and it will start tracking the movement of that person and see where that person sits or stands and turn on the light and fan of that particular position. That fan and light turn off automatically when there is no one under those appliances. Several kinds of research have been done before on this issue. But the issue is not fully solved and also not fully suitable for a classroom. The proposed model of this book can solve this issue.

1.1.1 Research Gap

An IoT-based system can help to solve this issue. Energy consumption can be reduced using IoT based system. The idea of the Internet of things (IoT) makes a network of

items that can convey, collaborate and coordinate together to arrive at a shared objective [4]. IoT devices can improve our everyday life. This gives us the subsequent information to be broken down for better dynamic, following our organizations and checking our properties while we are far away from them [5].
There are many applications of IoT. One of the most popular applications is classroom automation and this is the main focus of this book. For the automation of the classroom, object detection, and motion detection is required. There are different image processing models for doing object detection.
Object detection is an IoT-based procedure that permits us to recognize and find objects in a picture or video. There are several occupant detection models.
This book addresses an IoT-based intelligent classroom using the MOG2 algorithm with time buffer and also a comparative study of different occupant detection models. Here the main focus is reducing electric bills by reducing energy consumption. This can be done in several ways.

1.1.2 Problem Identification

Electrical energy saving is undeniably the crying need in the modern world since energy consumption is increasing every day but production is not increasing at a proportional rate. Without paying much attention, we use a lot of energy every day which causes wastage of a large amount of energy. Sometimes the electrical appliances are left turned on due to negligence of the user and the appliances are used unnecessarily even though no user is present in the room.
This is one of the major reasons behind this huge amount of electrical energy wastage. IoT can bring notable benefits to energy savings. So a system should be developed that can help in reducing the extravagance of energy resources. The main purpose of such a project is to save energy that would otherwise be wasted and also automate the system so that it does not need to rely on human intervention. Develop a smart intelligent system is very important that will control the electrical appliances through occupant detection using image processing and act accordingly.

1.1.3 Research Motivation

To implement this project an educational institution is focused here. Assuming an institution has two-hundred classrooms each consisting of sixty-four students, where daily six classes each consisting of ninety minutes take place and the institute has 200 working days on average per year, the energy consumption of the institution is a very huge amount. Moreover, the lights and fans are sometimes unnecessarily used as well when the students leave the classroom switching on the lights and fans and neglect to turn them off. After applying our developed model in the system, the energy consumption of the institution can be decreased in a large amount. Detail of the assumed energy saving and saving in the electric bill for a year are all described in detail in Chapter 3. So, we can see this system reduces the energy consumption that reduces the cost per year electric bill significantly. So the implementation of this type of system is very important to reduce energy consumption which will eventually reduce the electric bill. While classroom automation may sound expensive in reality, but the system is very cost-effective, user friendly, and customized.

1.1.4 Scope of the Research

The system can be implemented in many places other than classrooms like libraries, houses, banks, hospitals, laboratories, offices, industries, or any other place where occupants are present and appliances are needed to turn on/off frequently. As the system is using the existing electrical wiring for the appliances, it does not need to do any wiring. It only needs to integrate with the switchboard and connect the existing Raspberry Pi camera with the Raspberry Pi.

1.2 Research Objectives

The main objective of the thesis is to reduce electric bills by reducing energy consumption by developing a system that can potentially scan every person passing in front of it and it will start tracking the movement of that person and see where that person sits or stands and turn on the light and fan of that particular position. That fan and light turn off automatically when there is no one under those appliances. More specifically, the objectives are:

- To compare different occupant detection models.
- To find the most effective occupant detection model.
- To develop an energy-efficient intelligent classroom.
- To build a cost-effective user-friendly, customizable system.

1.3 Research Outcomes

Implementing this system, we will be able to build a smart classroom where all the appliances are controlled automatically depending on the occupant's presence.

- This provides the comparison of different occupant detection models. Several detection models is implanted in this work.

- This work finds the most effective occupant detection model.
- This work develops an energy-efficient intelligent classroom. Thus, a huge amount of energy can be saved yearly which will decrease the load on the national grid eventually.
- The developed system builds a cost-effective user-friendly, customizable system.

1.4 Novelty of the research

Our target of the research is to find the best occupant detection model that can best automate the classroom and eventually can be implemented in automating any crowded place. For this purpose, four types of occupant detection models analyzed in this book. The novelty of our work is using the MOG2 algorithm adding a time buffer with it. As we have studied so far, adding time buffer with MOG2 algorithm is a new idea.

1.5 Overview of Methodology

Four occupant detection models are compared in this book. The models are-
I. Background Subtraction(Pixel Subtraction)
II. Background Subtraction(Absolute Difference)
III. Background Subtraction(MOG2 algorithm)
IV. Background Subtraction(MOG2 algorithm with time buffer)
The main hardware is a Raspberry Pi full edged computer and Pi cam. Raspberry Pi used as computer and Pi cam as camera. For running the Raspberry Pi, an SD card, power adapter, and pendrive is used. Moreover for internet connection, LAN or wifi connection, router connection is needed. For the connection purpose, cables and for mounting everything at a place, a tripod as a stand is used. The environment is Python and the module is OpenCV. The software used in the research is the visual studio code.

1.6 Organization of the Thesis

The thesis paper has been divided into the 8 following chapters-

Chapter 1 presents a short introduction about the use of IoT in order to save the electrical energy. It represents the research gap, problem identification, research motivation, scope of the research, research objectives, research outcomes and so on.

Chapter 2 demonstrates a detailed description of the related work.

Chapter 3 provides detail information about the proposed model.

Chapter 4 describes the comparison of different image processed occupant detection models.

Chapter 5 represents the Mixture of Gaussians 2(MOG2) algorithm

Chapter 6 Contains the hardware and software setup.

Chapter 7 is devoted to represent the calculated results, graphs and explanation of the results obtained.

Chapter 8 contains the conclusions of the whole study.

At the end of each chapter a complete list of references corresponding to that chapter has been given.

Chapter 2

Related Works

Studies during recent years show that “the automation system” may be used in different ways in different fields that is an inspiration to work on.

Darshan Ganiger et al. [6] suggest a simple system that is pre-installed in an institution using the existing setup. But this system does not work for an individual appliance in the classroom. Moreover, the proposed algorithm here does not work properly when the sitting arrangement of the room is changed.

A system is proposed to control all the appliances using a web-based interface or local network by Chinmay Bepery et al. [7]. All the devices of this proposed system are associated with raspberry pi. The proposed framework additionally can control the appliances without the internet through the local network. But this system is not automated. It’s a manually controlled system through the local network.

A low-cost and secure classroom automation system that is both voice-based and sensor-based is proposed by Vaishnavi Kulkarni et al [8]. This automation system also utilizes raspberry pi and makes a network that is programmed by using python. But for being controlled by voice, the system is more like a manually controlled system that needs voice command as an input being voice-based and the sensor-based system is not so convenient as well.

Mrs. Tanuja Sali et al. [9] proposed a classroom automation system which is a union of hardware components like relay modules, voltage supplies, and softer components like mobile applications. This presents a model that has been proposed to meet the fundamental necessities of homeroom computerization. A significant advantage of this system is that it provides the system as a total model to develop more such models which will be cost-effective and can be used in the classroom.

Anisha Gupta et al. [10] suggests an automation system. The proposed result of the system points as various advantages, saving money on electricity bills of the university or some other association it is sent in, reducing human intervention and labor which is frequently needed to physically flip the lights and electrical gadget on/off, and last yet above all, conserve the valuable natural resources by lessening electrical energy utilization. But this system is more like a remote control system. Moreover, this system is very expensive. Also, the wiring of the relay board and other components of this system is done in a wired way which is not very convenient.

Chinju Paul et al. [2]represents an approach which interaction is carried out by understanding the user's order via his voice. Hidden commands are primarily used to communicate with the middleware. The raspberry pi serves as the system's backbone

Eueung Mulyana et al. has proposed a chatbot-based virtual assistant [11] . It is different from other systems as it is based on the chat ops paradigm instead of AI-based algorithms. But this system is more like a robot that can help in taking attendance, organizing class lectures, and answering questions related to courses, so it does not help in automating the appliances of the classroom and contributing to energy-saving.

L. A. Marcomini et al has proposed a comparative study of the performance of three background-modeling algorithms in segmenting and detecting [12]. This shows that the MOG2 algorithm has a better performance than others.

Rathy G. A et al. [13] proposed a model on classroom automation system by installing smart internet-connected devices in the classroom. It detects the movement and based on the detection of the movement this system turns on/off the lights.

Dhiren Tejani et al [14] has analyzed power consumption of standard homes considering regularly utilized appliances and the saving of power after the implementation of home automation . It shows that after the implementation of the home automation system the energy consumption is reduced in a noticeable amount.

Satyendra K. Vishwakarma et al. [3] proposed a smart, energy-efficient home automation device that can be accessed and controlled from anywhere in the world.The static IP address is used in this model. This project makes use of the user's voice recognition command through google assistant.

Chun-Yu, Chen et al. [15] gives a proposal to integrate a different group of smart appliances. More details about the development and execution of the vital advancements for the smart home network, smart appliances are described in this paper. The outcome is that it shows that the application of smart sensors and smart home networks will reduce energy consumption.

Chapter 3

Overview of the Proposed Model

Here we are taking classroom for our system implementation as we are making classroom efficient in terms of energy consumption. For the implementation purpose of the proposed model, we are having some illustrations taking the classroom of our university as a sample.

3.1 Model Classroom Representation

We are taking the classroom of our institution, Islamic University of Technology as the model classroom. Due to lockdown, we failed to collect picture from our classroom. That is why we made a power point presentation having the same number of sitting arrangement, appliances and student number.

Figure 3.1: The Model Classroom

Here, there is in total sixteen tables, each table is for four students. So, the sitting arrangement is for sixty-four students.

3.2 Targeted Amount of Energy Saving in A Year

Here we are representing a calculation of the targeted amount of energy saving in a year and also the saving in electric bill due to that.

As seen from Figure 3.1, we are assuming-

1.An institution has 200 classrooms.
2.Each consisting of 64 students.
3.One light of 40 Watt for every 4 students and 2 lights for the teacher.
4.One fan of 100 Watt for every 8 students.
5.Daily 6 periods take place.
6.Each period consists of 1.5 hour.
7.The institute has 200 working days on average per year.

Table 3.1: Description of the classroom

Total number of lights :

$$64 \div 4 + 2 = 18 \tag{3.1}$$

Total number of fans :

$$64 \div 8 = 8 \tag{3.2}$$

Total energy consumption of one classroom for one day in Kwh:

$$((18 \times 40 + 8 \times 100) \times 9) \div 1000 = 13.68 \tag{3.3}$$

Total energy consumption of the institution for one year in Kwh:

$$13.68 \times 200 \times 200 = 547200 \tag{3.4}$$

So, the energy consumption of the institution is calculated as 547.2 Mwh for a year. According to Bangladesh Energy Regulatory Commission (BERC) per unit energy consumption bill is 11.46 taka for consumers over 600 units. So we can see that, it costs 6.27 million taka per year as the electric bill.

After applying our developed model in the system, the working hour of one classroom for one day becomes 7.5 hours and the average power consumption of one classroom for one period becomes 1108.148148 Watt. So the energy consumption of one classroom for one day in Kwh from the collected data according to Table 6.4 is :

$$1108.15 \times 7.5 = 8.31 \tag{3.5}$$

So, the total energy consumption of the institution for one year in Kwh from the collected data according to Table 6.4 :

$$8.31 \times 200 \times 200 = 332400 \tag{3.6}$$

So, the energy consumption of the institution is calculated as 332.44 Mwh and we can see that it costs 3.81 million taka per year as the electric bill now. The energy-saving for the institution for a year is thus 214.7 Mwh and so the saving in the electric bill for a year is 2.46 million taka.

3.3 Technique

At first, the system does the image processing to detect if there is any occupant present. Then it matches the live situation, which is the present frame with the previous frame. The image processing method is utilized here to distinguish the movement and dependent on the outcome whether to turn on/off the appliances of the classroom [?]. The system uses ESP8266 and Raspberry Pi. These devices are connected through wifi. Because of its low cost modularity and open architecture, the Raspberry Pi is a series of compact single-board computers that are used in a variety of applications [16]. It is a powerful computer and supports the most recent web innovations and data sets, there is plenty of possible extensions for extending the project and coordinate more highlights easily and successfully [17]. The ESP8266 is a low-cost wifi microchip that includes a complete TCP/IP stack as well as a microcontroller. The relay module is also used in this system. It is a separate hardware device used for remote device switching. With it, a device is controlled remotely over a network or the internet [18]. When the Raspberry Pi boots up the content which incorporate the way of the proposed algorithm begins executing and starts recognizing the movement of occupants by comparing the previous frame and the present frame caught by the camera of the system. If there is any movement detected in any particular place of the classroom, then the system sends Raspberry Pi a message that there is somebody present in that particular place. Therefore, Raspberry Pi detects that occupant standing or sitting, gives a signal to ESP8266 and ESP8266 sends a signal to the relay module to turn on the appliances and the appliances of that particular place turn on [19].

Here the environment is Python and the module is OpenCV. The code is done in Python language using Visual Studio Code. Thonny Python is used to run the Python script for Raspberry Pi.

3.4 Input

The inputs can be primarily categorized into four types. These are written synchronously:
1. Reference image of the classroom is considered as the reference frame.
2. Image of the coordinate system of the room. For this, the system selects four coordinate points at four corners of the room to get a perfect bird's eye view.
3. Detects the position of the appliances.
4. A live video that will be always running and considered as the captured frame.

3.5 Dependency

The system is dependent on occupants coming to the room. Detecting occupants and their position, the system will start working.

3.6 Parameter

The constant parameters are- the reference frame, coordinate system, position of the device and the variable parameters are- live video of the classroom and the number of occupants standing or sitting under the appliances.

3.7 Checking Hardware and Software

For the checking purpose, we will input our code into our hardware and see if the system runs soundly.

3.8 Syncing Hardware and Software for collecting expected Data

It is a calibration process. If we give input and get the expected output, it shows our hardware and software are syncing. We will check continuously and repeatedly if the output is as expected or not. We need to check how many occupants are not detected correctly, how many times the appliances are not turned on or off as expected. We collect our data according to every situation. Then we change the calibration, create a different scenario, get different outputs, and collect data again.

In the next chapter, we will show the detailed methodology of our proposed model, pros and cons of all the four occupant detection models and compare all the four models to find out the best one.

Chapter 4

Methodology

In this chapter, four occupant detection models are compared to find the most optimal solution for reliable detection in indoor environments like the classroom. The models were all tested with a common test rig for the fairground. The models are:
1. Background subtraction (Pixel subtraction)
2. Background subtraction (Absolute difference)
3. Background subtraction (MOG2 algorithm)
4. Background subtraction (MOG2 algorithm with time buffer)

4.1 Background subtraction (Pixel subtraction)

Image subtraction, also known as pixel subtraction, is the mechanism of subtracting one pixel's or an entire image's digital numeric value from another image. This is done for one of two reasons: leveling uneven areas of an image, such as half of a shadowed image, or recognizing discrepancies between two images [20].
This shift detection technique can determine whether or not anything in the picture has changed. This is widely used in fields astrophotography to aid in the computerized hunt for asteroids or Kuiper belt objects where the target is moving and will be in one position in one image. Another an hour later, and using this method will make the fixed stars in the background vanish, leaving only the target [20].

4.1.1 Basics of Pixel Subtraction

The pixel subtraction operator takes two images as input and creates a third image with pixel values equal to the first image minus the corresponding pixel value of the second image. Using a single image as the input and subtracting a constant value from all of the pixels is often common. Any implementations of the operator will actually emit the absolute difference between pixel values instead of the straightforward signed display.

The subtraction of two images in a single pass is simple.
The pixel values for the display are as follows [21]:

$$Q(i,j) = P_1(i,j) - P_2(i,j) \tag{4.1}$$

If the pixel values in the input images are vectors rather than scalar values (as they are in color images), the individual components (red, blue, and green, for example) are easily subtracted to get the output value [21].
The operator's implementations change when the display pixel values are negative. Negative pixel values are acceptable when dealing with image formats that allow for them (and how they are displayed will be determined by the display colormap). Such pixels are normally set to 0 if the image format does not allow negative numbers (i.e. black typically). Negative values would be 'wrapped' in the figures, making them 226. (assuming 8-bit pixel values). The output pixel values cannot be beyond the spectrum that the input pixel type can reflect while the operator measures absolute differences for all input images of the same pixel value type, so this issue is avoided. The use of absolute inequality has a legitimate reason. The original offset shape can be selected with the diagoffset swap for contrast.

4.1.2 Working procedure of Pixel Subtraction

In this model, the system takes in a continuous image array from the camera and subtracts the captured frame from the reference frame. Thus, a new matrix is obtained which captures the objects that were not present in the reference frame.
The reference frame is taken first. The captured frame is the real-time image, which is subtracted from the reference frame to get the subtracted frame as result. Here every pixel is defined by an RGB value. Here the RGB value of every pixel is written in Figure 4.1 where every RGB value is expressed by two hexadecimal digits.

As seen from Figure 4.1, the 1st pixel of the captured frame has the RGB value of 800000ff, so here the red value is 80, whereas both the green and blue values are 00. That is why the pixel is red dominant. When this pixel is subtracted from the 1st pixel of the reference frame having the RGB value of a91313ff, where red value is a9, green value is 13 and blue value is 13, the subtracted pixel with RGB value is 291313ff, where red value is 29, green value is 13 and blue value is 13. So this is how every pixel of the captured frame is subtracted from the corresponding pixel of the reference frame to get the subtracted frame for canny and edge detection.

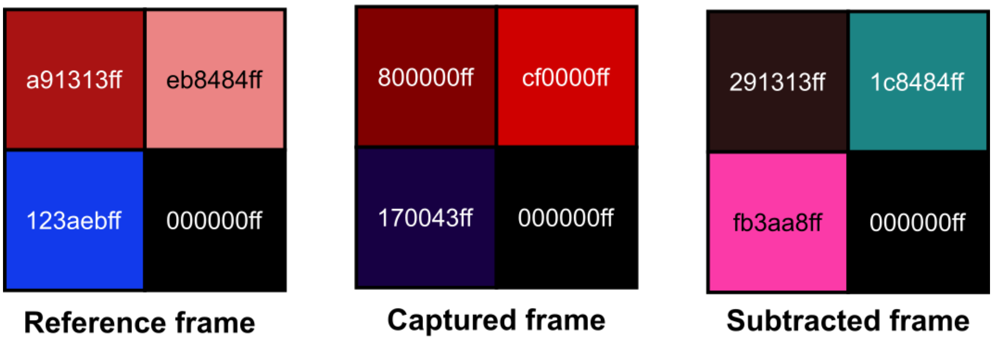

Figure 4.1: Background subtraction using Pixel subtraction

4.1.3 Object Detection algorithm for Pixel Subtraction

The Object Detection algorithm can be divided into six different steps:
Step 1: Reference Frame
Step 2: Captured Frame
Step 3: Edge Detection
Step 4: Contour Detection
Step 5: Marking Detected Objects
Step 6: Output using Region Detection
The following parts explain every step –

Step 1: Reference Frame

First, the system takes the reference image of the empty classroom represented by the four tiles in the middle in Figure 4.2 and this image is considered as the reference frame.

Figure 4.2: Reference Image

Step 2: Captured Frame

Secondly, the system takes the captured image which is the real-time image where there are many students in the classroom represented by the four tiles in the middle in Figure 4.3. It is a continuous image.

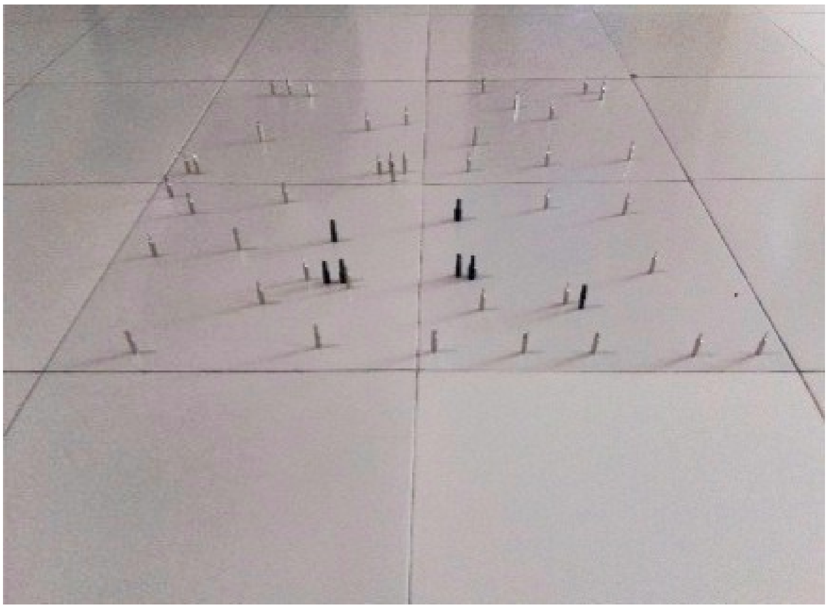

Figure 4.3: Captured Image

Step 3: Edge Detection

Thirdly, the system does edge detection. The bird's eye view of the captured image in Figure 4.3 is shown in this step in Figure 4.4. Gaussian blur and Canny are applied to find the edges. Here the "Image Subtraction" method is used. But the problem in this step is- as the computer calculates pixel by pixel and white object has high pixel value and black object has low pixel value, so when it subtracts white object from a white object of the same position the output is black. In the same way, when it subtracts white objects from black objects again it comes almost black. That's why Figure 4.4 is mostly black detecting very few students. So edge detection was not properly done.

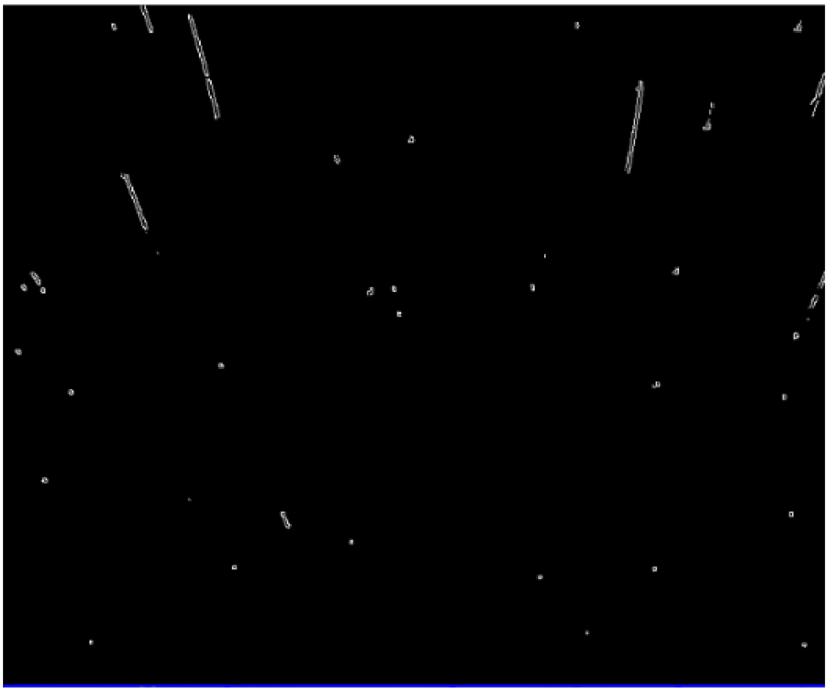

Figure 4.4: Edge Detection

Step 4: Contour Detection

Fourthly, the system does contour detection. The objects which ones were detected are bounded with bounding boxes. These boxes are also serialized according to the contrast of the detected objects as shown in Figure 4.5.

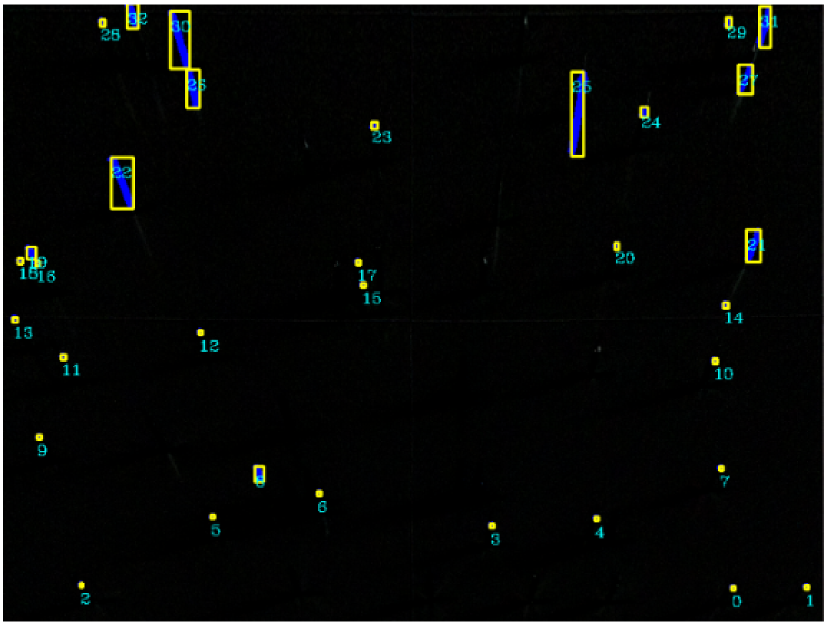

Figure 4.5: Contour Detection

Step 5: Marking Detected Objects

Fifthly, the system marks the detected objects after overlapping Figure 4.5 to Figure 4.3. After doing edge detection and contour detection, the yellow boxes are showing the detected objects as shown in Figure 4.6. The objects which are not in the yellow box are not detected.

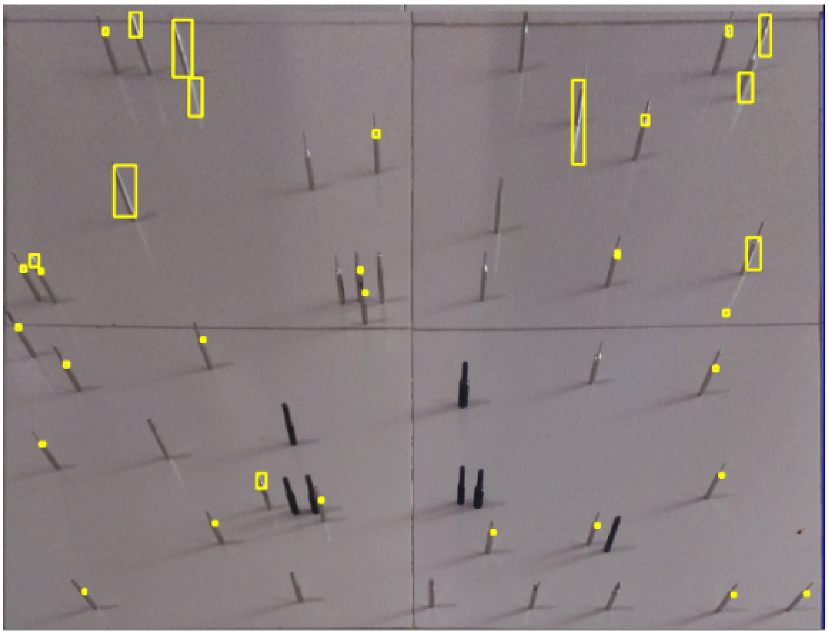

Figure 4.6: Marking Detected Objects

Step 6: Output using Region Detection

In the final step, the system results in the output using region detection. Here the classroom is divided into 16 sub-section each having one fan and one light. When there is at least one object in any sub-section, that sub-section is marked by a green indication and black tick mark, and the light-fan is turned on. If not, then it is marked by a red indication and black cross mark and is turned off. The output is shown here in Figure 4.7.

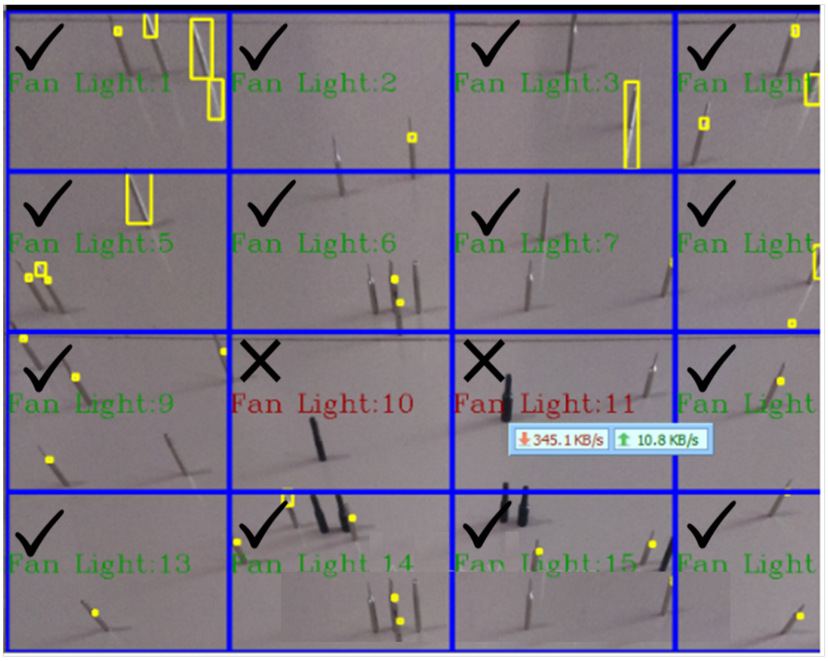

Figure 4.7: Output using Region Detection

For convenience, the whole working process of the 1st model is shown in Figure 4.8 by a flow chart:

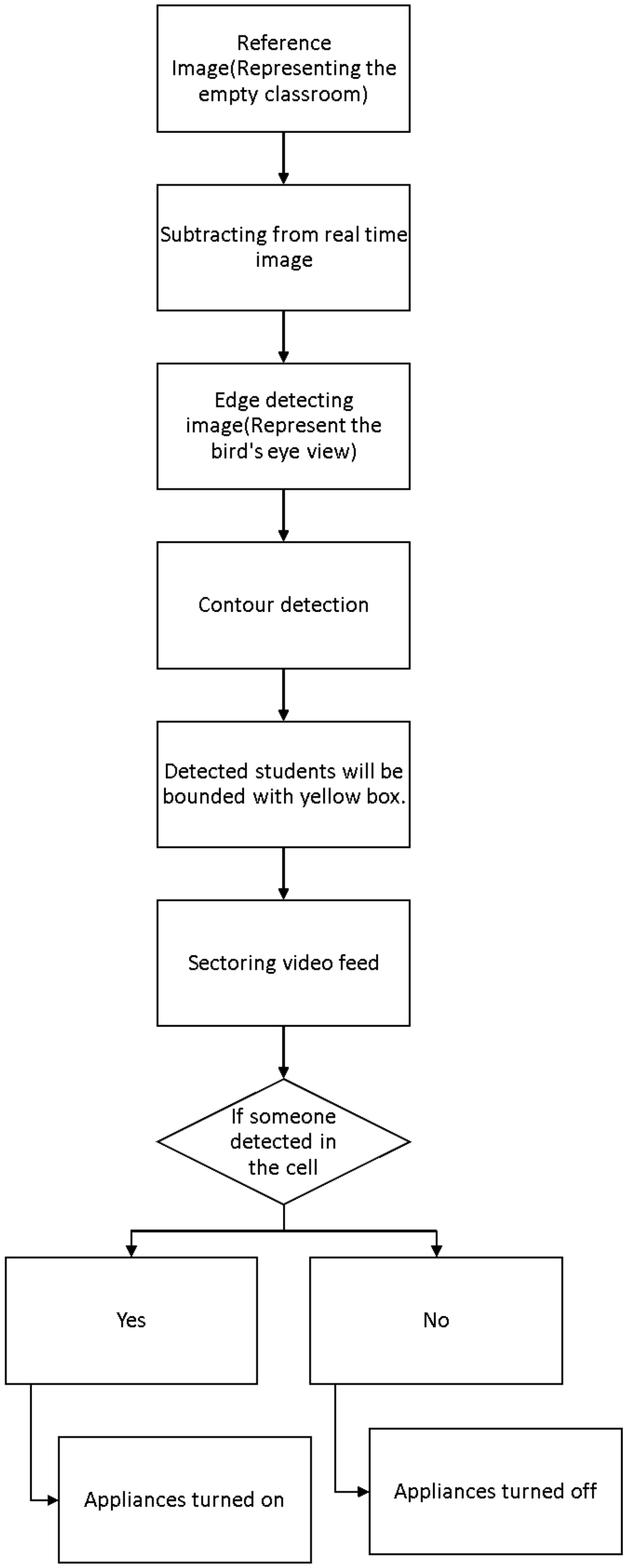

Figure 4.8: Flowchart for the 1st model

4.1.4 Issues with Pixel subtraction process

The resultant pixel value in the subtracted frame becomes non-distinct after subtraction. Like in Fig., the 1st subtraction between a91313ff and 800000ff results in RGB value 291313ff which almost seems the same as the RGB value of the resultant from the subtraction of 000000ff from 000000ff which results in RGB value 000000ff. So even if these are two different subtractions resulting in different resultants, they almost look similar, so that is why here canny and edge detection is tough. Here occupant detection depends on the existing pixel value of the reference and captured frame. Therefore, if the subtracted frame is not distinct depending on the RGB value of the pixel, occupant detection cannot be done accurately. Therefore, this model is not working fine. There are many problems in detection, also in function. That is why the 2nd model which is Background Subtraction (Absolute difference) is proposed for a better result.

4.2 Background subtraction (Absolute Difference)

In optical image processing, the absolute difference is a measure of image block similarities. To measure it, the absolute difference between each pixel in the original block and the corresponding pixel in the reference block is used [22].

4.2.1 Basics of Absolute Difference

To construct a simple metric of block similarity, the L1 norm of the difference image or Manhattan distance between two image blocks is generated by adding these differences. Item detection, stereo image variance mapping, and video encoding motion prediction can all be done with the number of absolute differences [22].
The first frame is shot using a static image, followed by a series of frames taken at regular intervals. Secondly, the absolute difference between the consecutive frames is determined and the machine saves the difference picture. Thirdly, there's the difference picture is converted to a gray-scale image, which is then converted to a binary image. Lastly, to minimize noise, morphological filtering is used [21].
The operator computes absolute differences between the two input images, the following results will be obtained [23]:

$$Q = |P_1(i,j) - P_2(i,j)| \tag{4.2}$$

Context subtraction was essentially used in this process. When the amount of absolute variance (SAD) is smaller than the motion threshold, the background picture is measured every 0.8 seconds. Until background subtraction, the input picture is luminance normalized. The findings were amplified with dilation and erosion after being transformed into binary images by auto-threshold. For each motion object, blobs were formed. The findings by using a background image determined by periodic background calculation in background subtraction for a real-world scene show their robustness and efficacy.

4.2.2 Working procedure of Absolute Difference

In this model, another approach is used to solve the problems faced in the 1st model. Here, in the case of subtracting the captured frame, the computer considers the absolute difference between the reference frame and the captured frame.
Similar to the 1st model, the reference frame is taken first. The captured frame is subtracted from the reference frame to get the absolute difference frame as result. In this case, the RGB values of every two corresponding pixels of the reference frame and captured frame are matched and the absolute difference is found. If two corresponding RGB values completely match, then the resultant RGB value becomes 000000ff that resembles black and if two RGB values do not completely match, then the resultant RGB value becomes ffffffff that resembles white.

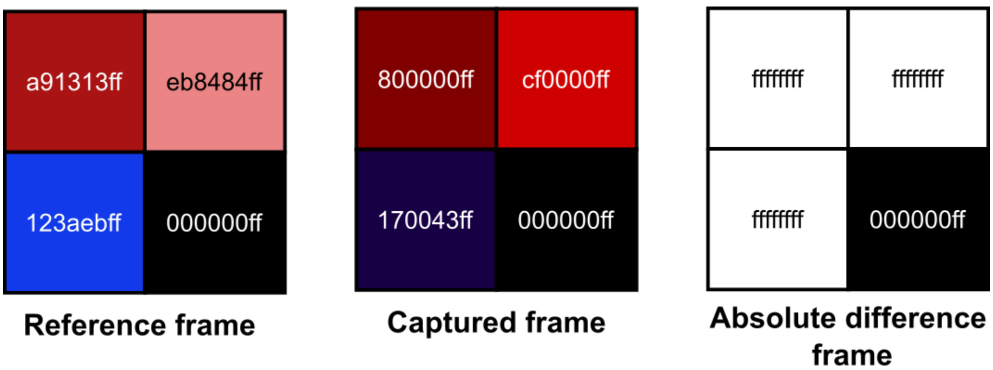

Figure 4.9: Background subtraction using Absolute difference

As seen from Figure 4.9, the 1st pixel of the captured image has the RGB value of 800000ff, when this pixel is subtracted from the 1st pixel of the reference image having the RGB value of a91313ff, these two RGB values do not match. Therefore, the resultant pixel has the RGB value of ffffffff where the highest value of red, green, and blue is obtained and the resultant pixel resembles white. In another case, the last pixel of the captured image has the RGB value of 000000ff, when this pixel is subtracted from the 1st pixel of the reference image having the RGB value of 000000ff, these two RGB values completely match. Therefore, the resultant pixel has the RGB value of 000000ff where the lowest value of red, green, and blue is obtained and the resultant pixel resembles black. Therefore, this is how every pixel of the captured frame is subtracted from the corresponding pixel of the reference image to get the absolute difference frame for canny and edge detection.

The advantage of this method is, whenever two corresponding RGB values of two pixels do not match, then a distinct RGB value of the resultant pixel is set, so the edge detection becomes very easy, and eventually, object detection is done very precisely. It also works for seat exchanging purposes or if anyone leaves the room, in that case, that appliance will be turned off and another will be turned on.

4.2.3 Object Detection algorithm for Absolute Difference

The Object Detection algorithm can be divided into six different steps:

3Step 1: Reference Frame

Step 2: Captured Frame

Step 3: Canny Edge Detection

Step 4: Edge Detection

Step 5: Marking Detected Objects

Step 6: Output using Region Detection

The following parts explain every step –

Step 1: Reference Frame

First, the system takes the reference image of the empty classroom represented by the four tiles in the middle in Figure 4.10 and this image is considered as the reference frame.

Figure 4.10: Reference Image

Step 2: Captured Frame

Secondly, the system takes the captured image which is the real-time image where there are many students in the classroom represented by the four tiles in the middle in Figure 4.11. It is a continuous image.

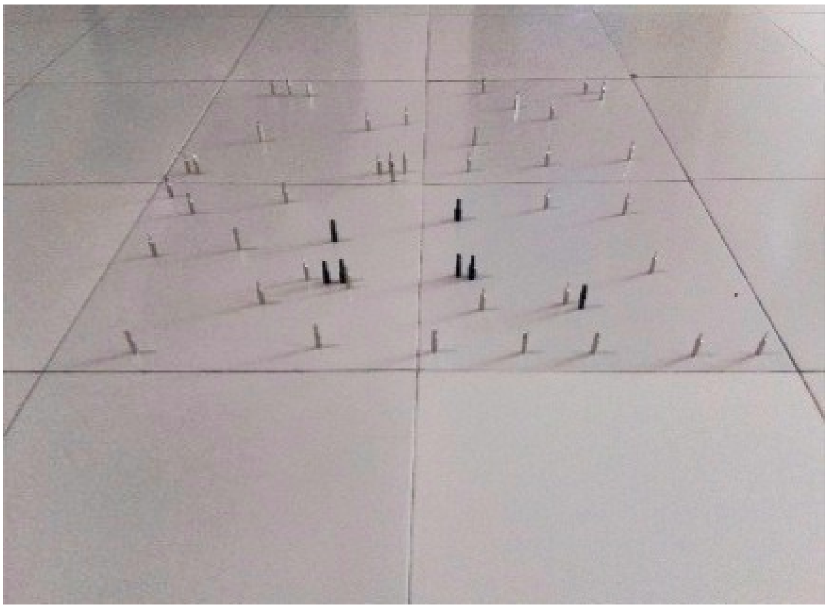

Figure 4.11: Captured Image

Step 3: Canny Edge Detection

Thirdly, the device detects the object's canny edges in the picture. Canny edge detection is a technique for detecting the edges of objects in a picture. It accepts a grayscale image as input and uses a multi-stage algorithm [24] as shown in Figure 4.12. Here the computer detects the canny edge of every object detecting a wide range of edges from the bird's eye view of Figure 4.11.

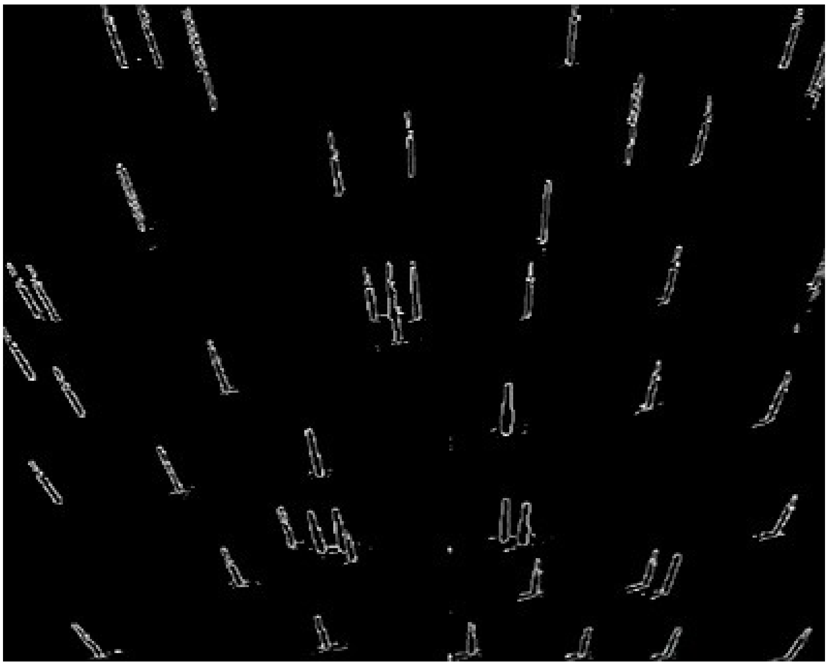

Figure 4.12: Canny Edge detection

Since all edge detection effects are quickly influenced by image noise, it is important to filter out the noise to avoid false detection. A Gaussian filter kernel is convolved with the image to smooth it out. This phase smoothes the picture slightly to reduce the impact of visible noise on the edge detector. The equation for a Gaussian filter kernel of size$(2k+1) \times (2k+1)$is given [24] by:

$$H_{ij} = \frac{1}{(2\pi\sigma^2)} e^{(-\frac{(i(k+1))^2+(j(k+1))^2}{(2\sigma^2)})} ; 1 \leq i, j \leq (2k+1) \tag{4.3}$$

The smoothed image is then filtered in both horizontal and vertical directions with a Sobel kernel to acquire the first derivative in both horizontal (Gx) and vertical (Gy). The following is the canny edge gradient and path for each pixel derived from these two images cc:

$$E_{edgegradient}(G) = \sqrt{(G_x^2 + G_y^2)} \tag{4.4}$$

$$Angle(\Theta) = tan^{-1}\frac{G_x}{G_y} \tag{4.5}$$

Step 4: Edge Detection

Fourthly, the device is capable of sensing edges. Edge detection refers to a collection of mathematical techniques for detecting points in a digital image where the image brightness abruptly varies or, more formally, where there are discontinuities [24]. The sharp differences in image brightness are normally grouped into a series of curved line segments known as edges. For modeling the results of edge blur in functional applications, some researchers have used a Gaussian smoothed step edge (an error function) as the simplest extension of the ideal step edge model. As a result, a one-dimensional image f with exactly one edge at $x = 0$ can be modeled [24] as:

$$f(x) = \frac{I_r - I_l}{2}(erf(\frac{x}{\sqrt{2}\sigma}) + 1) + I_l \tag{4.6}$$

As shown in Figure 4.13, the edges of all the objects are detected correctly.

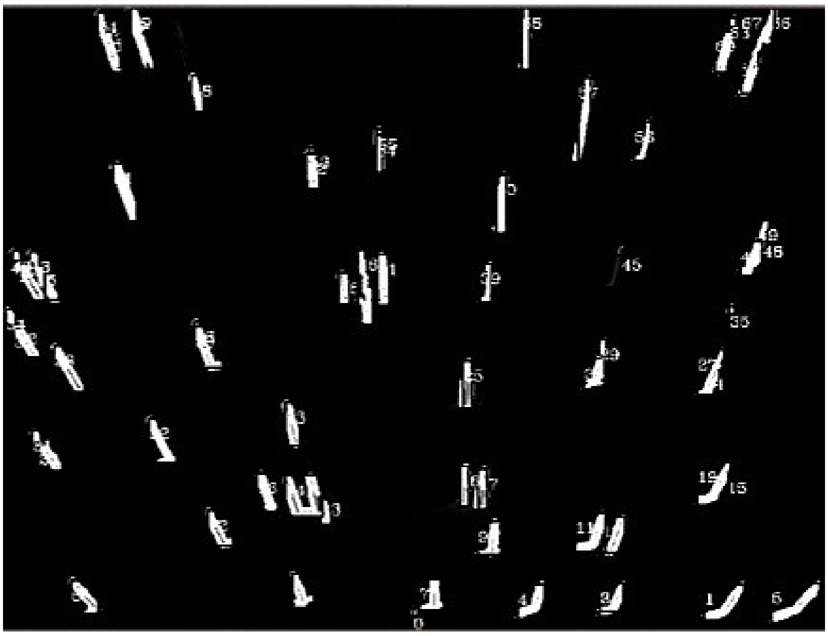

Figure 4.13: Edge detection

Step 5: Marking Detected Objects

Fifthly, the system marks the detected objects after overlapping Figure 4.13 to Figure 4.11. The yellow rectangular boxes as shown in Figure 4.14 are showing the detected objects. The objects, which are not in the yellow box, are not detected. The only object not in the yellow box was detected by edge detection, but in the filtering process, as it was detected as bigger than the bounding box, so it was skipped.

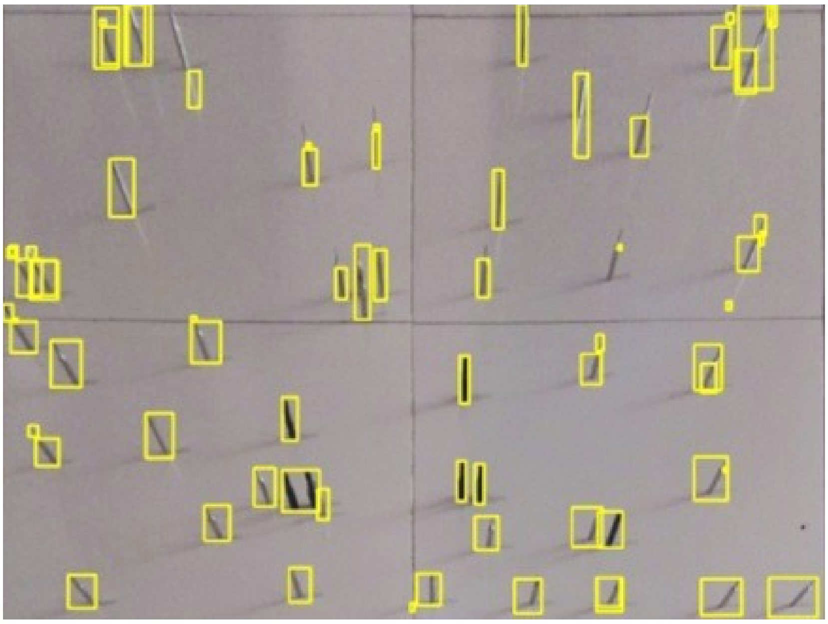

Figure 4.14: Occupant detection

Step 6: Output using Region Detection

In the final step, the system results in the output using region detection. Here the classroom is divided into 16 sub-sections each having one fan and one light. When there is at least one object in any sub-section, that sub-section is marked by a green indication and black tick mark, and the light-fan is turned on. If not, then it is marked by a red indication and black cross mark and is turned off. The output is shown here in Figure 4.15

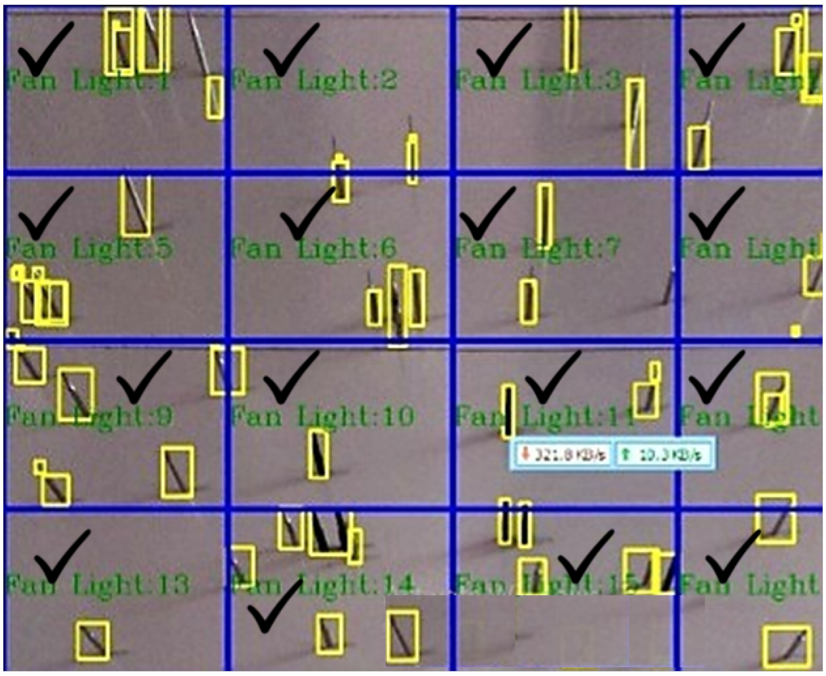

Figure 4.15: System Output

For convenience, the whole working process of the 2nd model is shown in Figure 4.16 by a flow chart:

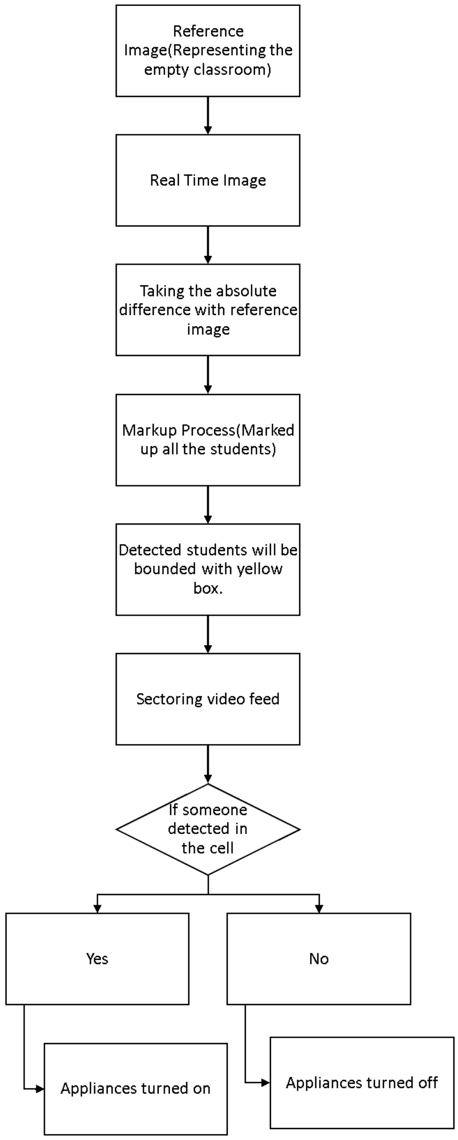

Figure 4.16: Flowchart for the 2nd model

4.2.4 Issues with Absolute Difference process

The issue with this method is it detects every object that does not match with the reference frame absolutely and thus detects non-living objects as occupants. Whenever any occupant leaves one particular place, the chair-table or any other object of that place moves, this movement causes a difference from the reference frame, which is detected by the system. As a result, the appliances of that place turn on. Nevertheless, this should be turned off after the occupant leaves.

As shown in Figure 4.17, initially on the left table two books are there, and on the right table, two books are there as well. In Figure 4.17, on the left table, a new book is placed and on the right table, the left book is moved to its right. Even if the newly placed and moved books in Figure 4.17 are non-living objects, so these should not be detected as occupants. But in Figure 4.17 it is seen that the newly placed and moved books are also detected as occupants. Therefore, the system is active and appliances turn on even if there is no occupant present and electrical energy will be wasted.

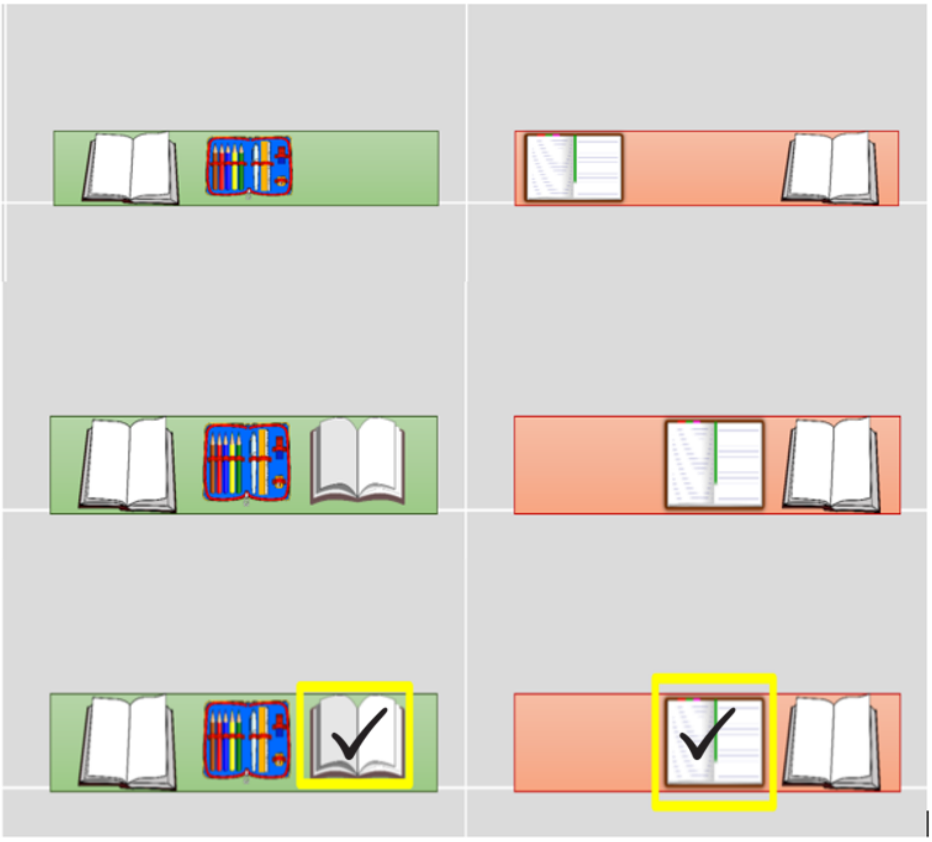

Figure 4.17: Issues with Absolute Difference process

That is why the 3rd model which is Background subtraction using the MOG2 algorithm is proposed for a better result. In the next section, the overview of Background subtraction using the Mixture of Gaussians 2 (MOG2) algorithm will be discussed in detail.

4.3 Mixture of Gaussians 2 (MOG2) Algorithm

In daily life, background subtraction is used for target segmentation, protection improvement, pedestrian monitoring, counting the number of pedestrians, number of vehicles in traffic, and so on. It recognizes and learns the foreground mask. It is capable of removing background portion of an image, as the name implies. Its display is binary segmented image about the image's non stationary elements. This idea of locating non stationary portions has a flaw in it since the shadow of a moving object will move and be classified in the foreground [25].

4.3.1 Basics of MOG2 Algorithm:

It's also a Background/Foreground Segmentation Algorithm based on Gaussian Mixtures. It is more adaptable to changing scenes due to variations in lighting and other factors [26]. It employs a procedure that involves combining K Gaussian distributions (K = 3 to 5) to model each background pixel. The proportions of time that certain colors remain in the scene are represented by the weights of the mixture [27]. This algorithm, which is a key function, selects the necessary number of gaussian distributions for each pixel. Because of changes in lighting and another aspect, it makes for better adaptability to shifting scenes [28].
The MOG2 Subtractor comes preinstalled in Opencv and is much easier to use than the Manual-mode. The Mog2 Subtractor has the advantage of working with a framed past (by example, it deals with the last 120 frames, but this can be changed).

Benefits of dealing with a framed history
Try taking the first frame of a high way during the day; after a few hours, when the sun sets, calculating the difference from the first frame taken a few hours ago would fail and the images taken during the day and night will be completely different. When dealing with the frame past, this isn't a problem [29].

4.3.2 Construction of MOG2 Algorithm:

The constructor **"cv.BackgroundSubtractorMOG2"**(history = 500, varThreshold = 16, detectShadows = true) is used while coding [28].

Parameters:

1. history- The history's duration.
2. varThreshold- A threshold on the squared distance between the pixel and the sample is used to decide whether a pixel is close to a reference. The history update is unaffected by this parameter.
3. detectShadows- If this is so, the algorithm can detect and label shadows. It slows things down a little, so if you don't need it, set the parameter to false.

Returns:
Instance of cv.BackgroundSubtractorMOG2

The steps are shown sequentially for better understanding [29]:

1. The databases and video are loaded.

Following that, the subtractor is loaded. Inside the parenthesis, the sense of the subtractor can be changed. The number of the most recent frame taken into account is referred to as history (by default 120). The threshold value is used to calculate the difference in order to recover the history. More differences would be visible with a lower threshold, but the image will be noisier. If enabled, Detectshadows is an algorithm function that can remove shadows. There are no right or wrong values; you must play with different configurations to find the one that best meets your requirements.
2. The frames from the video are then extracted using the while loop.
while True:
frame = cap.read()
3. The subtractor is used to search the context until having the frames.
mask = subtractor.apply(frame)
4. Finally, everything is shown on the projector.

4.3.3 Real-life Example:

A real-life example for better understanding is shown here [28]:

Frame from the original:

The 200th frame of a video is depicted in Figure 4.18 below:

Figure 4.18: Input for MOG2 Algorithm

Background Subtractor MOG2 result:

The shadow area is represented by the gray color in Figure 4.19 below:

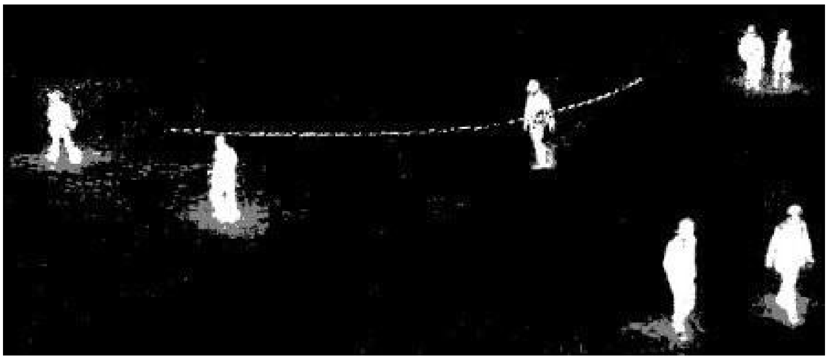

Figure 4.19: Output for MOG2 Algorithm

4.3.4 MOG2 in OpenCV:

More components are produced when Tg is lower since the threshold for comparing a new measure to a previous mode is higher. As a result, a high Tg could lead to a limited number of components with unnecessary material, causing us to reject the current foreground. Old modes are not forgotten as new foreground is discovered so the cumulative number of mixtures has a hard limit. Certain modes have been extended to accommodate the new information.

Finally, since this is the parameter that decreases ambiguity and affects adaptation speed, setting fCT = 0 results in the standard Stauffer Grimson algorithm.

It's difficult to get decent pixel modeling at a lower computing cost than for a Gaussian mixture. Some may argue that using KDE or sample consensus (e.g. ViBe) would have greater precision, but this is debatable (it will depend on your videos and lots of parameters). Few latest history subtraction methods have source code for performance assessment, but don't consider them to be quicker than MOG2 (that one is called 'KNN' or 'GMM'). You should use a modern pixel modeling approach and apply it in parallel or on a GPU if you want a higher rate [27].

Parameter	Explanation
History	The number of previous frames that have an effect on the context model.
Cthr	The noise variation in the picture is believed to be.
n-mixtures	The maximum number of Gaussian mixtures that can be used is four.
Tb	If the part should be used in the default context model, this is the threshold. cf = 0.1 ⇔ Tb = 0.9.
Tg	The minimum level of similarity to the previous part. $3 \times \sigma \rightarrow Tg = 3 \times 3 = 9$
fTau	The shadow threshold dictates how dark the shadow will be. Tau= 0.5 means that if a pixel is a twice darker shadow.
fCT	The number of samples justifies the component's complexity reduction parameter. CT=0.05 is a default.
nShadowDetection	127
fV arInit	Variance at the start for newly created components.
fV arMin	The parameter is used to manage the variance even further.
fV arMax	The parameter is used to manage the variance even further.

Table 4.1: Average Prediction of Recovered Cases for all 6 algorithms

There can be two approaches for MOG2 algorighm. The approaches are:

- MOG2 algorithm
- MOG2 algorithm with time buffer

4.4 1st Approach: MOG2 Algorithm

This model is a Gaussian Mixture-based Background Segmentation Algorithm. The MOG2 algorithm defines the appropriate number of Gaussian distributions for every pixel. MOG2 gains a better representation of the complexity of colors using a variable amount of Gaussians distributions, mapped pixel by pixel in each frame [12].
In this model, the real-time image passes through a filter that detects occupants based

on the difference of the present frame with that of the previous frame thus eliminating the issue of the reference frame. So here, no fixed reference frame is needed.

4.4.1 Occupant Detection process for MOG2 Algorithm

This model is like a real-time application. It only detects for the time when the occupant is moving. When the movement is stopped, it does not detect.
As shown in Figure 4.20, two students are in a sitting position (before starting the detection process) in the 1st picture and the system takes the image for the 1st frame. After some time, as shown in 2nd picture, both of the students have moved and the system has taken the image for the 2nd frame. Afterwards, the system compares the 1st frame with the 2nd frame for detecting movement from the previous position. As both of the students have moved, so the system has detected both of the occupants. And also the system is marked by the green box and black tick mark, which means the system is active as shown in 3rd picture of Figure 4.20.
In Figure 4.20, in the 4th and 5th picture, there is no movement from the left student but the right student has moved to the right and the system takes the image for the 3rd frame. Afterwards, the system compares the 2nd frame with the 3rd frame for detecting movement from the previous position. As no movement from the left student, he is not detected. The right student has moved, so the system has done the occupant detection. And even if only one occupant is detected the system is marked by the green box and black tick mark, which means the system is active as shown in Figure 4.20.

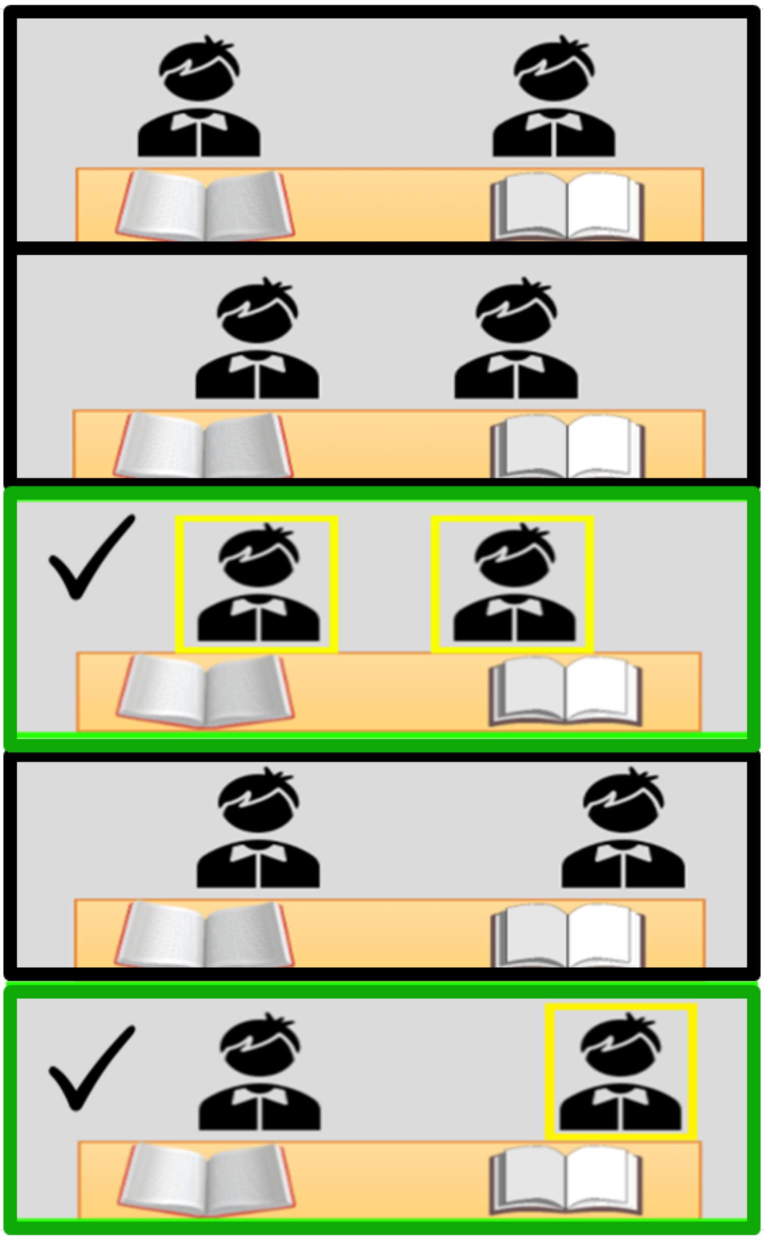

Figure 4.20: Occupant Detection and System Activation

For convenience, the whole working process of the 3rd model is shown in Figure 4.21 by a flow chart:

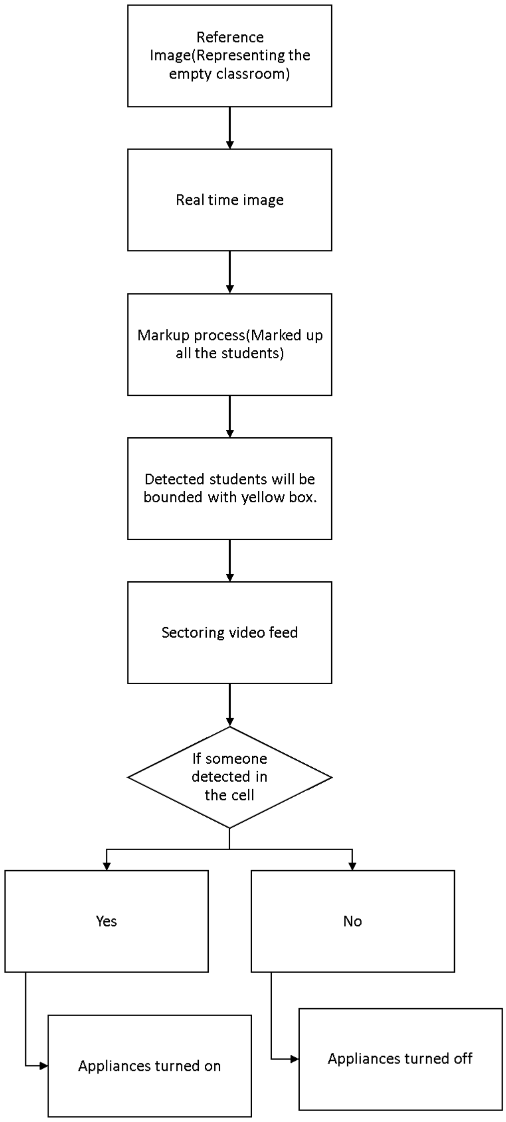

Figure 4.21: Flowchart for the 3rd model

4.4.2 Issues with the MOG2 Algorithm process

The issue with this model is that it only detects for the time when the occupant is moving. When the movement stops, it does not detect. Therefore, if the occupant does not move continuously, detection is compromised and therefore is not usable. For turning on appliances, there should be a continuous movement in this model. However, continuous moving is difficult for a human, and it is not feasible.
As shown in Figure 4.22 1st picture, the system takes this image as the previous frame. Next, after some time, the system takes another frame as the present frame as shown in Figure 4.22 2nd picture. Afterwards, the system compares the present frame with the previous frame for detecting movement from the previous position. As none of the students has moved, so the system has not detected any of the occupants. As a result, the system is marked by the red box and black cross mark, which means the system is deactivated. So occupant detection is not done, no student is detected and the appliances are turned off even if there are two students in the cell as shown in Figure 4.22.

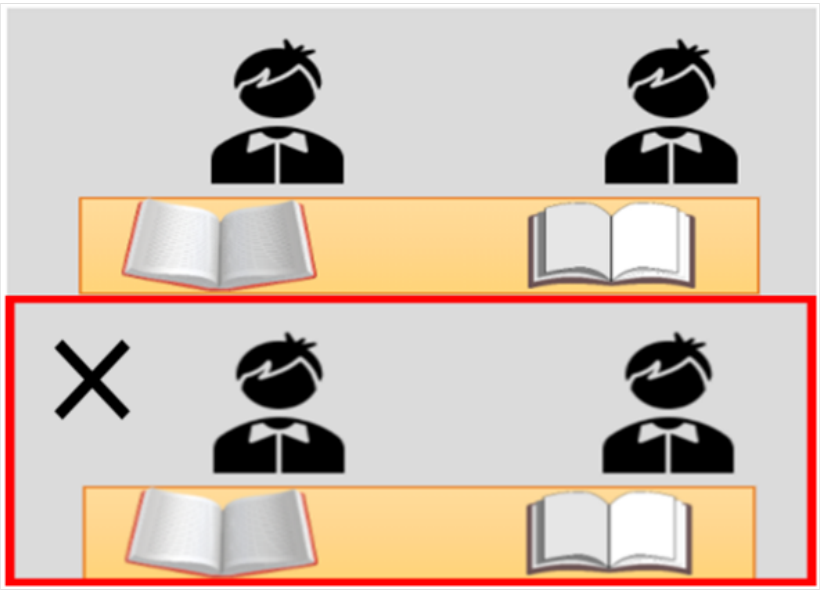

Figure 4.22: Issues with MOG2 Algorithm process

That is why the 4th model which is Background subtraction using the MOG2 algorithm with time buffer is proposed for a better result.

4.5 2nd Approach: MOG2 Algorithm with a time buffer

4.5.1 Occupant Detection process for MOG2 Algorithm with time buffer

In this case, 10 seconds is taken for buffering. This time buffer can be changed according to the demand. For this model, if any occupant comes to any sub-section of the classroom covered by the system, sits or stands there, or does any kind of movement and the time exceeds 10 seconds, then the appliances of that particular subsection turn on. After that, the system again checks for another 10 seconds, if it does not detect any movement in that particular time, then the appliances of that particular subsection turn off.
This model is similar to the 3rd model discussed in the previous chapter with the addition of a time buffer which keeps track of each sub-section for motion detection and starts the appliance only if the motion persists for more than 10 seconds and turns off the appliances only when a sub-section is not detected for a motion for more than 10 seconds. Thus this 4th model eliminates the need for the continuous movement to keep the appliances running.

As shown in Figure 4.23 1st picture, a student is present in a particular sub-section at 12:00:00 o'clock and the presence of the student is detected as shown here, marking him by a yellow box. But as the system has not activated yet, so the sub-section is red and marked by black cross mark meaning the appliances are not turned on yet. The system continuously keeps comparing the exact previous moment's frame with the next moment's frame to detect any kind of movement every 10 seconds to get activated. As in Figure 4.23 2nd picture, the student is still present, so he is detected marking him by a yellow box. Moreover, he has movement in the next 10 seconds meaning before 12:00:10 o'clock, so the system detects the movement and it becomes activated. That is why the sub-section is green and marked by black tick mark meaning the appliances are turned on.

Figure 4.23: Background subtraction using MOG2 algorithm with time buffer (activation)

As shown in Figure 4.24 1st picture, a student is present in a particular sub-section at 12:30:00 o'clock and the presence of the student is detected as shown here, marking him by a yellow box. As the system is activated, so the sub-section is green and marked by black tick mark meaning the appliances are turned on. The system continuously keeps comparing the exact previous moment's frame with the next moment's frame to detect any kind of movement every 10 seconds to get activated. As in Figure 4.24 2nd picture, at 12:00:15 o'clock, after 15 seconds, there is no student present, so no detection is possible. Moreover, as after the next 10 seconds meaning after 12:00:10 o'clock, there is no movement, so the system cannot detect any movement for more than 10 seconds. So it becomes deactivated. That is why the sub-section is red and marked by black cross mark meaning the appliances are turned off.

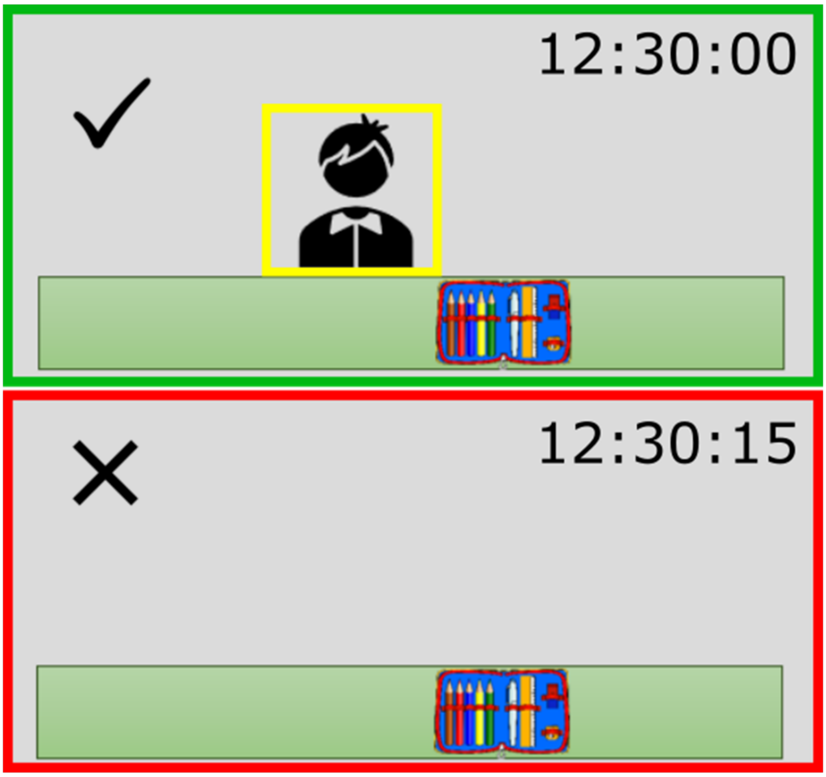

Figure 4.24: Background subtraction using MOG2 algorithm with time buffer (deactivation)

The good thing about this model is the occupant does not need to move continuously for keeping the appliances turned on. If the occupant moves once in the given particular period that is 10 seconds in this case, then the appliances turn on. If no movement in 10 seconds, then the appliances turn off.

For convenience, the whole working process of the 3rd model is shown in Figure 4.25 by a flow chart:

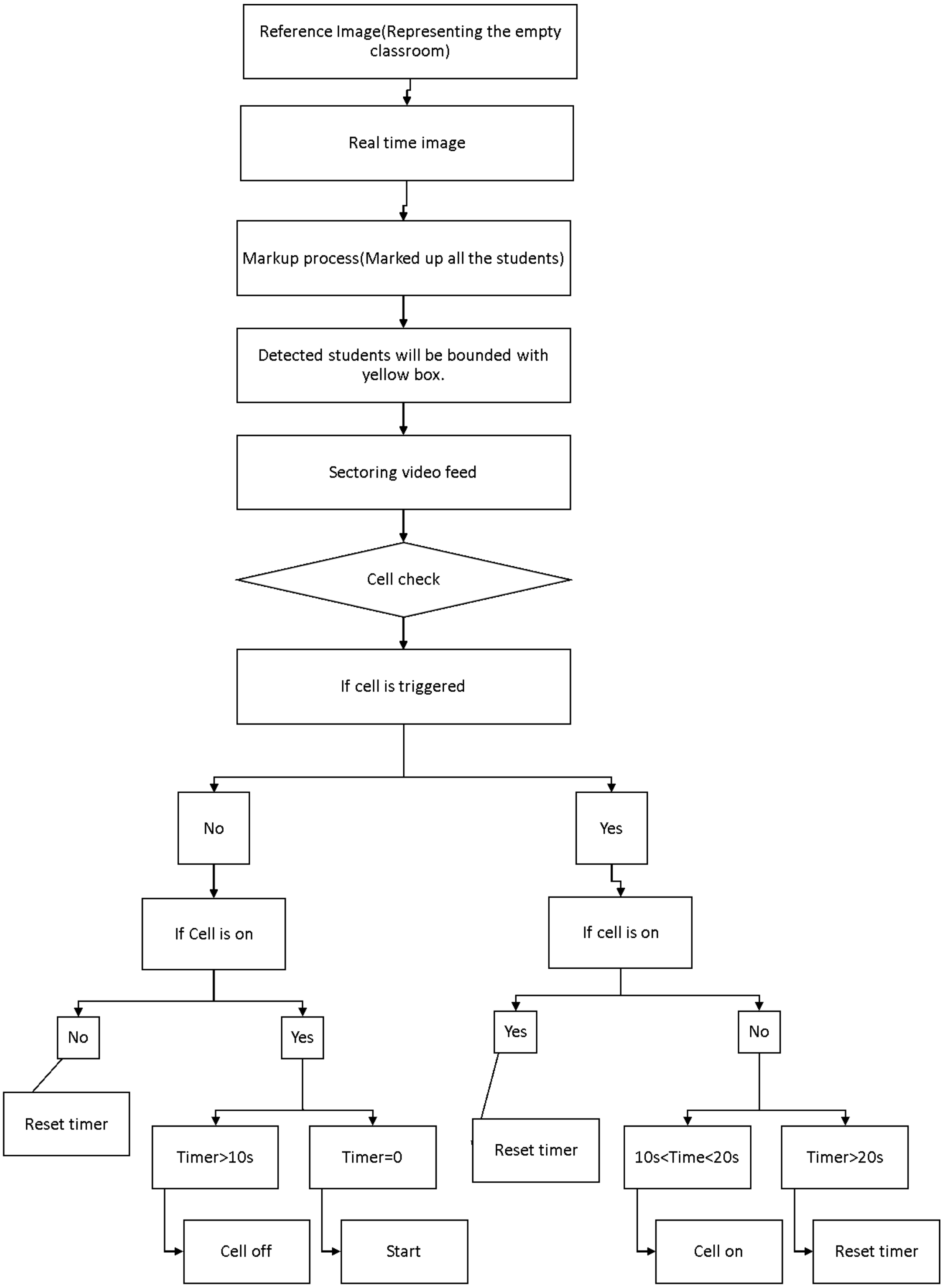

Figure 4.25: Background subtraction using MOG2 algorithm with time buffer (deactivation)

Chapter 5

Hardware and Software Setup

5.1 Hardware Configuration:

Our goal was to build a complete system implemented in a real classroom. But due to the pandemic situation, we couldn't manage a functioning classroom with students. Therefore we tested and debugged the entire system on a test rig that has all the components that would be in the actual classroom but made with portability in mind.
Our test rig consisted of the following components:

1. Primary Processing Unit: The main image processing unit is a Raspberry Pi Model 3 B [30]. Raspberry Pi is inexpensive, with a huge library of computer vision software for it.

2. Camera: For real-time video recording, Pi Cam 2.1 [30] is used.

3. Esp8266 Node MCU: Micro-controller with integrated wifi module for controlling the appliances. It is a very low-cost module with multiple GPIO pins.

4. Relay module: Electro-mechanical switch for handling the mains electricity. The AC side is galvanically isolated from the low voltage logic side.

5. Wireless connectivity: Wifi router providing an access point for both Raspberry Pi and node-MCU.

6. Power supply: USB Power Bank for powering Raspberry Pi with 5V and 2amps.

7. Storage: Micro SD card for Raspberry Pi storage.

8. Power cables: Cables to transfer to Raspberry Pi.

9. Stand: Mount for the Test rig. For the adjustable orientation of the Test rig.

10. PSU: Power supply for ESP8266 node MCU and relay module.

11. Keyboard and mouse: For configuring Raspberry Pi.

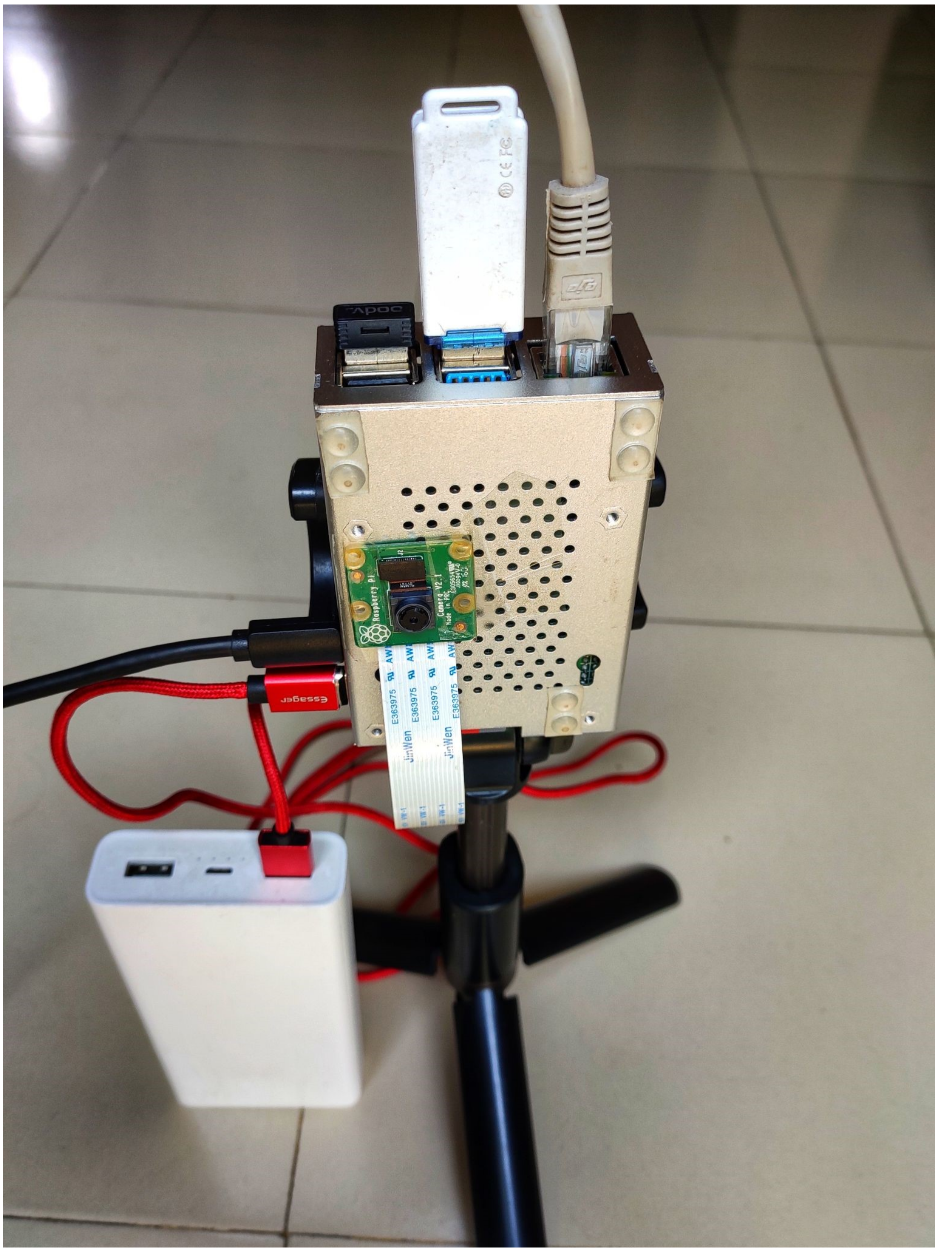

Figure 5.1: Test Rig

12. Screen: Display for Raspberry Pi.

In Appendix B the individual components of the test rig are shown.

5.2 Configuration of Raspberry Pi:

The Raspberry Pi being a fairly inexpensive, small form-factor computer that uses a standard mouse and keyboard to connect to a computer monitor or television. It's a lightweight, portable gadget that lets people of different ages experiment with coding and learn to code in languages like Scratch, Java and Python [31].

5.2.1 Technical Specification:

1. Broadcom BCM2711, 64-bit System on chip @ 1.5GHz
2. 2GB LPDDR4-3200 SDRAM
3. dual-band wifi with bluetooth and BLE
4. Ethernet
5. 4USB ports
6. 40 pin GPIO header
7. 2 micro-HDMI ports supporting 4k60p
8. DSI display port
9. CSI camera port
10. audio and video port
11. standard encoding and decoding of video
12. OpenGL
13. primary storage as Micro-SD card slot
14. USB-C connector
15. power over GPIO
16. Power over Ethernet (PoE)
17. 0-50 degree celcius operational temp

5.3 Configuration of Pi camera:

The Rpi Camera Module v2 is a proprietary Raspberry Pi add-on module with a permanent focused lens and an 8 MP Sony IMX219 image sensor.

5.3.1 Technical Specification

1. Sony IMX219 8-megapixel sensor
2. sensor size: 1/4inch
3. aperture: 2.0
4. FL: 3.04mm
5. AOV: 62.2 degree
6. picture resolution: 3280×2464
7. Supports standard video formats
8. proportions: $25mm \times 24mm \times 9mm$

5.4 Installation of Raspbian:

There are a few different ways to get the Raspberry Pi operating system and use it. The most easy way to use it is to use NOOBS installer.

We set up the OS by downloading the ISO file, formatting the SD card, mounting the ISO, and finally booting the Pi. It consists primarily of the following steps:

1. Downloading the NOOBS package
2. Extracting the downloaded package
3. Copying the contents to the micro-SD card
4. Plugging the micro-SD card in the Raspberry Pi and starting it up
5. Choosing the OS option as Raspbian and clicking the install button
6. Once installed we hit okay and the device should reboot

5.4.1 Configuration of Raspberry Pi Camera Rev 2.1

The first step in configuring the camera is to attach the camera ribbon to the Pi camera ribbon port. Following that, the following steps were taken:

1. Enabling the camera module by typing 'sudo raspi-config' in the command window first
2. Then choose the 'Interfacing options' there
3. Afterward choosing the 'Camera'
4. Pressing 'Yes' to enable and rebooting

5.5 OpenCV and Python installation on the Raspberry Pi:

We had to use the OpenCV library to deal with image and video processing. In our case, we used Python Scripts with OpenCV. The steps below demonstrate how to setup OpenCV within Raspberry Pi and merging it with Python:

1. First of all, we updated the package lists by the command and then typing sudo apt-get update. After that sudo apt-get upgrade and then by typing sudo rpi-update
2. Then we cloned OpenCV from git-hub repo
https://github.com/Itseez/opencv.git cd OpenCV git checkout 3.0.0

3. We installed Python 2.7 afterward. To do so we had to type the following-sudo apt-get install python2.7-dev

4. We also needed pip which is a package management tool, and installs NumPy right away: cd wget https://bootstrap.pypa.io/get-pip.py sudo python get-pip.py

5. Then we simply installed NumPy-pip install numpy

6. Now to compile OpenCV we create a build folder in which the compiled files land: cd /opencv mkdir build cd build cmake

-D CMAKE-BUILD-TYPE
=RELEASE-D CMAKE-INSTALL-PREFIX
=/usr/local
-D INSTALL-PYTHON-EXAMPLES=ON-D INSTALL-C-EXAMPLES
=ON
-D OPENCV-EXTRA-MODULES-PATH
= /opencv-contrib/modules
-D BUILD-EXAMPLES=ON ..

7. Then we compiled

8. After compilation we installed OpenCV:
sudo make install sudo ldconfig

Chapter 6

Result Analysis

At first, we compared four occupant detection models, there were different problems in the first three models. Comparatively, in the last model, we get a better result, which solves all the problems we were facing previously. The model is Background Subtraction (MOG2 algorithm with time buffer).

In the 1st model, there are many problems in detection and also in function. The 2nd model is better than the 1st model but the issue is, it observes a little bit of change of position from its reference image. Whenever an occupant leaves from his previous position and so the chair-table or any object of that place moves, this movement causes a difference from the reference image which is detected by the system. As a result, appliances in that place do not act accordingly. To solve this problem, we have approached the 3rd model. The issue of the 2nd model is solved in the 3rd model. But the problem of the 3rd model is, for turning on and to keep the appliances turned on, there should be a continuous movement. But continuous moving is difficult for an occupant and so it is not feasible also. Therefore, we have come up with the idea of the 4th model for solving this problem.

In the 4th model, we have solved all the problems of the previous three models. There is no problem with the detection and the function. And it does not need continuous movement for keeping the appliances turned on.This model builds the cost-effective,user friendly and customizable system. So as among all these models, we get the best result from the 4th method, we are proposing this one, defined as- Background Subtraction (MOG2 algorithm with time buffer).

We are therefore attaching a comparison table for all four approaches we proposed here.

Methods	1	2	3	4
Proper detection	No	Yes	Yes	Yes
Detection of the only moveable object	No	No	Yes	Yes
No. Of dependency on the fixed frame	No	No	Yes	Yes
Can comply with unexpected changes	No	No	Yes	Yes
Stability	Yes	Yes	No	Yes
Period control	No	No	No	Yes

Table 6.1: Comparison of Four Models

We have discussed three states of the system. The first one is system inactive, which means the system is not active yet. The power consumption for this state is the highest. The second state is with the system active (management off), this means the system is active. But the management off means the student can seat anywhere in the classroom, and the system will work for that particular place. When management is off, the system keeps the appliances of a sub-section on if it detects any movement every 10 seconds regardless of where the row is placed and if any seat in the front row is vacant or not. The third state is with the system active (management on), this will force the students two fill up the rows of the classroom serially. When the management is on, the system will detect if there is any vacant seat starting from the 1st row and if any seat is vacant and still any student is sitting in the back row, then automatically the appliances of that particular sub-section will be turned off. If any student skips the vacant space and goes to the back row of the classroom the system will not detect that place and the appliances of that place will not be turned on.

Here we are attaching all the tables with all the data which we used for generating the graphs and calculating the results.

		With system inactive				
Time	No. of student	Light on	Fan on	Power (Light)	Power (Fan)	Total power
8:00:00 AM	2	18	8	720	800	1520
8:01:00 AM	3	18	8	720	800	1A520
8:02:00 AM	5	18	8	720	800	1520
8:03:00 AM	6	18	8	720	800	1520
8:04:00 AM	6	18	8	720	800	1520
8:05:00 AM	10	18	8	720	800	1520
8:06:00 AM	15	18	8	720	800	1520
8:07:00 AM	18	18	8	720	800	1520
8:08:00 AM	22	18	8	720	800	1520
8:09:00 AM	24	18	8	720	800	1520
8:10:00 AM	26	18	8	720	800	1520
8:11:00 AM	29	18	8	720	800	1520
8:12:00 AM	34	18	8	720	800	1520
8:13:00 AM	42	18	8	720	800	1520
8:14:00 AM	45	18	8	720	800	1520
8:15:00 AM	45	18	8	720	800	1520
8:16:00 AM	45	18	8	720	800	1520
8:17:00 AM	46	18	8	720	800	1520
8:18:00 AM	47	18	8	720	800	1520
8:19:00 AM	48	18	8	720	800	1520
8:20:00 AM	52	18	8	720	800	1520
8:21:00 AM	51	18	8	720	800	1520
8:22:00 AM	51	18	8	720	800	1520
8:23:00 AM	53	18	8	720	800	1520
8:24:00 AM	54	18	8	720	800	1520
8:25:00 AM	55	18	8	720	800	1520
8:26:00 AM	55	18	8	720	800	1520
8:27:00 AM	56	18	8	720	800	1520
8:28:00 AM	56	18	8	720	800	1520
8:29:00 AM	56	18	8	720	800	1520
8:30:00 AM	56	18	8	720	800	1520
8:31:00 AM	54	18	8	720	800	1520
8:32:00 AM	53	18	8	720	800	1520
8:33:00 AM	53	18	8	720	800	1520
8:34:00 AM	51	18	8	720	800	1520
8:35:00 AM	50	18	8	720	800	1520
8:36:00 AM	47	18	8	720	800	1520

		With system inactive				
Time	No. of student	Light on	Fan on	Power (Light)	Power (Fan)	Total power
8:37:00 AM	47	18	8	720	800	1520
8:38:00 AM	48	18	8	720	800	1520
8:39:00 AM	47	18	8	720	800	1520
8:40:00 AM	46	18	8	720	800	1520
8:41:00 AM	46	18	8	720	800	1520
8:42:00 AM	45	18	8	720	800	1520
8:43:00 AM	45	18	8	720	800	1520
8:44:00 AM	45	18	8	720	800	1520
8:45:00 AM	45	18	8	720	800	1520
8:46:00 AM	46	18	8	720	800	1520
8:47:00 AM	46	18	8	720	800	1520
8:48:00 AM	47	18	8	720	800	1520
8:49:00 AM	47	18	8	720	800	1520
8:50:00 AM	48	18	8	720	800	1520
8:51:00 AM	50	18	8	720	800	1520
8:52:00 AM	52	18	8	720	800	1520
8:53:00 AM	52	18	8	720	800	1520
8:54:00 AM	52	18	8	720	800	1520
8:55:00 AM	52	18	8	720	800	1520
8:56:00 AM	52	18	8	720	800	1520
8:57:00 AM	53	18	8	720	800	1520
8:58:00 AM	53	18	8	720	800	1520
8:59:00 AM	53	18	8	720	800	1520
9:00:00 AM	54	18	8	720	800	1520
9:01:00 AM	54	18	8	720	800	1520
9:02:00 AM	54	18	8	720	800	1520
9:03:00 AM	54	18	8	720	800	1520
9:04:00 AM	56	18	8	720	800	1520
9:05:00 AM	57	18	8	720	800	1520
9:06:00 AM	60	18	8	720	800	1520
9:07:00 AM	60	18	8	720	800	1520
9:08:00 AM	60	18	8	720	800	1520
9:09:00 AM	60	18	8	720	800	1520
9:10:00 AM	60	18	8	720	800	1520
9:11:00 AM	57	18	8	720	800	1520
9:12:00 AM	55	18	8	720	800	1520
9:13:00 AM	45	18	8	720	800	1520

		With system inactive				
Time	No. of student	Light on	Fan on	Power (Light)	Power (Fan)	Total power
9:14:00 AM	31	18	8	720	800	1520
9:15:00 AM	25	18	8	720	800	1520
9:16:00 AM	10	18	8	720	800	1520
9:17:00 AM	5	18	8	720	800	1520
9:18:00 AM	4	18	8	720	800	1520
9:19:00 AM	0	18	8	720	800	1520
9:20:00 AM	0	18	8	720	800	1520

Table 6.2: With system inactive

Here the average power consumption of a classroom for one period (Ninety minutes) in system state with system inactive is 1520 Watt.

		With system active (management off)				
Time	No. of student	Light on	Fan on	Power (Light)	Power (Fan)	Total power
8:00:00 AM	2	1	1	40	100	140
8:01:00 AM	3	2	1	80	100	180
8:02:00 AM	5	2	1	80	100	180
8:03:00 AM	6	3	1	120	100	220
8:04:00 AM	6	3	1	120	100	220
8:05:00 AM	10	5	2	200	200	400
8:06:00 AM	15	7	3	280	300	580
8:07:00 AM	18	10	4	400	400	800
8:08:00 AM	22	11	5	440	500	940
8:09:00 AM	24	12	5	480	500	980
8:10:00 AM	26	14	6	560	600	1160
8:11:00 AM	29	14	6	560	600	1160
8:12:00 AM	34	14	6	560	600	1160
8:13:00 AM	42	15	7	600	700	1300
8:14:00 AM	45	15	7	600	700	1300
8:15:00 AM	45	15	7	600	700	1300
8:16:00 AM	45	15	7	600	700	1300
8:17:00 AM	46	15	7	600	700	1300
8:18:00 AM	47	14	6	560	600	1160
8:19:00 AM	48	13	6	520	600	1120
8:20:00 AM	52	16	7	640	700	1340
8:21:00 AM	51	16	7	640	700	1340
8:22:00 AM	51	16	7	640	700	1340
8:23:00 AM	53	17	8	680	800	1480
8:24:00 AM	54	17	8	680	800	1480
8:25:00 AM	55	17	8	680	800	1480
8:26:00 AM	55	17	8	680	800	1480
8:27:00 AM	56	17	8	680	800	1480
8:28:00 AM	56	18	8	720	800	1520
8:29:00 AM	56	18	8	720	800	1520
8:30:00 AM	56	18	8	720	800	1520
8:31:00 AM	54	16	7	640	700	1340
8:32:00 AM	53	15	7	600	700	1300
8:33:00 AM	53	15	7	600	700	1300
8:34:00 AM	51	15	7	600	700	1300
8:35:00 AM	50	15	7	600	700	1300
8:36:00 AM	47	14	6	560	600	1160

		With system active (management off)				
Time	No. of student	Light on	Fan on	Power (Light)	Power (Fan)	Total power
8:37:00 AM	47	14	6	560	600	1160
8:38:00 AM	48	15	7	600	700	1300
8:39:00 AM	47	15	7	600	700	1300
8:40:00 AM	46	14	6	560	600	1160
8:41:00 AM	46	14	6	560	600	1160
8:42:00 AM	45	14	6	560	600	1160
8:43:00 AM	45	13	6	520	600	1120
8:44:00 AM	45	13	6	520	600	1120
8:45:00 AM	45	14	6	560	600	1160
8:46:00 AM	46	13	6	520	600	1120
8:47:00 AM	46	14	6	560	600	1160
8:48:00 AM	47	14	6	560	600	1160
8:49:00 AM	47	14	6	560	600	1160
8:50:00 AM	48	15	7	600	700	1300
8:51:00 AM	50	15	7	600	700	1300
8:52:00 AM	52	15	7	600	700	1300
8:53:00 AM	52	15	7	600	700	1300
8:54:00 AM	52	15	7	600	700	1300
8:55:00 AM	52	15	7	600	700	1300
8:56:00 AM	52	15	7	600	700	1300
8:57:00 AM	53	14	6	560	600	1160
8:58:00 AM	53	16	7	640	700	1340
8:59:00 AM	53	16	7	640	700	1340
9:00:00 AM	54	16	7	640	700	1340
9:01:00 AM	54	16	7	640	700	1340
9:02:00 AM	54	16	7	640	700	1340
9:03:00 AM	54	15	7	600	700	1300
9:04:00 AM	56	15	7	600	700	1300
9:05:00 AM	57	17	8	680	800	1480
9:06:00 AM	60	18	8	720	800	1520
9:07:00 AM	60	18	8	720	800	1520
9:08:00 AM	60	18	8	720	800	1520
9:09:00 AM	60	18	8	720	800	1520
9:10:00 AM	60	18	8	720	800	1520
9:11:00 AM	57	18	8	720	800	1520
9:12:00 AM	55	17	8	680	800	1480
9:13:00 AM	45	13	6	520	600	1120

		With system active (management off)				
Time	No. of student	Light on	Fan on	Power (Light)	Power (Fan)	Total power
9:14:00 AM	31	9	4	360	400	760
9:15:00 AM	25	6	3	240	300	540
9:16:00 AM	10	3	1	120	100	220
9:17:00 AM	5	2	1	80	100	180
9:18:00 AM	4	2	1	80	100	180
9:19:00 AM	0	0	0	0	0	0
9:20:00 AM	0	0	0	0	0	0

Table 6.3: With system active(Management off)

Here the average power consumption of a classroom for one period (Ninety minutes) in system state with system active (Management off) is 1117 Watt.

		With system active (management on)				
Time	No. of student	Light on	Fan on	Power (Light)	Power (Fan)	Total power
8:00:00 AM	2	1	1	40	100	140
8:01:00 AM	3	1	1	40	100	140
8:02:00 AM	5	2	1	80	100	180
8:03:00 AM	6	2	1	80	100	180
8:04:00 AM	6	2	1	80	100	180
8:05:00 AM	10	3	2	120	200	320
8:06:00 AM	15	5	2	200	200	400
8:07:00 AM	18	6	3	240	300	540
8:08:00 AM	22	7	3	280	300	580
8:09:00 AM	24	8	4	320	400	720
8:10:00 AM	26	8	4	320	400	720
8:11:00 AM	29	9	4	360	400	760
8:12:00 AM	34	11	5	440	500	940
8:13:00 AM	42	13	6	520	600	1120
8:14:00 AM	45	14	6	560	600	1160
8:15:00 AM	45	14	6	560	600	1160
8:16:00 AM	45	14	6	560	600	1160
8:17:00 AM	46	14	7	560	700	1260
8:18:00 AM	47	15	7	600	700	1300
8:19:00 AM	48	15	7	600	700	1300
8:20:00 AM	52	16	7	640	700	1340
8:21:00 AM	51	16	7	640	700	1340
8:22:00 AM	51	16	7	640	700	1340
8:23:00 AM	53	16	8	640	800	1440
8:24:00 AM	54	17	8	680	800	1480
8:25:00 AM	55	17	8	680	800	1480
8:26:00 AM	55	17	8	680	800	1480
8:27:00 AM	56	17	8	680	800	1480
8:28:00 AM	56	17	8	680	800	1480
8:29:00 AM	56	17	8	680	800	1480
8:30:00 AM	56	17	8	680	800	1480
8:31:00 AM	54	17	8	680	800	1480
8:32:00 AM	53	16	8	640	800	1440
8:33:00 AM	53	16	8	640	800	1440
8:34:00 AM	51	16	7	640	700	1340
8:35:00 AM	50	15	7	600	700	1300
8:36:00 AM	47	15	7	600	700	1300

		With system active (management on)				
Time	No. of student	Light on	Fan on	Power (Light)	Power (Fan)	Total power
8:37:00 AM	47	15	7	600	700	1300
8:38:00 AM	48	15	7	600	700	1300
8:39:00 AM	47	15	7	600	700	1300
8:40:00 AM	46	14	7	560	700	1260
8:41:00 AM	46	14	7	560	700	1260
8:42:00 AM	45	14	6	560	600	1160
8:43:00 AM	45	14	6	560	600	1160
8:44:00 AM	45	14	6	560	600	1160
8:45:00 AM	45	14	6	560	600	1160
8:46:00 AM	46	14	7	560	700	1260
8:47:00 AM	46	14	7	560	700	1260
8:48:00 AM	47	15	7	600	700	1300
8:49:00 AM	47	15	7	600	700	1300
8:50:00 AM	48	15	7	600	700	1300
8:51:00 AM	50	15	7	600	700	1300
8:52:00 AM	52	16	7	640	700	1340
8:53:00 AM	52	16	7	640	700	1340
8:54:00 AM	52	16	7	640	700	1340
8:55:00 AM	52	16	7	640	700	1340
8:56:00 AM	52	16	7	640	700	1340
8:57:00 AM	53	16	8	640	800	1440
8:58:00 AM	53	16	8	640	800	1440
8:59:00 AM	53	16	8	640	800	1440
9:00:00 AM	54	17	8	680	800	1480
9:01:00 AM	54	17	8	680	800	1480
9:02:00 AM	54	17	8	680	800	1480
9:03:00 AM	54	17	8	680	800	1480
9:04:00 AM	56	17	8	680	800	1480
9:05:00 AM	57	18	8	720	800	1520
9:06:00 AM	60	18	8	720	800	1520
9:07:00 AM	60	18	8	720	800	1520
9:08:00 AM	60	18	8	720	800	1520
9:09:00 AM	60	18	8	720	800	1520
9:10:00 AM	60	18	8	720	800	1520
9:11:00 AM	57	18	8	720	800	1520
9:12:00 AM	55	17	8	680	800	1480
9:13:00 AM	45	14	6	560	600	1160

		With system active (management on)				
Time	No. of student	Light on	Fan on	Power (Light)	Power (Fan)	Total power
9:14:00 AM	31	10	5	400	500	900
9:15:00 AM	25	8	4	320	400	720
9:16:00 AM	10	3	2	120	200	320
9:17:00 AM	5	2	1	80	100	180
9:18:00 AM	4	2	1	80	100	180
9:19:00 AM	0	0	0	0	0	0
9:20:00 AM	0	0	0	0	0	0

Table 6.4: With system active(Management on)

Here the average power consumption of a classroom for one period (Ninety minutes) in system state with system active (Management on) is 1108 Watt.

	With system inactive	With system active (management off)	With system active (management on)
Daily active hour for one classroom	9	7.5	7.5
Per day energy consumption (wh) for one classroom	13680	8375.925926	8311.111111
Per day energy consumption (unit) for one classroom	13.68	8.375925926	8.311111111
Energy consumption (unit) for a year (200 days) for one classroom	2736	1675.185185	1662.222222
Electricity bill for a year for one classroom	31354.56	19197.62222	19049.06667
Per day Saving in energy for one classroom	0	5.304074074	5.368888889
Per day Saving in taka for a year for one classroom	0	60.78468889	61.52746667
Per day Saving in taka for the institution	0	12156.93778	12305.49333
Saving in taka for the institution for a year	0	2431387.556	2461098.667

Table 6.5: Comparison table

A table is given here showing approximate power consumption of a classroom for one period (Ninety minutes) of three states of the system from the collected data according to Table 6.2,6.2 and 6.4. It is observed that power consumption is lowest for the system state with system active management on.

System State	Total Power(W)
With system inactive	1520
With system active (management off)	1117
With system active (management on)	1108

Table 6.6: Approximate Power Consumption for One period

We implement the system in a classroom and collect energy consumption data according to Table 6.2, 6.3 and 6.4 for one period consisting of ninety minutes. The consumption of power with system active is 1108 kW, which is very lower than the system inactive one which is 1520 kW. In the automated system implemented classroom, the energy consumption with the management system on is 1108 KW which is lower than the management system off having power of 1117 kW.

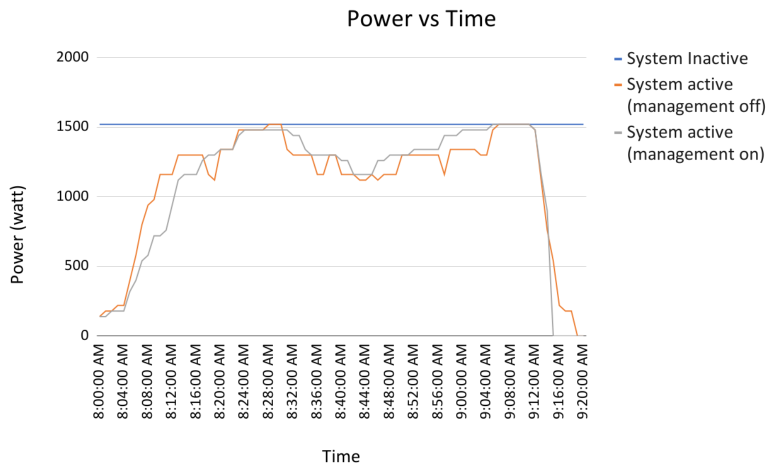

Figure 6.1: Total Power vs Time graph

We represent our result in the 2-D graph shown in Figure 6.1. The Y-axis of the graph shows the power in terms of Watt and the X-axis shows the time. The result shows that the power with system active is lower than system inactive.

Here we are attaching the table which contains all the data of Energy Consumption, Savings and Bills according to increased number of classroom.

Classroom number	Consumption(Unit)	Saving(in Taka)	Bill(in Taka)
1	1662.22	12305.4933	19049.0667
20	33244.4	246109.866	380981.334
40	66488.8	492219.732	761962.668
60	99733.2	738329.598	1142944.002
80	132977.6	984439.464	1523925.336
100	166222	1230549.33	1904906.67
120	199466.4	1476659.196	2285888.004
140	232710.8	1722769.062	2666869.338
160	265955.2	1968878.928	3047850.672
180	299199.6	2214988.794	3428832.006
200	332444	2461098.66	3809813.34

Table 6.7: Estimation of energy consumption, Savings and Bills according to increased number of classroom

We implement the system in one classroom and increase the number of classrooms up to two-hundred assuming an institution has two-hundred classrooms and collect the energy consumption in unit for a year according to Table 6.7 which is 3.5 hundred thousand unit according to increased classroom number.

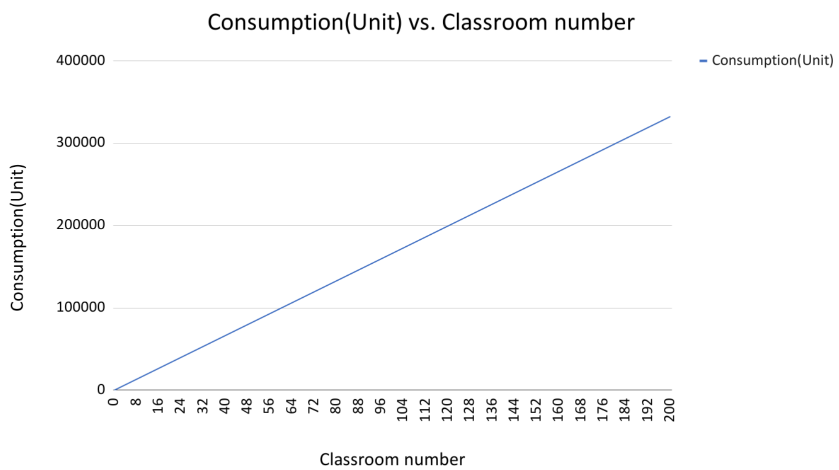

Figure 6.2: Consumption (unit) vs Classroom number

As observed in Figure 6.2, the energy consumption increases proportionally with increased classroom numbers. In the figure, the Y-axis represents energy consumption in Unit and X-axis represents the classroom number.

We implement the system in one classroom and increase the number of classrooms up to two-hundred assuming an institution has two-hundred classrooms and collect the electricity bill in taka for a year according to Table Table 6.7 which is 4.5 million taka according to increased classroom number.

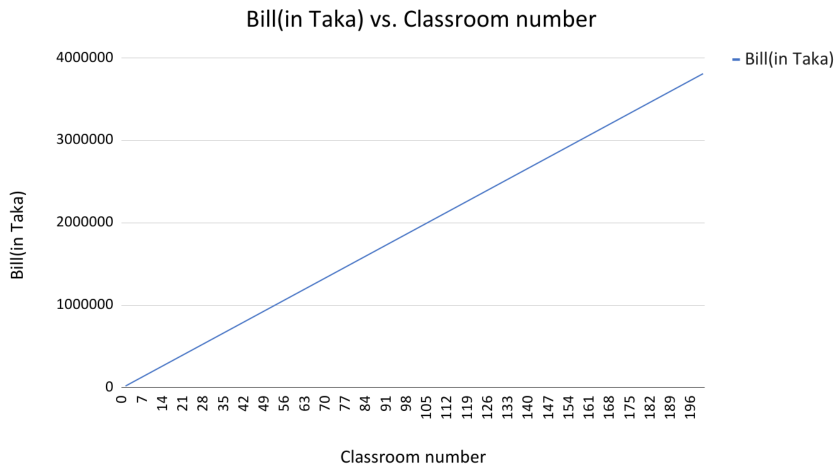

Figure 6.3: Electricity bill (in Taka) vs Classroom number

As observed in Figure 6.3 the electricity bill increases proportionally with the increased classroom number. In the figure, Y-axis represents the Bill in Taka and X-axis represents the classroom number.

We implement the system in one classroom and increase the number of classrooms up to two-hundred assuming an institution has two-hundred classrooms and collect the savings in taka for a year according to Table 6.7 which is 3 million Taka according to increased classroom number.

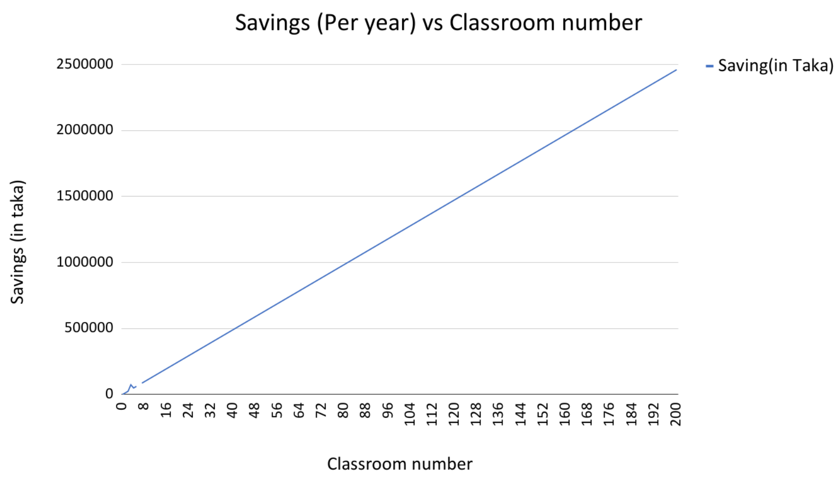

Figure 6.4: Saving per year vs Classroom number

As observed in Figure 6.4 the saving per year increases proportionally with the increased classroom number. The Y-axis shows the savings in Taka and the X axis shows the classroom number.

We take the energy consumption in unit for a year of an institution having two-hundred classrooms according to Table 6.7 which is 3.29 hundred thousand unit and take the saving and electricity bill in taka according to increased energy consumption. We can see, the electricity bill is 3.8 million taka whereas the saving is 2.5 million taka.

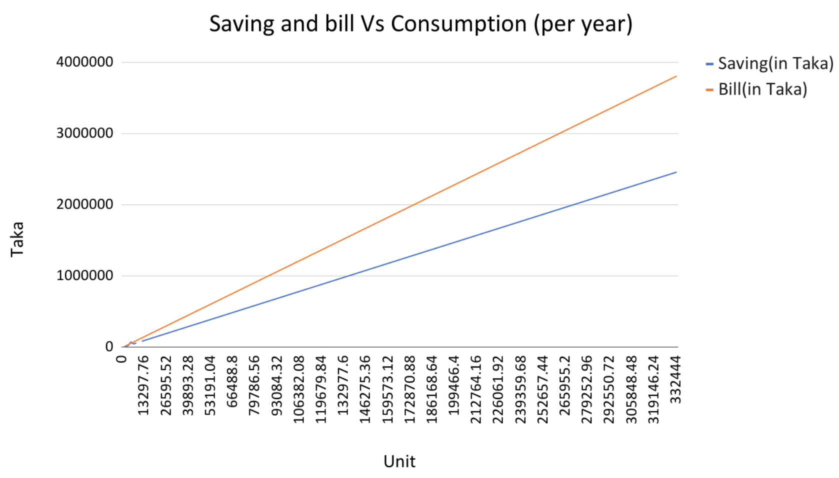

Figure 6.5: Saving and Bill vs Consumption per year

As observed in Figure 6.5 the saving and bill per year increases proportionally with increased energy consumption. In the figure, Y-axis represents the saving and bill in taka and X-axis represents energy consumption for each year.

We take the electricity bill in taka for a year of an institution having two-hundred classrooms according to Table 6.7 which is 3.8 million taka and take the saving in taka according to increased electricity bill which is 2.5 million taka.

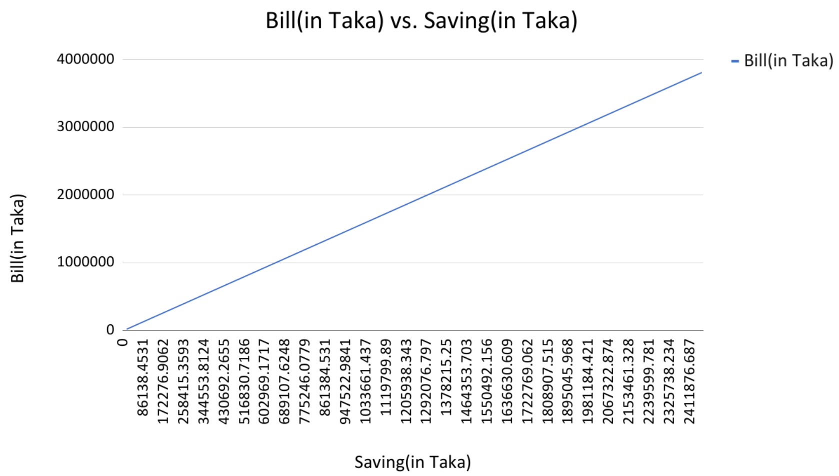

Figure 6.6: Bill (in Taka) vs Saving (in Taka)

As observed in Figure 6.6 the saving increases proportionally with the increased electricity bill. In the figure, the Y-axis represents the bill in taka and X-axis represents saving in taka.

We implement our system in one classroom using the 4th model that is Background Subtraction using the MOG2 algorithm with a time buffer and get the expected output.

A classroom consisting of sixteen tables and each table is considered as one cell. The system first takes the image before activating is shown in Figure 6.7.

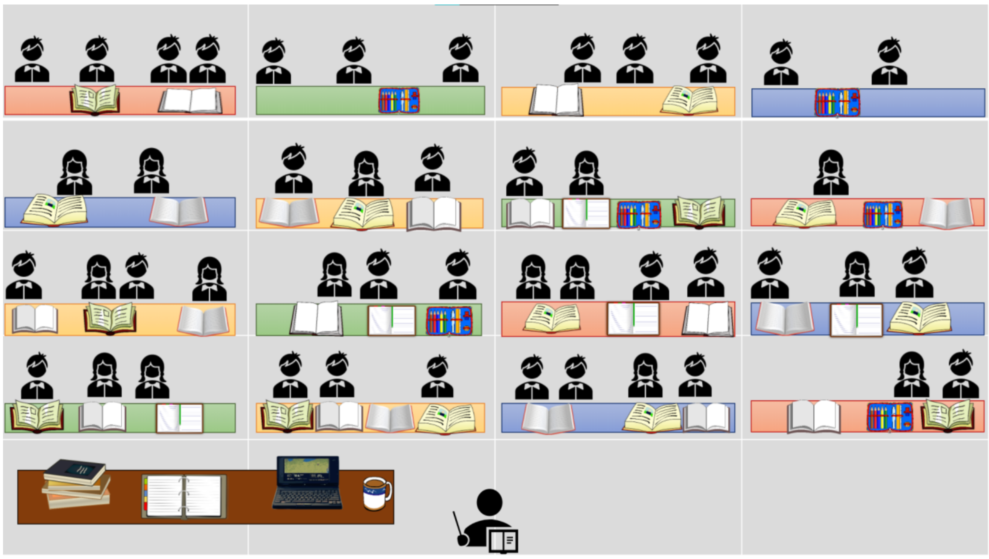

Figure 6.7: Scenario 1 without System Activation

Next, Figure 6.8 shows that the system is activated and as there are students in each of the cells, so all the cells are marked by green boxes and black tick mark meaning the system is active for that particular cell.

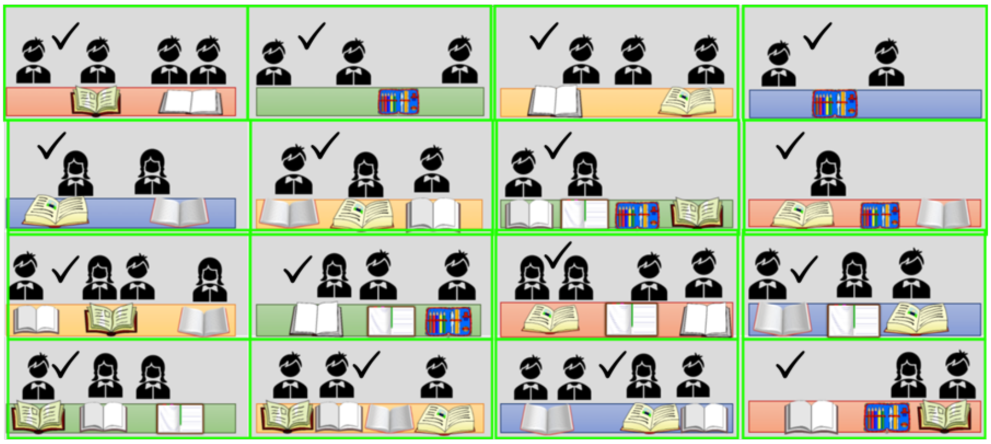

Figure 6.8: Scenario 1 with System Activation

Now, after some time, in some tables, there are no students and again the system first takes the image before activating which is shown in Figure 6.9.

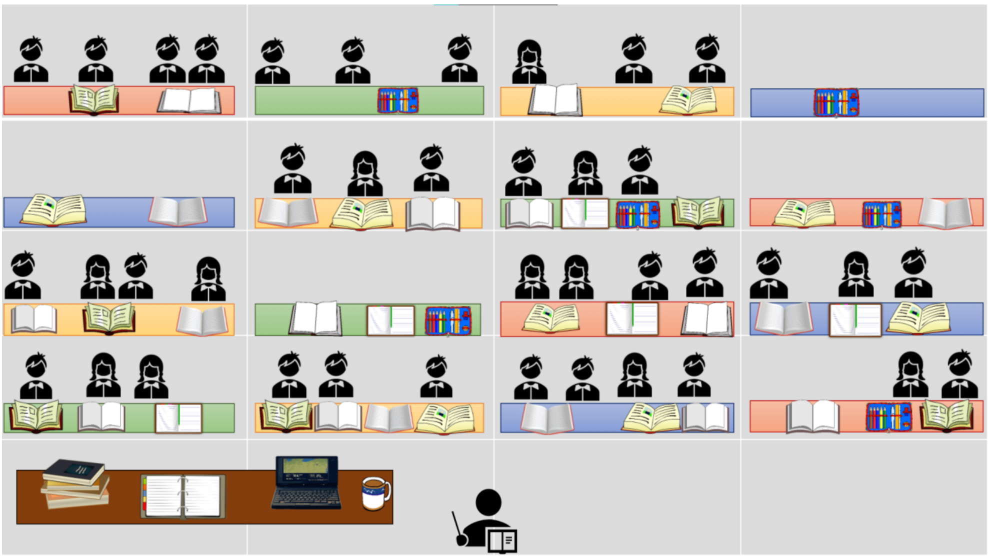

Figure 6.9: Scenario 2 without System Activation

Next, the system is activated and as there is no student in few cells, so the system can not detect any occupant in those cells and as a result for those cells, the system is not active as shown in Figure 6.10. That is why the cells that have occupants in them are marked by green boxes meaning the system is active for that particular cell and the cells that have no occupant in them are marked by red boxes and black cross mark meaning the system is not active for that particular cell.

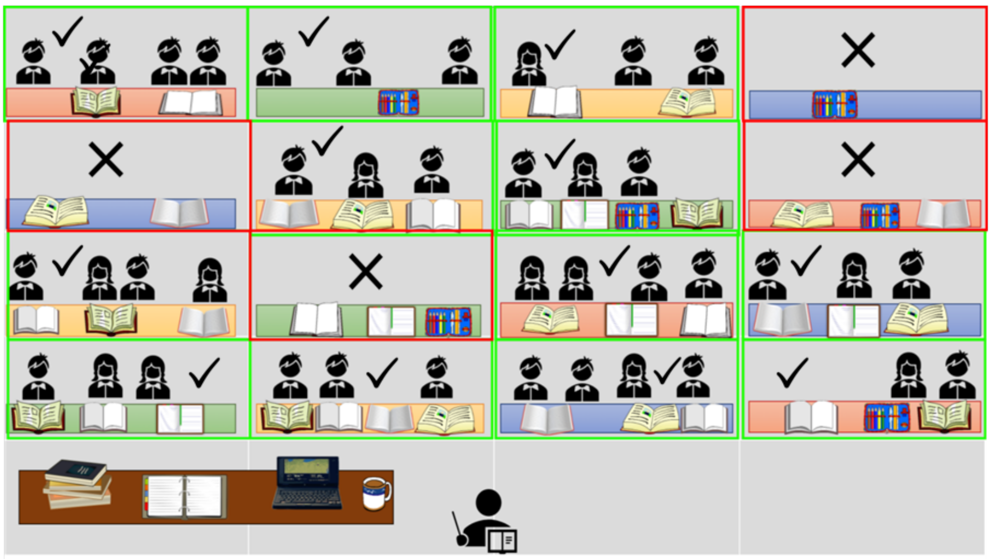

Figure 6.10: Scenario 2 with System Activation

A table is given here showing the approximate cost of one device for one classroom for applying the system:

Components	Price (BDT)
Raspberry pi 4 (2GB)	2,975
Pi cam	315
Sd card	300
Power supply	250
Esp8266	220
Relay module	235
Miscellaneous	100
Total	4,395

Table 6.8: Approximate cost

As a device can run for infinite time and we can see the cost is very low compared to how much electric bill is charged every month for one classroom, so this system is very cost-friendly and user-friendly as we were expecting.

So, as per our research objective and outcome, we have successfully compared different occupant detection models. To find the most effective occupant detection model, we have come to this decision that- the 4th model is the most effective occupant detection model. We have developed this model by adding time buffer to MOG2 algorithm to develop an energy efficient intelligent classroom. From the above analysis, it is observed that if our system is implemented in a classroom, then we can build an energy-efficient intelligent classroom which is cost-effective, user-friendly and customizable.

Chapter 7

Conclusion and Future work

7.1 Conclusion

In this book, a model is proposed that uses image processing to monitor the movement and activity of students in a classroom and acts accordingly to reduce energy consumption. The four detection models are also discussed in detail and the outputs that can help maintain a better learning environment in this technology-dependent world.

Here an effective detection model is proposed for the detection of occupants in the classroom and then switch on/off the appliances accordingly. The four detection models are tested in our research to find the most effective detection model that takes into account all the unregulated variables in the classroom that hinders reliable detection. The model works properly and reduces a huge amount of energy consumption. This system can be implemented in many places other than classrooms like libraries, houses, banks, hospitals, laboratories, offices, industries, or any other place where occupants are present and appliances are needed to turn on/off frequently. As the existing electrical wiring is used for the appliances, no extra wiring is needed. The system just needs to be integrated with the switchboard and connected to the existing CCTV camera or raspberry pi-cam with the Raspberry Pi.

As this system can be implemented in the existing surveillance system, so the additional cost is not high. Moreover, the equipment is low cost as well. So this system introduces design and implementation at a low cost and provides a flexible and wireless solution to the high energy consumption problem. It is hoped and believed that this ideology will prove to be effective in improving the classroom condition if it is implemented practically. Such a system is desirable in the current scenario to help in energy saving and making people aware of it.

7.2 Future work

So far as we have discussed, we have done the detection in Open CV using background subtraction. The fourth model which is applied here detects the motion. Here 10 sec is taken for buffering. So the occupant needs to move in every 10 seconds.

The problem is if the occupant doesn't move in every 10 seconds then the system will not be active. As seen in Figure 7.1 1st picture, an occupant is present in a particular sub-section at 12:30:00 o'clock and this subsection is marked by a green rectangular box and black tick mark meaning the system has detected the presence of an occupant or movement and this presence or movement persists for more than 10 seconds to make the system active.

But the problem is if the occupant doesn't show any movement in every 10 sec then the system will not detect any occupant and it will be deactivated. As seen in Figure 7.1 2nd picture, after 11 seconds, at 12:30:11 o'clock, the occupants are in the same place. No movement is detected. So, that subsection is marked by a red rectangular box and black cross mark meaning the system can not detect any occupant or movement there after 11 seconds and so, the appliances are deactivated.

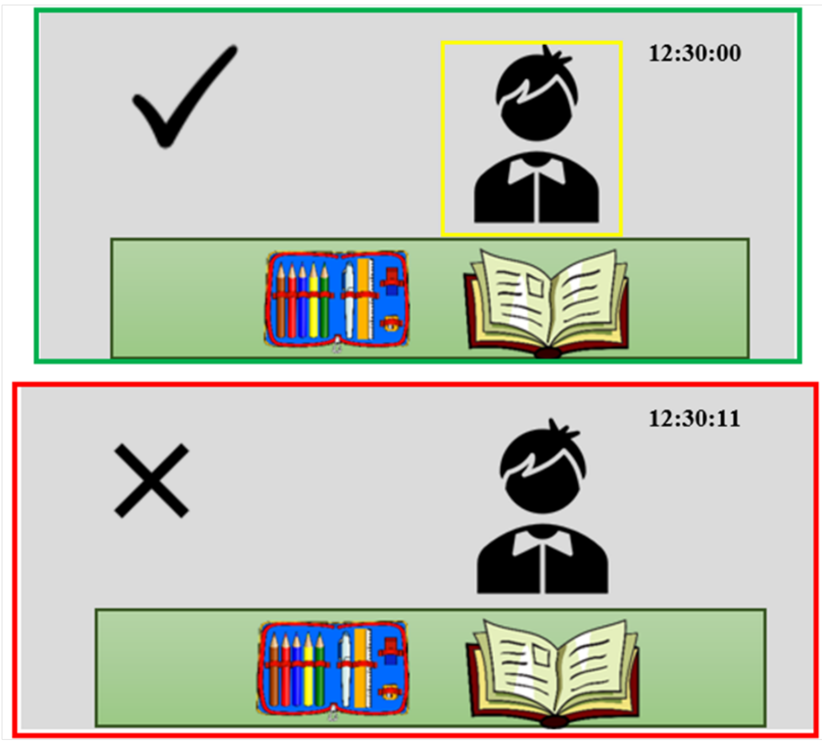

Figure 7.1: Issue with MOG2 Algorithm with Time Buffer

So, the better approach to it can be person detection. We can use machine learning to detect a single person and track where he stands and sits and turn on the lights and fans of that position. Machine learning is a part of artificial intelligence. It's a learning of computer algorithms that improve naturally through experience [27]. If machine learning is used here it will detect the single person not the movement which will solve the problem of the model which is used by us.

Another improvement can be gesture recognition. Gesture recognition is a subject of software engineering and language innovation to interpret human signals via mathematical calculations. Motions can begin from any real movement or state yet normally start from the face or hand [27]. When a person feels like turning on/off the light/fan above him, he will give a simple hand gesture and the system will respond accordingly. Another idea is cloud syncing, the system will take weather data like the temperature, humidity, etc, and change its function accordingly.

Moreover, future work can be centered around implementing applications that can help in automating a whole department or an educational organization. A similar model also can be executed in making a home/office/emergency ward/any jam-packed spot's automation framework that will be cost-effective. Diverse technological advances can help in making the system more feasible and adaptable. Machine language can be added to this model. This will make the system able to decide based on the temperature, humidity, etc atmospheric conditions of the classroom and also on the ambient light of the particular location. As our proposed system is still in the experimental stage, further development can be done if our proposed system is implemented practically.

REFERENCES

[1] H. Haijing and C. Gang, “Study on energy saving lighting of classroom based on cirtopic,” in *2010 International Conference on Digital Manufacturing Automation*, 2010, pp. 471–474.

[2] C. Paul, A. Ganesh, and C. Sunitha, “An iot-based smart classroom,” in *International Conference on Computer Networks and Communication Technologies Lecture Notes on Data Engineering and Communications Technologies*, 2018, p. 9–14.

[3] S. K. Vishwakarma, P. Upadhyaya, B. Kumari, and A. K. Mishra, “Smart energy efficient home automation system using iot,” in *2019 4th International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU)*, 2019, pp. 1–4.

[4] D. Babu and A. Siridhara, “Energy efficient home automation system,” *International Journal for Scientific Research & Development*, vol. 4, no. 4, pp. 596–599, 2016.

[5] S. Khedkar and G. Malwatkar, “Using raspberry pi and gsm survey on home automation,” in *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*, 2016, pp. 758–761.

[6] D. Ganiger, K. A. Patil, P. Patil, and M. Anandhalli, “Automatic control of power supply in classroom using image processing,” in *2017 International Conference on Recent Advances in Electronics and Communication Technology (ICRAECT)*, 2017, pp. 230–234.

[7] C. Bepery, S. Baral, A. Khashkel, and F. Hossain, “Advanced home automation system using raspberry-pi and arduino,” *International journal of computer science and engineering*, vol. 8, no. 2, pp. 1–10, 2019.

[8] K. W. V. Kulkarni and P. Wani, “Iot based secured classroom automation system using nlp,” *International journal of computer science and engineering*, vol. 5, no. 2, pp. 1–6, 2019.

[9] T. Sali, C. Pardeshi, V. Malshette, A. Jadhav, and V. Thombare, "Classroom automation system," *International Journal of Innovations in Engineering and Technology*, vol. 8, no. 3, 2017.

[10] A. Gupta, P. Gupta, and J. Chhabra, "Iot based power efficient system design using automation for classrooms," in *2015 Third International Conference on Image Information Processing (ICIIP)*, 2015, pp. 285–289.

[11] E. Mulyana, R. Hakimi, and T. Hendrawan, "Bringing automation to the classroom: A chatops-based approach," in *2018 4th International Conference on Wireless and Telematics (ICWT)*, 2018, pp. 1–6.

[12] L. Marcomini and A. Cunha, "A comparison between background modelling methods for vehicle segmentation in highway traffic videos," *CoRR*, vol. abs/1810.02835, 2018. [Online]. Available: http://arxiv.org/abs/1810.02835

[13] G. Rathy and P. Sivasankar, "Iot based classroom automation using zigbee," *International Journal of Innovative Science, Engineering & Technology*, vol. 6, no. 3, pp. 98–102, 2019.

[14] P. Mrityunjaya, S. Muley, and D. Panda, "Intelligent classroom automation system using pic microcontroller," *International Journal of Research in Engineering and Technology*, vol. 05, no. 06, p. 154–160, 2016.

[15] C.-Y. Chen, Y.-P. Tsoul, S.-C. Liao, and C.-T. Lin, "Implementing the design of smart home and achieving energy conservation," in *2009 7th IEEE International Conference on Industrial Informatics*, 2009, pp. 273–276.

[16] C. Bepery, S. Baral, A. Khashkel, and F. Hossain, "Advanced home automation system using raspberry-pi and arduino," *International journal of computer science and engineering*, vol. 8, no. 2, pp. 1–10, 2019.

[17] A. N. Ansari, M. Sedky, N. Sharma, and A. Tyagi, "An internet of things approach for motion detection using raspberry pi," in *Proceedings of 2015 International Conference on Intelligent Computing and Internet of Things*, 2015, pp. 131–134.

[18] A. Maslekar, K. Aparna, K. Mamatha, and T. Shivakumara, "Smart lighting system using raspberry pi," *International Journal of Innovative Research in Science, Engineering and Technology*, vol. 04, no. 07, p. 5113–5121, 2015.

[19] K. Venkatesh, P. Rajkumar, S. Hemaswathi, and B. Rajalingam, "Iot based home automation usingraspberry pi," *Journal of Advanced Research in Dynamical Control Systems*, vol. 10, no. 07, pp. 1721–1728, 2018.

[20] M. Piccardi, "Background subtraction techniques: a review," in *2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No. 04CH37583)*, vol. 4, 2004, pp. 3099–3104.

[21] M. Shruthi and G. Indumathi, “Motion tracking using pixel subtraction method,” in *2017 International Conference on Computing Methodologies and Communication (ICCMC)*, 2017, pp. 550–552.

[22] C. Watman, D. Austin, N. Barnes, G. Overett, and S. Thompson, “Fast sum of absolute differences visual landmark detector,” in *2004 IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA’04. 2004*, vol. 5, 2004, pp. 4827–4832.

[23] S. S. Shazali, W. L. Cheong, S. Mohamaddan, A. A. Kamaruddin, A. Yassin, and K. Case, “Motion detection using periodic background estimation subtraction method,” in *2011 7th International Conference on Information Technology in Asia*, 2011, pp. 1–4.

[24] L. Ding and A. Goshtasby, “On the canny edge detector,” *Journal of Pattern Recognition*, vol. 34, no. 3, pp. 721–725, 2001.

[25] A. Manju and P. Valarmathie, “Video analytics for semantic substance extraction using opencv in python,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 3, pp. 4057–4066, 2021.

[26] W. F. Abaya, J. Basa, M. Sy, A. C. Abad, and E. P. Dadios, “Low cost smart security camera with night vision capability using raspberry pi and opencv,” in *2014 International conference on humanoid, nanotechnology, information technology, communication and control, environment and management (HNICEM)*, 2014, pp. 1–6.

[27] I. Benraya and N. Benblidia, “Comparison of background subtraction methods,” in *2018 International Conference on Applied Smart Systems (ICASS)*, 2018, pp. 1–5.

[28] I. Culjak, D. Abram, T. Pribanic, H. Dzapo, and M. Cifrek, “A brief introduction to opencv,” in *2012 proceedings of the 35th international convention MIPRO*, 2012, pp. 1725–1730.

[29] B. Vaidya, A. Patel, A. Panchal, R. Mehta, K. Mehta, and P. Vaghasiya, “Smart home automation with a unique door monitoring system for old age people using python, opencv, android and raspberry pi,” in *2017 International Conference on Intelligent Computing and Control Systems (ICICCS)*, 2017, pp. 82–86.

[30] M. Richardson and S. Wallace, *Getting started with raspberry PI.* O’Reilly Media, Inc., 2012.

[31] G. Van Rossum and F. L. Drake Jr, *Python tutorial.* Centrum voor Wiskunde en Informatica Amsterdam, 2020, vol. 620.

Appendix A

Appendix A contains all the figures of hardware setup.

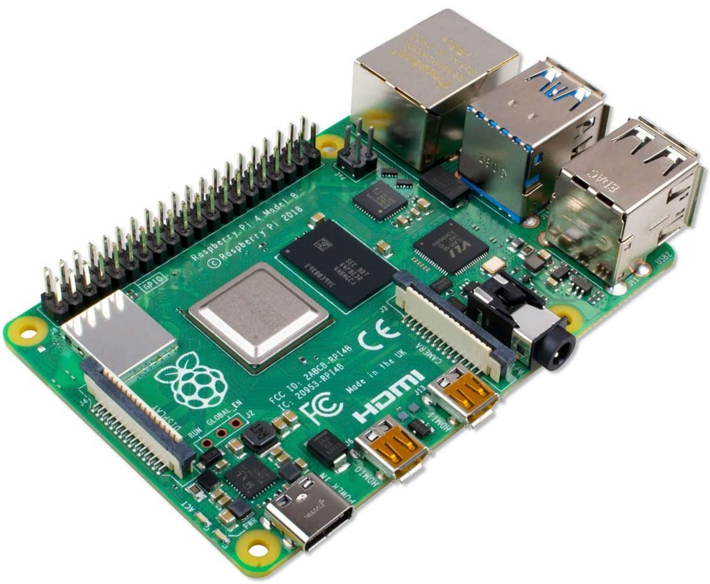

Figure A.1: Raspberry Pi Model

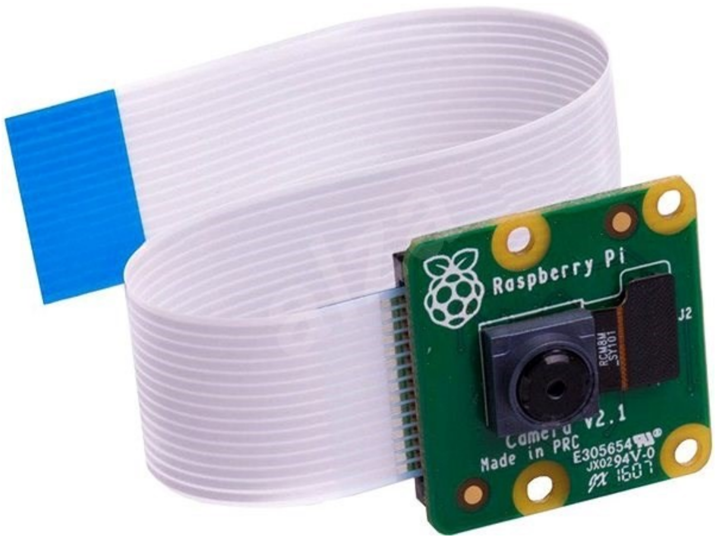

Figure A.2: Pi Cam 2.1

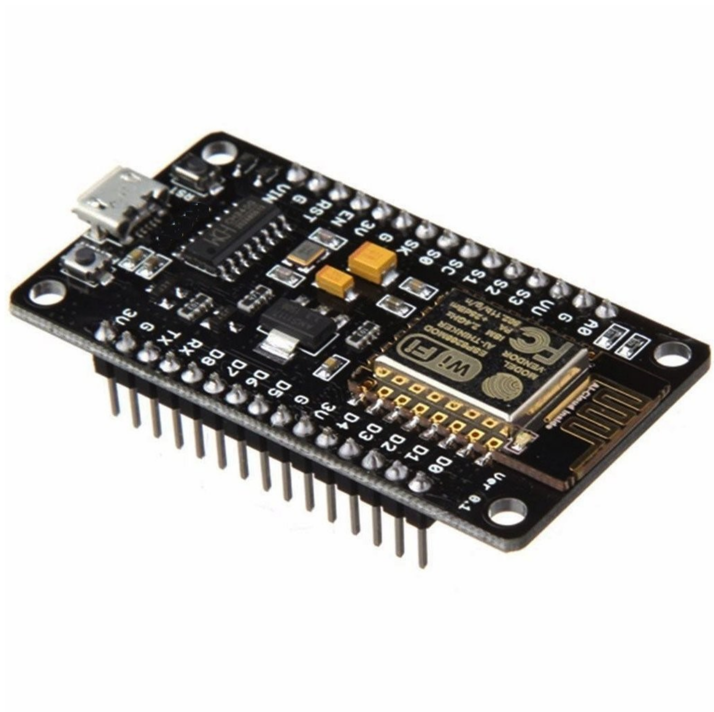

Figure A.3: Esp8266 Node MCU

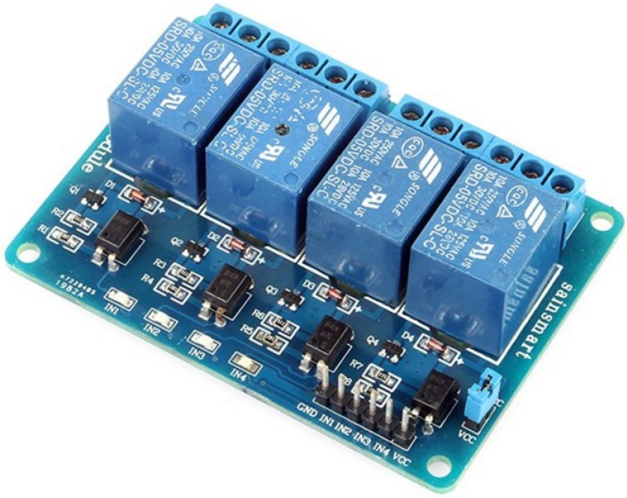

Figure A.4: 4-channel Relay Module

Figure A.5: Wifi router

Figure A.6: USB power bank

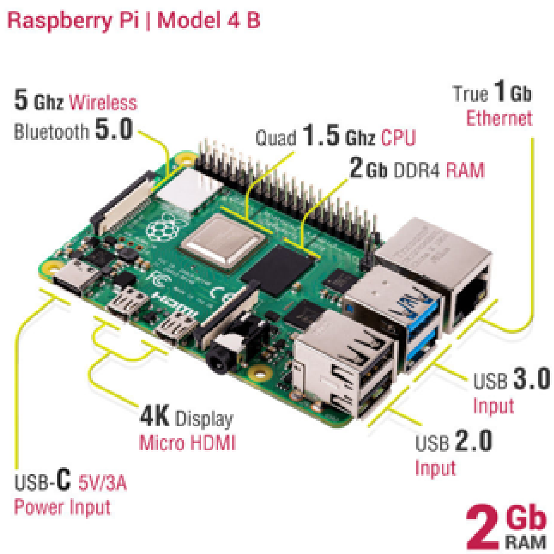

Figure A.7: Raspberry Pi

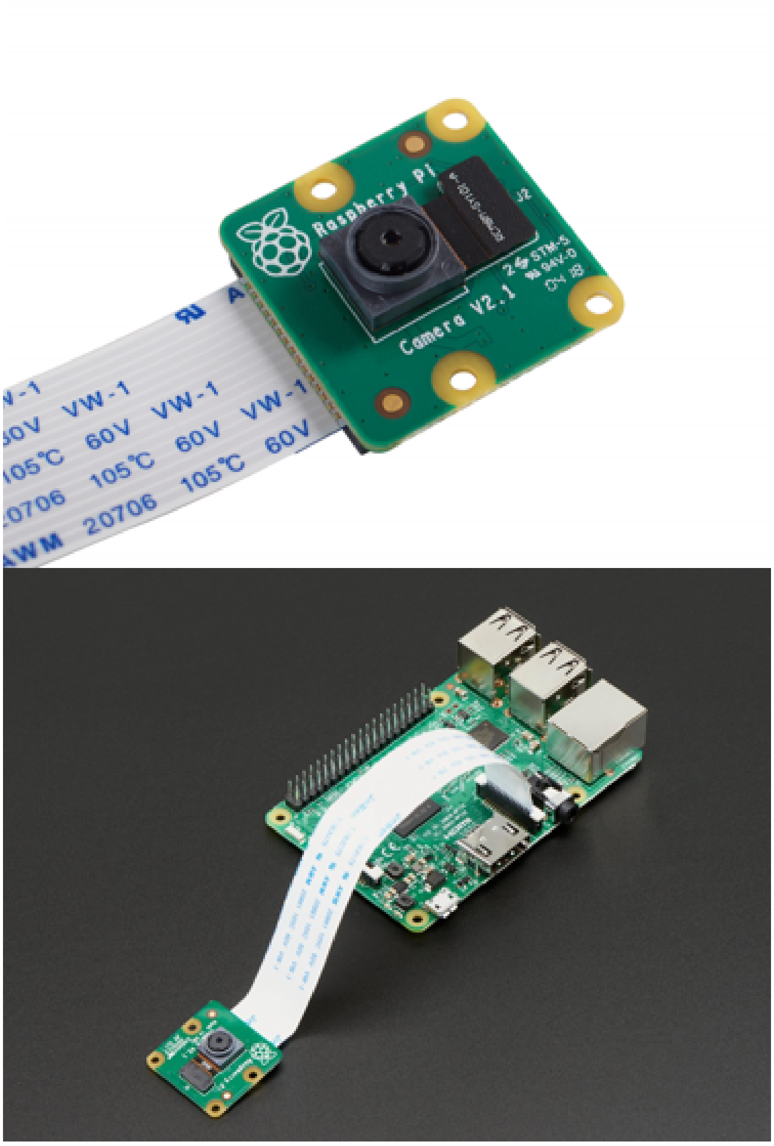

Figure A.8: Pi Camera