**SMART VIDEO SURVEILLANCE USING**

**AI AND IOT**

**A project report submitted for the partial fulfillment of the Bachelor of Technology Degree in Information Technology under Maulana Abul Kalam Azad**

**University of Technology**

BY

### Sudeshna Chaudhuri

ROLL NO: 10400214113, REGISTRATION NO: 141040110319

**&**

### Swapnil Roy

ROLL NO: 10400214122, REGISTRATION NO: 141040110329

Under the Guidance of:

### Prof. Tapan Kumar Hazra

Department of Information Technology For the Academic Year 2017-2018



Institute of Engineering & Management Y-12, Salt Lake, Sector-V, Kolkata-700091

Affiliated To:



Maulana Abul Kalam Azad University of Technology BF-142, Salt Lake, Sector I, Kolkata-700064

**PREFACE**

The thesis has been submitted towards the partial fulfillment of the requirement for the **AWARD** of the **DEGREE of BACHELOR OF TECHNOLOGY IN INFORMATION TECHNOLOGY** of **INSTITUTE OF ENGINEERING & MANAGEMENT, MAULANA ABUL KALAM AZAD UNIVERSITY OF TECHNOLOGY, KOLKATA.**

### Dated:May, 2018.

**Place: Salt Lake, Kolkata**

**. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .**

**Sudeshna Chaudhuri Swapnil Roy**

**ACKNOWLEDGEMENT**

We should like to take this opportunity to extend our gratitude to the following revered persons without whose immense support, completion of this project wouldn’t have been possible.

We are sincerely grateful to our advisor and mentor **Prof. Tapan Kumar Hazra** of **Information Technology**, IEM Kolkata, for his constant support, significant insights and for generating in us a profound interest for this subject that kept us motivated during the entire duration of this project.

We would also like to express our sincere gratitude to **Prof. Dr. Satyajit Chakrabarti** (Director**,**IEM)**, Prof. Dr. Amlan Kusum Nayak (**Principal,IEM) and **Prof. Dr**. **Mohuya Chakraborty**, HOD of **Information Technology** and other faculties of Institute of Engineering & Management, for their assistance and encouragement.

Last but not the least, we would like to extend our warm regards to our families and peers who have kept supporting us and always had faith in our work.

### Sudeshna Chaudhuri Swapnil Roy

Reg.No: 141040110319 Reg.No: 141040110329

Dept. of Information Technology Dept. of Information Technology

Institute of Engineering & Management, Kolkata Institute of Engineering & Management, Kolkata

# ABSTRACT

The proposed work is aimed at providing security to a location when someone is far away from there. There have been some works on this which target to provide solution to the same problem but the main criterion which sets apart lies in its effectiveness. Hence this model has been made to focus on the parts where the consumption of power and accuracy of detection could be enhanced. The basic functionality is to send alert mail or message to the user in his registered mail id upon detecting an intrusion. Certain approaches have been adopted to make the system more feasible and intelligent enough to make smart moves. An interesting feature of this system is that it gets smarter day by day and thus the efficiency keeps on increasing. By “smarter” we mean that the system starts to learn whether the person who is trying to open the locker or door is actually allowed to do so or not. The involvement of Artificial Intelligence plays an important role in this part. The use of PIR motion sensor has made it possible to lower down the power consumption to a great extent. The camera starts to capture only when there is a movement sensed by the PIR.

An image of the person will be sent to the user via email. If the user prompts that the person is not harmful then no further action will be taken. But if he feels that the person might have some bad intention then the user can command to turn on the burglary alarm remotely.

**TABLE OF CONTENTS**

**TOPIC PAGE NUMBER**

**CHAPTER 1. INTRODUCTION**

* 1. [Motivation 1](#_TOC_250007)
  2. [Objective 1](#_TOC_250006)

CHAPTER 2. BACKGROUND

* 1. [Literature Survey 3](#_TOC_250005)
  2. [Technologies Used 3](#_TOC_250004)
  3. [Components Used 4](#_TOC_250003)

CHAPTER 3. PROPOSED STRATEGY

* 1. [Project Architecture 7](#_TOC_250002)
  2. [Working Principle 8](#_TOC_250001)

CHAPTER 4. EXPERIMENTAL SETUP AND PRACTICAL IMPLEMENTATION

[4.1. Circuit Diagram 12](#_TOC_250000)

* 1. Implementation and Results 12

CHAPTER 5. APPLICATIONS 20

CHAPTER 6.CONCLUSION AND FUTURE SCOPE 21

CHAPTER 7. REFERENCES 22

# INTRODUCTION

## Motivation

There have been many proposed models targeting to provide security to homes and other commercial places. Each of them has taken different approach to achieve their own goal.

Usually in such home security system, presence of intruder is felt using some motion sensor. Human beings and animals emit infrared radiation. These sensors detect motion by sensing such radiation. Security must be ensured when an “unwanted” person tries to break into our house. But here, using just a motion sensor won’t help in distinguishing between the presence of an unwanted person and a family member (who is not unwanted). In such scenario, the system would raise false alarms.

Some systems keep on capturing video throughout the whole day. This is a mere wastage of power as the system is needed to stay alert only when someone actually passes by.

There exist systems which allow the member of family to have access to some place or to something. But if a new member like a relative comes to our place, it would be very embarrassing if the system starts to ring the burglary.

## Objective

Most of the people tend to overlook the benefits of installing surveillance device when they are far away from their location. This is because some of them think that it would be too costly and the others think that they don’t need it. Such thoughts are very natural to come when they have never been broken into in past. But the reality is that we never know when we become the next victim.

Our project can help prevent robbery in homes as well as in places like banks.

We hereby present a very cost effective system which will keep a constant watch over a place and will send notifications “smartly”.

The proposed work exploits the concept of Artificial Intelligence (AI) and Internet of Things (IOT) which together helps in achieving our goal.

The system ensures safety and security to a location by recognizing unknown and unwanted people by itself and decides whether to inform the owner about this or not. Also

the owner can operate burglary alarm far away from the location to alert the neighbours if needed.

In brief, AI is that area of computer science which enables a machine to acquire intelligence just like a human who gradually learns from his experience. IOT is a technology in modern computer science which enables “things” to communicate with each other and exchange useful information among them. It uses networking, data analytics, sensing etc. to serve its purpose.

# BACKGROUND

## Literature Survey

1. Door security nodes, infrared security nodes etc. are used to detect an intrusion [1]. Sometimes it may happen that the member of our family enters into the area where sensors are placed. This will raise false alarm in such cases.
2. The face detected in front of the door is captured using a camera and is sent to the user via mail or tweeter [2]. But the camera is kept on for all the time which wastes lot of charge.
3. Only PIR motion sensors have been used in [3] for the security part but the motion sensor detects motion of any object that emit infrared wave. Animals to some extent also emit infrared waves. In case of movement by dogs or cats the sensors won’t be able to distinguish it from that of humans.
4. In [4] the major drawback is that the security of zigbee is not as tight as wifi, also replacement cost will be high in case of issues in zigbee.
5. [5] illustrates that GSM is communication method between remote user and home network server but doesn’t apply it to home security system.

## Technologies Used

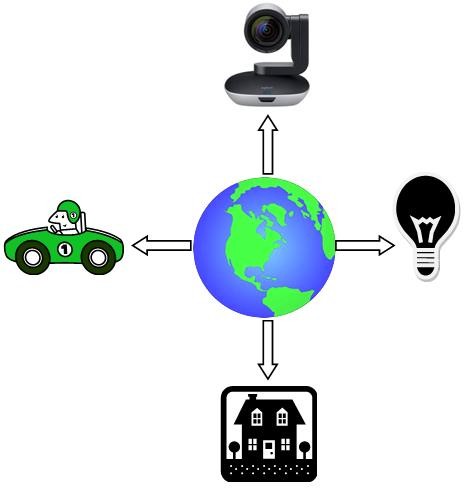
### Artificial Intelligence

A baby slowly starts to recognize his parents from their appearance and also from their tonal modulation. This helps him to correctly identify them. Obviously this process of learning does take some time. Similarly in case of AI, the machine slowly learns from all the data that it has been fed and eventually gets to distinguish between the correct and the incorrect. Depending upon the extent to which it is sure about the answer it gives the result which ranges from 0 to 1. That is, if its 100% sure then it returns 1, if its not sure at all then it returns 0,if its half confident about its assessment then returns 0.5 and so on.

### Internet of Things

To understand the importance of IOT let’s take an example. Suppose an accident has occurred on a road and as a consequence all the vehicles behind it have stuck. Now this blockage causes a huge problem mainly during the peak hours. With the help of IOT, every object in a network is connected to each other and is capable of exchanging

information. Hence when an accident occur causing jam in the entire route, this information spreads across all the connected devices and lets them take decision on which route to choose well in advance. Fig. 1 demonstrates overview of IOT.



## Components Used

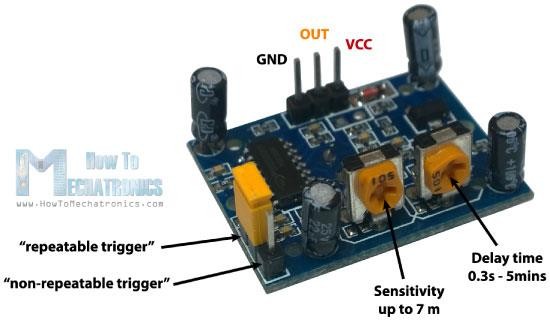
* Raspberry Pi 3B
* 5 MP Raspberry Pi 3B compatible camera module
* PIR motion sensor
* Bread board
* Jumper wires
* Cloud based API for Visual Recognition

### Raspberry pi 3B

The Raspberry Pi 3 is the third-generation Raspberry Pi.

* 1GB RAM
* Quad Core 1.2GHz Broadcom BCM2837 64bit CPU
* BCM43438 wireless LAN and Bluetooth Low Energy (BLE) on board
* 40-pin extended GPIO
* 4 USB 2 ports
* 4 Pole stereo output and composite video port
* Full size HDMI
* CSI camera port for connecting a Raspberry Pi camera
* Micro SD port for loading your operating system and storing data
* Upgraded switched Micro USB power source up to 2.5A

### PIR Motion Sensor



PIR motion sensor - "Passive Infrared", "Pyroelectric", or "IR motion" sensors. It is used to detect any movement made by infrared emitting source.

PIR sensor responds to the infrared wave radiated by entities whose radiation is strongest at wavelength between 9.4 μm and 10.4 μm. The sensor in the PIR detects or “reads” infrared radiation “emitted” from objects.

Each object with a temperature above [absolute zero](http://en.wikipedia.org/wiki/Absolute_zero) ( -273.15° Celsius, -459.67° Fahrenheit, or 0 Kelvin) will radiate infrared, even us humans, and even though we mere humans cannot see this.

There are two knobs as shown in the diagram, one is for controlling the sensitivity and other is for the delay.

The sensitivity knob when rotated clockwise increases the sensitivity of the sensor. The delay knob also functions in the similar way.

### Raspberry pi 5MP Camera Module

The Raspberry Pi camera module is capable of taking photo and video and can be controlled programmatically.

The flex cable inserts into the connector situated between the Ethernet and HDMI ports, with the silver connectors facing the HDMI port. The flex cable connector should be opened by pulling the tabs on the top of the connector upwards then towards the Ethernet port. The flex cable should be inserted firmly into the connector, with care taken not to bend the flex at too acute an angle. The top part of the connector should then be pushed towards the HDMI connector and down, while the flex cable is held in place.

### Enabling the camera

Open the raspi-config tool from the Terminal:

sudo raspi-config

Select Enable camera and hit Enter, then go to Finish and you'll be prompted to reboot.

### Piezoelectric Speaker



A piezoelectric speaker (sometimes colloquially called a "pizo") or buzzer is a [loudspeaker](https://en.wikipedia.org/wiki/Loudspeaker) that uses the [piezoelectric effect](https://en.wikipedia.org/wiki/Piezoelectric_effect) for generating [sound](https://en.wikipedia.org/wiki/Sound). The initial mechanical motion is created by applying a voltage to a piezoelectric material, and this motion is typically converted into audible sound using diaphragms and resonators. Compared to other speaker designs piezoelectric speakers are relatively easy to drive; for example they can be connected directly to [TTL](https://en.wikipedia.org/wiki/Transistor%E2%80%93transistor_logic) outputs, although more complex drivers can give greater sound intensity. Typically they operate well in the range of 1-5kHz and up to 100kHz in ultrasound applications.

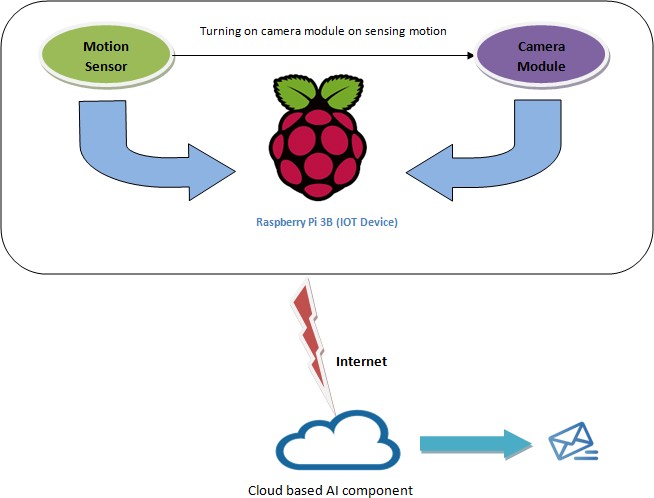
# PROPOSED STRATEGY

## Project Architecture

A raspberry pi 5 mp camera module and PIR motion sensor are needed. The sensor detects movement of any object that emits infrared heat situated within a range of 6-7 meters.

When such movement is observed, only then the camera gets activated and captures photo and sends it to the Raspberry pi 3B.

The image is then sent to IBM Watson pre-trained Visual Recognition API. This API has been trained to recognize the emptiness of the room. It sends back its confidence level (later referred to as “score”) to the Pi. This Pi in turn sends messages to the user based on its confidence level.



## Working Principle

After sensing motion when the camera is turned on, the purpose of this project is to implement Artificial Intelligence so that e-mail sending or notification process is efficient and some sort of intelligence is involved in sending those notifications. We have used Visual Recognition(API) tool of IBM Watson, which is a cloud based computing system. We have used the free version of Watson which is known as ‘Watson Lite’. Watson is a computing system for [question answering](https://en.wikipedia.org/wiki/Question_answering) (QA) related to advanced [natural language](https://en.wikipedia.org/wiki/Natural_language_processing) [processing](https://en.wikipedia.org/wiki/Natural_language_processing), [information retrieval](https://en.wikipedia.org/wiki/Information_retrieval), [knowledge representation](https://en.wikipedia.org/wiki/Knowledge_representation), [automated reasoning](https://en.wikipedia.org/wiki/Automated_reasoning),

and [machine learning](https://en.wikipedia.org/wiki/Machine_learning) technologies to the field of [open domain question answering](https://en.wikipedia.org/wiki/Open_domain_question_answering). For using Visual Recognition we had to create a service of the tool after logging on to our IBM Bluemix account.

Artificial Intelligence is a domain of computer science where computers are fed with information using which the computing system give response upon posing some questions. To call and use the corresponding API we have used cURL, which is a computer software providing a library and command-line tool for transferring data using various protocols. To use the Visual Recognition API we have to create classifiers. A classifier is a set of instruction that predicts some output on giving in some input. Here we have to provide information to the classifier in form of images. These images were gathered into zip file and then uploaded in the corresponding classifiers. According to the requirements mentioned by IBM Watson for the use of this API, we had to create two positive classifiers (or one positive and another negative classifier). So, one positive and one negative classifier were created by using cURL command.

This will give an output as follows:

{

"classifier\_id": "TKH\_849252517", "name": "TKH",

### "status": "training",

"owner": "c866864b-5a68-4d41-9858-85ee2356b46d", "created": "2018-04-18T12:54:15.698Z",

"updated": "2018-04-18T12:54:15.698Z",

"classes": [

{

"class": "Sir"

}

],

"core\_ml\_enabled": true

}

After creating classifier we have to check for the readiness of the classifiers. Once the classifier is ready, we will get the following output on giving corresponding cURL command. The status will change from “training” to “ready”.

{

"classifier\_id": "TKH\_849252517", "name": "TKH",

### "status": "ready",

"owner": "c866864b-5a68-4d41-9858-85ee2356b46d", "created": "2018-04-18T12:54:15.698Z",

"updated": "2018-04-18T12:54:15.698Z",

"classes": [

{

"class": "Sir"

}

],

"core\_ml\_enabled": true

}

After creating classifier and checking for its readiness, we fed images and tracked the corresponding output. The positive classifier was trained with known people and negative

classifier was trained with images of some outsider. Then we checked the result on giving a known person’s images, whose other images were fed to the API and was trained. We got the following output with the “score” that was positive enough (>.75) to understand that the classifier was working fine.

{

"images": [

{

"classifiers": [

{

"classifier\_id": "TKH\_849252517", "name": "TKH",

"classes": [

{

"class": "Sir",

### "score": 0.87

}

]

}

],

"image": "TkhSirOut.jpg"

}

],

"images\_processed": 1,

"custom\_classes": 1

}

If the captured image is either of unknown or from negative class, the corresponding output is generated:

{

"images": [

{

"classifiers": [

{

"classifier\_id": "TKH\_849252517", "name": "TKH",

### "classes": []

}

],

"image": "ne.jpg"

}

],

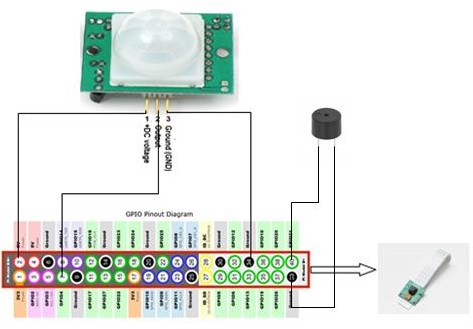
"images\_processed": 1,

"custom\_classes": 1

}

# EXPERIMENTAL SETUP AND PRACTICAL IMPLEMENTATION

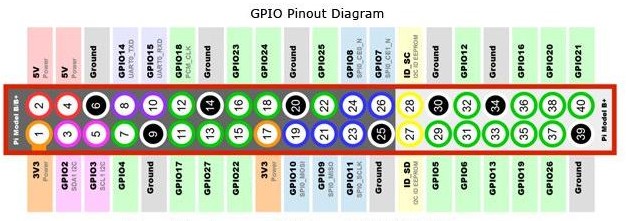
## CIRCUIT DIAGRAM



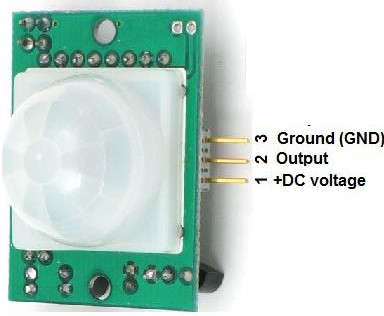
* 1. **Implementation**

We have used two python files. The first one was for detecting the motion and capturing the photo of person on detecting his motion. We have named it FYP\_part1. After capturing, it sends the image to the IBM Watson VR custom classifier to classify it with the already fed photos of a person or of some specific people. After the classifier checks whether the image matches with the pre existing ones or not and returns the result. For carrying out all these functions we need some packages namely RPi, os, sys and time.

The pinout diagram of Raspberry pi 3B is shown below:-



GPIO stands for General Purpose Input Output. Any of the GPIO pins can be designated (in software) as an input or output pin and used for a wide range of purposes. For our project we have used GPIO4 as the input pin. The input is received from the output terminal of the PIR motion sensor. Before going into any further discussion with this, let us know a bit about PIR motion sensor.



Here the VCC is connected to the 5V pin of Pi and GND to GND of Pi. The OUT terminal indicates the output of the motion sensor. If no movement is detected then OUT gives 0V

otherwise produces some voltage up to 3.3v. This OUT is hence connected to GPIO4 of Pi so that it receives any signal whenever some motion is sensed.

GPIO.input(pinNumber) is a function present in the RPi package which returns 0 when pinNumber( which is GPIO4 here) is at 0v i.e. no motion and returns 1 when the pinNumber is at HIGH level. We store this return value in a variable “currentState” which helps in further decision making.

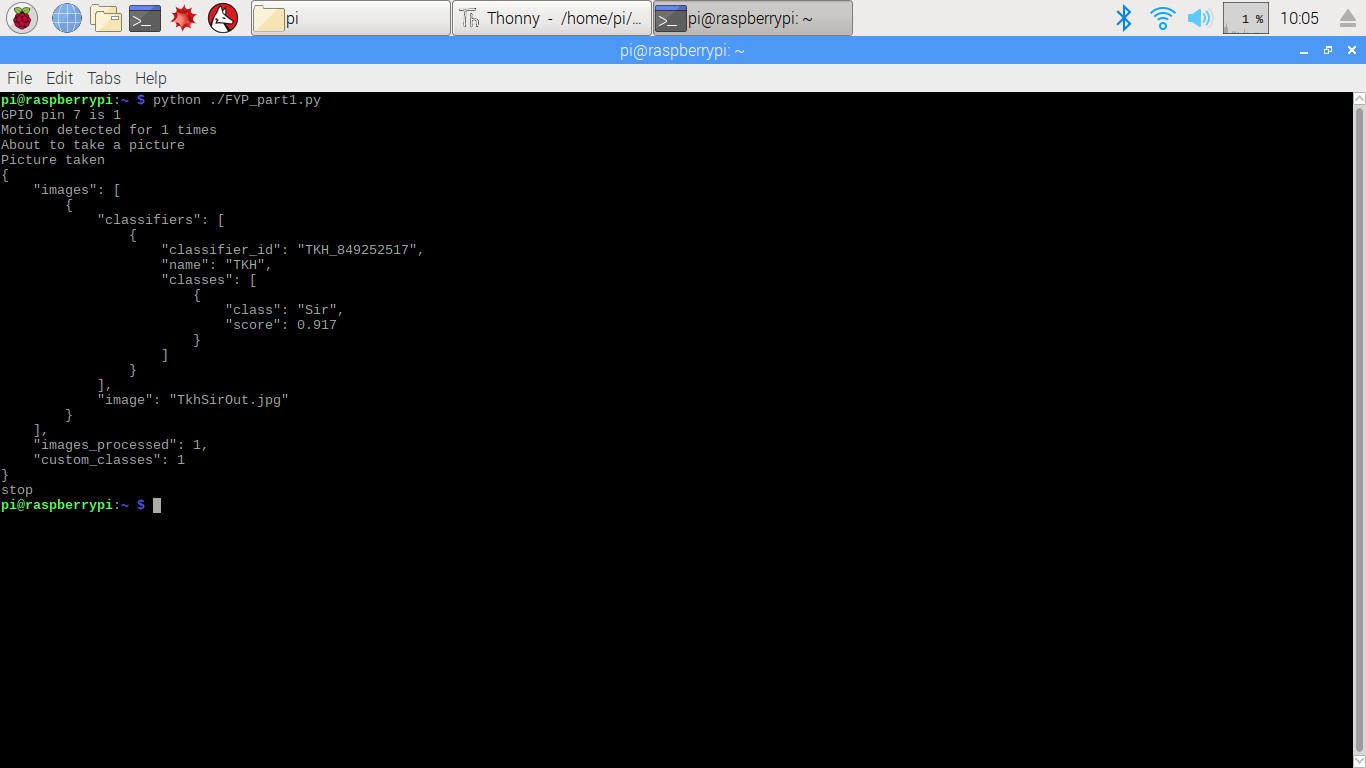
sys.exit([*arg*]) is used to exit from Python. This is implemented by raising the [SystemExit](https://docs.python.org/2/library/exceptions.html#exceptions.SystemExit) exception, so cleanup actions specified by finally clauses of [try](https://docs.python.org/2/reference/compound_stmts.html#try) statements are honored, and it is possible to intercept the exit attempt at an outer level.

time.sleep(*secs*) suspends execution of the current thread for the given number of seconds. The argument may be a floating point number to indicate a more precise sleep time. The actual suspension time may be less than that requested because any caught signal will terminate the [sleep()](https://docs.python.org/2/library/time.html#time.sleep) following execution of that signal’s catching routine. Also, the suspension time may be longer than requested by an arbitrary amount because of the scheduling of other activity in the system.

When “currentState” is ‘1’ then the camera is turned on and the picture is taken. This picture is saved in the working directory and sent for further examination to the Watson VR classifier.

After that the classifier sends back the result.

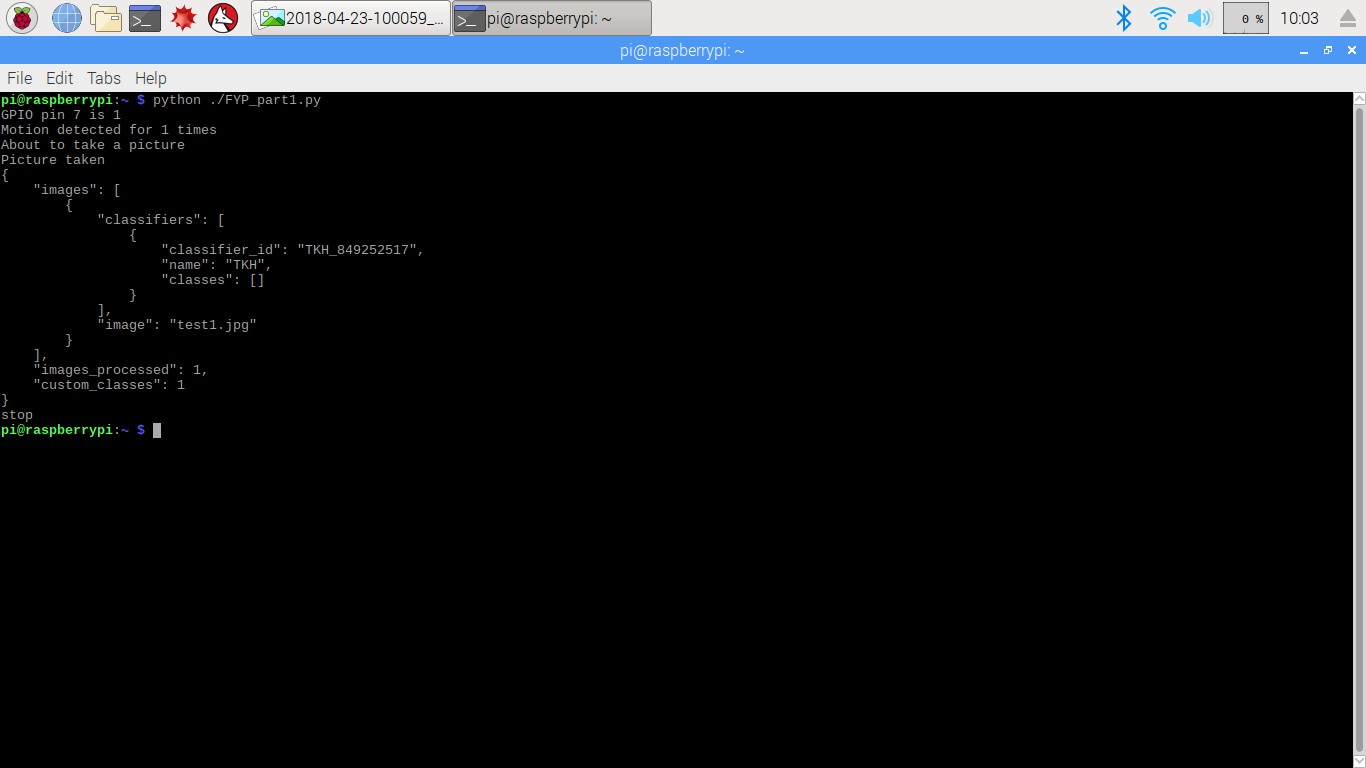
If the photo captured is of known person then the output will look like:



Here the photo has matched with the existing ones and the classifier has returned “score”

= 0.917 which means that it is 91.7% sure about the result returned. Here “class”: “Sir” means that the photo sent to the classifier named “TKH” has been checked with photos existing in “Sir” class depending on which the result is procured.

If the photo captured is of some unknown person then the output will be:



Here since the unknown face has not matched with any of the pre existing photos hence no “score” is returned. Also the unknown face matched with no classes so “classes”: [] i.e. empty class.

The other python file is called the FYP\_part2. It calls FYP\_part1 and stores its output in a text file called “output.txt”. As stated above the output will contain the result of whether the photo has matched or not. It also says to what extent the photo has matched by using an entity named “score”.

If the “score” is less than 0.75 or there is nothing present as “score” which means the classifier is unable to recognize the face or is not sure enough about it then the photo will be mailed to the recipient.

Otherwise, if the “score” is more than 0.75 then no notification will be sent. Now AI functions just like a human whose knowledge gets widened when fed with more information. When a baby sees a person for the first time, it won’t be possible for him to retain the image and recognize the person the next time he comes. But if the baby sees him for multiple times then slowly his brain gets trained and now he can identify the person well. Same is the case with AI. It would show great results when more and more training is given to it.

Here training means feeding the classifier with more photos. So if we get “score” >0.75 then we would like to feed the classifier with that photo. Here the photos are stored in zip file. So we add it to the zip file and train the classifier.

In this file we need to import many packages namely smtplib, MIMEImage from email.mime.image, MIMEMultipart from email.mime.multipart, subprocess and zipfile.

The [**smtplib**](https://docs.python.org/2/library/smtplib.html#module-smtplib) module defines an SMTP client session object that can be used to send mail to any Internet machine with an SMTP or ESMTP listener daemon. For details of SMTP and ESMTP operation.

Simple Mail Transfer Protocol (SMTP) is a protocol, which handles sending e-mail and routing e-mail between mail servers.

Python provides **smtplib** module, which defines an SMTP client session object that can be used to send mail to any Internet machine with an SMTP or ESMTP listener daemon.

import smtplib

smtpObj = smtplib.SMTP( [host [, port [, local\_hostname]]] )

host- This is the host running your SMTP server. You can specify IP address of the host or a domain name like tutorialspoint.com. This is optional argument.

port- If you are providing *host* argument, then you need to specify a port, where SMTP server is listening.

Local\_hostname- If your SMTP server is running on your local machine, then you can specify just localhost as of this option.

We have called the GMAIL server which is 'smtp.gmail.com' . Port number 587 is used to communicate with a mail client and a server.

Port number 25 is used primarily for SMTP relaying. SMTP relaying is the transmittal of email from email server to email server.

This is the default mail submission port. When a mail client or server is submitting an email to be routed by a proper mail server, it should always use this port.

An SMTP object has an instance method called sendmail, which is typically used to do the work of mailing a message. It takes three parameters −

* + - The sender − A string with the address of the sender.
    - The receivers − A list of strings, one for each recipient.
    - The message − A message as a string formatted as specified in the various RFCs.

We have created a Gmail account that sends the captured negative image by mail to another Gmail account. To do this, we had to use login() function with parameters e-mail address and its corresponding password.

attachment ='/home/pi/test1.jpg'

Then open the image file using open() function. fp= open(attachment, 'rb')

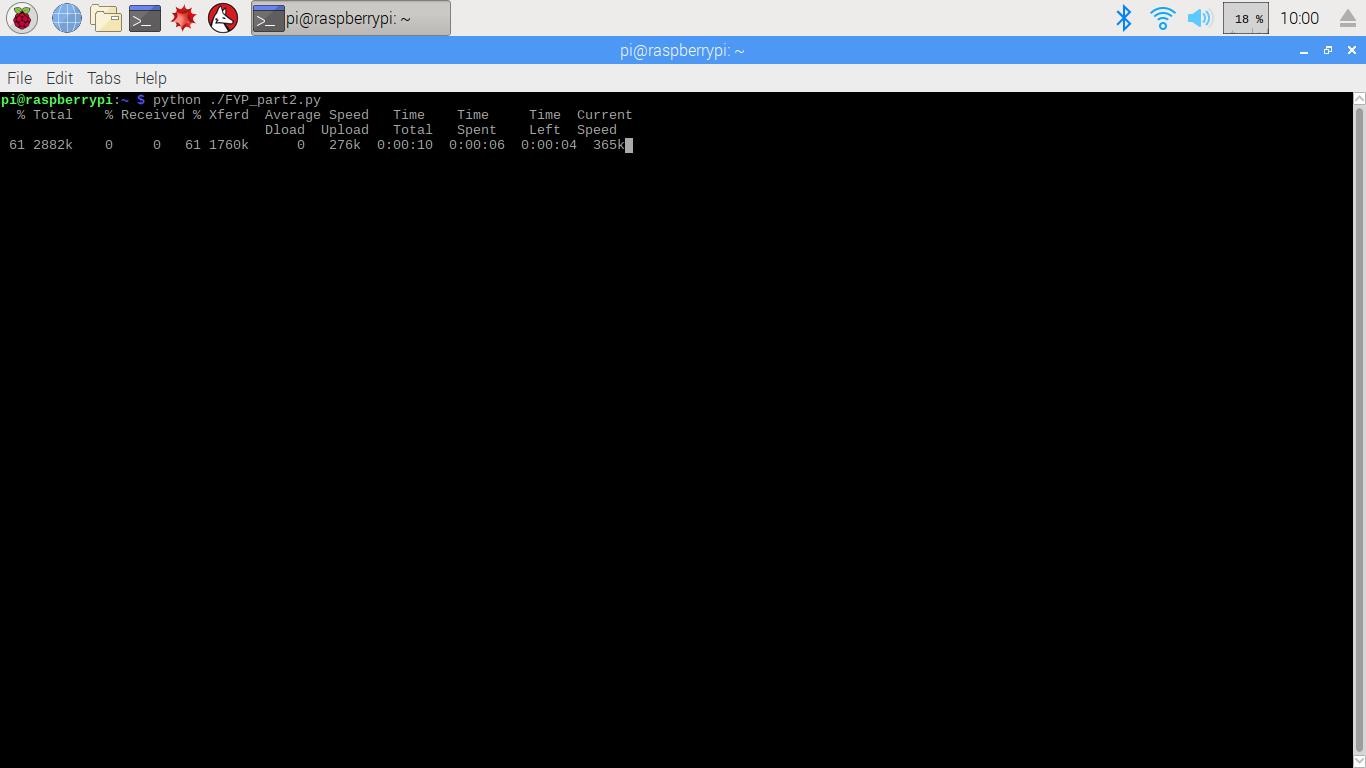
‘rb’ is for reading in binary mode.

Then the file was read using read() function and then saved in a variable called img. img= MIMEImage(fp.read())

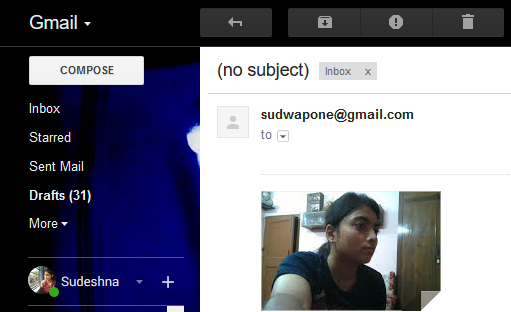
Then an instance of MIMEMultipart, namely “msg” was created.

Using, mail= smtplib.SMTP('smtp.gmail.com',587) connection with GMAIL server was set.

Start TLS for security mail.starttls()



Now an e-mail containing the photo of the intruder as an attachment is received by the owner (as the face detected did not get a score value implying that it is not a known face).



After this if the owner prompts to ring the burglary alarm then the instruction is followed. Otherwise if the owner ignores it then no action will take place. We have discussed more about it below:

After getting the mail with attached image of the intruder, the owner can take a call whether the intruder is known or unknown. If after seeing the attached image, the owner feels that the person is not having any wrong intentions or he/she is a known person then he can reply a text mail to the Raspberry pi. Here, the text message was ‘Ignore’. Once the pi gets an e-mail with its body containing text ‘Ignore’ it means the owner knows the person and thus the person’s image will be programmatically included in the positive images’ zip file. For this purpose, zip module was imported.

**with ZipFile('/home/pi/tkhSir photos.zip', 'a') as myzip:**

this command was used to append the image to the existing zip file of positive images.

If the user feels that the person might have some bad intentions then he can mail back “Ring”. In the same way as mentioned above if the python file finds “Ring” in the body of mail, then it rings the buzzer.

If the pi does not find any unseen mail even after sending the photo of the intruder, it means that the user might not have seen the mail. It is not okay to keep silent in such case. Since it is unsure of whether the person is known or not, to be on the safe side it rings the buzzer after waiting for few seconds.

How does the pi get to know about the response of the user is by scanning the inbox for any unseen messages sent by the registered user id. This is the last part of the 2nd python file.

# APPLICATIONS

Let us take some instances where our project finds its application.

Suppose X has a locker in which he keeps some of his valuables. He stays with his family members who are allowed to access the locker. But they usually stay outside the house during the day and the house is kept vacant for a long time. He wants to keep an eye over the locker and get informed only when some unknown person tries to use it.

How will our system work efficiently in such cases?

Most of the devices available in the market try to sense any kind of intrusion, be it from a known or an unknown person. But there is no need for X to get notified if his family member uses the locker. Here our system works well because it does not send any mail for such cases and only sends whenever it fails to recognize the person. This ability of recognition is enabled with the help of Artificial Intelligence.

Let’s take another scenario in which our system can be used.

All banks have cash rooms in which they store the cash money and only some authorized people are allowed to enter into it. But there have been many cases of bank robbery where the dacoits break open into the cash room by threatening the bank employees at gunpoint to give away the keys or unlock the room. During such cases even the security guards can’t do anything and in some cases the guards are also involved in the robbery as well.

Our project can prevent such incidents from occurring.

We install the system inside the cash room. If the system detects known faces in the room then it’s perfectly alright. Otherwise if some unauthorized person tries to intrude then immediately an alert message will be sent to the nearby police station notifying them about the incident.

# CONCLUSION AND FUTURE SCOPE

The system consumes very less power as the camera is not kept on throughout the period and captures only on sensing motion. We intend to take our project to the next level by adding some features.

On receiving the photo via mail, the user would get two options in form of buttons – “Ring” or “Ignore”. If he presses “Ring” button then a burglary alarm fitted near the system will ring to alert the neighbours about the intrusion and indicating that help is needed.

If he presses on “Ignore” that means the person standing in front of the camera is safe and no action needs to be taken. If the same action (i.e. “Ignore”) is repeated for the same person for few times then the AI would slowly start to understand that the person is reliable and that it should not bother the user with mail if that very person comes the next time.

Further to make the system work in dark, we would like to use night vision camera with greater clarity.

# REFERENCES

1. Zhao, Y., & Ye, Z. (2008). A Low Cost GSM/GPRS Based Wireless. *IEEE Transactions on Consumer Electronics* , 567-572.
2. Chowdhury, M. N., Nooman, M. S., & Sarker, S. (2013). Access Control of Door and Home Security by Raspberry Pi Through Internet. *International Journal of Scientific & Engineering Research* , 550-558.
3. Kishore, K. R., Jain, V., Bose, S., & Boppana, L. (2016). IoT Based Smart Security and Home Automation. *IEEE* , 1286-1289.
4. Hou, J., Wu, C., Yuan, Z., Tan, J., Wang, Q., & Zhou, Y. (2008). Research of Intelligent Home Security Surveillance System Based on Zigbee. *IEEE computer society* , 554-557.
5. B. Yusekkaya, A. A. Kayalar, M. B. Tosun, M. K. Ozcan, and A. Z. Alkar, “A GSM, internet and speech controlled wireless interactive home automation system” IEEE Trans Consumer Electron.,vol. 52, no. 3, pp. 837-843, Aug. 2006.
6. Lee, S., Ha, K. N., & Lee, K. C. (2005). A pyroelectric infrared sensor-based indoor location-aware system for the smart home. *IEEE* , 1311-1317.
7. [https://www.ibm.com/watson/developercloud/visualrecognition/api](https://www.ibm.com/watson/developercloud/visualrecognition/api%20%20/v3/curl.html?curl&amp;introduction)

[/v3/curl.html?curl#introduction](https://www.ibm.com/watson/developercloud/visualrecognition/api%20%20/v3/curl.html?curl&amp;introduction)

1. <https://programminghistorian.org/lessons/working-with-text-files>
2. <https://www.raspberrypi.org/>
3. <https://pymotw.com/2/zipfile/>
4. [https://medium.com/@petehouston/capture-images-from-raspberry-pi-camera- module-using-picamera-505e9788d609](https://medium.com/%40petehouston/capture-images-from-raspberry-pi-camera-module-using-picamera-505e9788d609)
5. <https://learn.adafruit.com/pir-passive-infrared-proximity-motion-sensor?view=all>