# IoT and Face Recognition based Remote Access System

Amey Bairolu, Sanjukta Sheth, Binay Hazra, Arpit Rawankar

*Abstract*— **There are many security systems available in the market and in thesis, which are stringent and can make things worse in the hour of need as there are many situations which demand customized authentication. These systems either require the visitors to provide details of authenticity, for instance, ID cards, passwords, etc. or require someone to explicitly be near the entrance to manually check the details. There are even some existing security systems which become cumbersome for the owners by seeking permission time and again even when the visitor is a frequent one. This paper, therefore is focused on providing a smart solution, without compromising security, by employing a system which will not only grant access to authorized visitors, but will also allow some expected visitors to walk through the doors and restrict the rest even when there is no one physically present near the entrance or anywhere near the concerned area to manually check on the visitors. This system will therefore solicit permission from the owner only when the visitor is unknown.**

**This remote access system therefore uses face recognition, which is a biometric artificial intelligence based technology, and Internet of Things (IoT) to give access to the known visitors when it recognizes them successfully or to give access to the expected visitors by sending an e-mail to the owner with a picture of the visitor, to let the owner decide whether or not to give access to the visitor even when the owner isn't anywhere near the entrance. For face recognition, histogram of oriented gradient has been used as it requires lower computational time and lower processing power. To allow the owner to remotely send controlling commands to the raspberry pi, Flask has been used to host a webpage designed solely for the stated purpose, along with serveo that allows remote port forwarding. The system also uses an IoT API (ThingSpeak) to upload the name of the authorized visitors and their time of arrival.**

***Index Terms***—**E-mail, Face Recognition, Remote control, Internet of things (IoT).**

## I. Introduction

THE world has become a lot busy lately and one barely finds time for himself. Everybody is in a race to keep up with the world's pace. It has become difficult for any human to separate professional life from personal life.

But there are some problems which are just unavoidable or come at a time when they are least expected. One may suddenly have to leave for some work and after some time he gets the news that someone from his family has fallen ill and requires immediate medical attention. In that case the obvious option for the man would be to forget about the work and return home.

The smart door comes into picture in such situations as it would allow the owner to let a caretaker, a neighbor or a medical professional to enter into the house and take quick measures even when the owner isn't physically present in the house. The system is also useful in industries where the employees and workers are the ones who seek access into the office premises. Lastly, there are certain points exposed to civilians where access is only given to certain qualified people. For instance, the switch board panel installed in buildings, trains and at many points in industries, are only required to be accessed by authorized personnel. Therefore, there is a need of a smart technology which covers up for all the situations listed above.

## II. Existing Systems

There are methods proposed in the literature that use different wireless communication standards like ZigBee, RFID [4], Bluetooth [7] and Wi-Fi [9] to exchange data and signaling between components. These systems offer a way to simply open the door either remotely or by connecting to the LAN.

There are some advanced systems worth mentioning. One of them is based on Raspberry Pi technology where cameras, keypad and pi-lids are being utilized. The authorized individuals are allotted user-IDs by the system. Whenever an authorized visitor wants to have access to the house, the system captures the face of the visitor through the camera and compares with the ones stored in the database. The visitor or else is asked to input a pass key. If the key is valid or the image of the visitor is matching with the image in the database, the door is opened. Else, the door is not opened and a photo is taken and sent to the owner of the house by e-mail and then if the owner of the house wishes to allow him to enter, an e-mail is sent by the owner to the database which should have entry code followed by "OK". A key and the image of the person is then stored in the database. The person is granted permission to enter any time by approving the authentication of the password in the database or his/her image [3].

There is another system which makes use of Raspberry Pi and Cameras along with speakers. This system allows to remotely access the door i.e. the owner can control the devices and appliances from anywhere around the globe. The owner

Amey Bairolu, Sanjukta Sheth, Binay Hazra and Arpit Rawankar are with Vidyalankar Institute of Technology, Wadala, Mumbai, 400037, India. (e-mail: ameybairolu@gmail.com, sanjuktasheth@gmail.com, binay.hazra21@gmail.com, arpit.rawankar@vit.edu.in)

can monitor the entrance whenever a visitor arrives. The owner can then give access to the visitor if he knows the visitor or else can interact with the visitor to know about him or to know the purpose of visiting [5]. It is also worth mentioning that RING, which is a known company for home security systems, uses the same approach in their systems [10].

## III. Technology

The proposed technology that the system mentioned in this paper employs the use of the micro-computer, Raspberry Pi, along with a camera, LCD display and a motor driving IC L293D for an electronic solenoid door lock. The camera is used to acquire face of the visitor which then compares the same with the ones in the database. The L293D gets the control pins from the raspberry pi. The respective output pins are given to the input terminals of the solenoid lock. The supply voltage Vs is 12V from an adapter so that the solenoid lock receives enough current for its operation. The overall connections and circuit diagram is displayed in the Fig 1.

The system also has a provision to remotely control the door lock via a webpage, which is hosted by the Raspberry pi globally, so that the owner can control the door from anywhere on Earth, provided the owner has internet access. So, if the visitor is an authorized person, the door lock opens on its own. Otherwise an e-mail, which contains the image of the visitor, is sent to the owner who can then refer to the image to check if he personally knows the visitor or not. If the owner knows the visitor, the owner can grant him access no matter where he is around the globe or can deny access otherwise. The time at which any visitor, including the owner, arrives, can then be uploaded to an Internet of Things API (like ThingSpeak [16]).

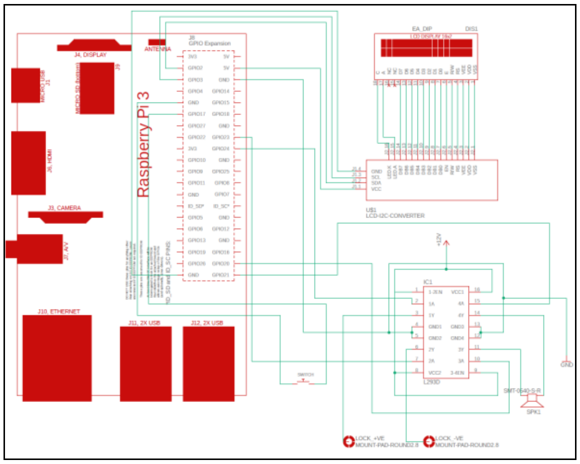


Fig 1: The Eagle schematic for depicting circuit connections

## IV. Methodology

### A. Face detection and Recognition

A camera is interfaced with the Raspberry Pi. For the overall process of face recognition, OpenCV library is used. A face recognition module is available on github [11] which uses the functions of OpenCV for face acquisition and pre-processing of the image. This Face recognition module has the accuracy of above 99% provided that many sample pictures for each authorized person should be available in the database. The owner has to create a folder in database for each authorized person. The name of these folders must be same as the name of the person whose pictures are saved in it. Then an appropriate .py script has to be run in order to obtain 128-d vectors for each authorized person.

When the visitor rings the bell, the camera starts acquiring frames of the area it is exposed to and then by using Haar cascade the faces in the frames are detected. By using Histogram of Oriented Gradients (HOG) algorithm [1], 128-d face embeddings are computed. The 128-d vector is then compared with the local database which contains 128-d vectors for every authorized person to check if the visitor is amongst one of them. The working of the face recognition module is depicted in the Fig 2 [12].

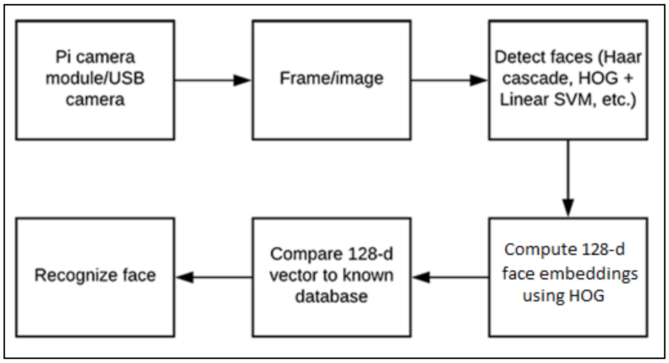

Pi camera module/USB camera
Frame/image
Detect faces (Haar cascade, HOG + Linear SVM, etc.)
Compute 128-d face embeddings using HOG
Compare 128-d vector to known database
Recognize face

Fig 2: Flowchart of Face Recognition as depicted in the Face Recognition module by

If the face of the visitor is recognized, the door will open on its own and the name of the visitor is uploaded on ThingSpeak. If the visitor is not recognized, the raspberry pi captures the image of the visitor through the camera and sends an e-mail to the owner through internet over Simple Mail Transfer Protocol (SMTP).

### B. Remotely giving Commands to the Raspberry pi

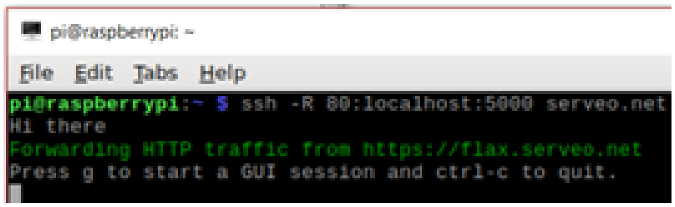
Once the owner receives an e-mail from the system, he can refer to the image attached to it to check if the visitor is a known person or not. Then the system should provide a facility to the owner to grant or deny access to the visitor.

This facility is provided by using a webpage which is designed using Flask, which is a micro web framework written in Python, which has the routes defined to open the door or to display a message that access has been denied [14]. This webpage is initially hosted locally by the raspberry pi. To host the webpage globally, there is a need to forward the port on which the webpage is hosted. A free tool is available online, known as serveo, which is an SSH server used for remote port forwarding [13]. Another advantage of using serveo is that there is no additional set ups required. We should just know the port on which web server is hosted and then use the terminal command.

Once a webpage is hosted globally with the help of serveo, a public URL is displayed that anybody can access. By using this URL, the user can login to the webpage hosted by raspberry pi, by filling proper login details, to be able to send controlling commands to GRANT ACCESS or to DENY ACCESS.

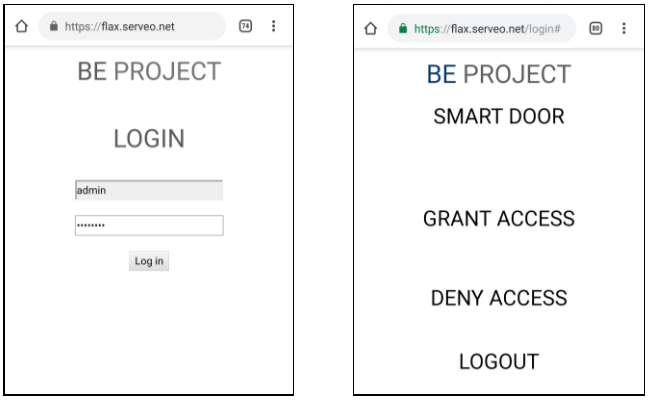
Note that, the port on which the local webpage is hosted for Raspberry Pi is 5000. The webpage is hence hosted by using the following command on the terminal:

```
ssh -R 80:localhost:5000 serveo.net
```



```
pi@raspberrypi:~ $ ssh -R 80:localhost:5000 serveo.net
Hi there
Forwarding HTTP traffic from https://flax.serveo.net
Press g to start a GUI session and ctrl-c to quit.
```

Fig 3: Exposing locally hosted webpage on port 5000 to the internet



https://flax.serveo.net

BE PROJECT

LOGIN

admin

••••••••

Log in

https://flax.serveo.net/login#

BE PROJECT

SMART DOOR

GRANT ACCESS

DENY ACCESS

LOGOUT

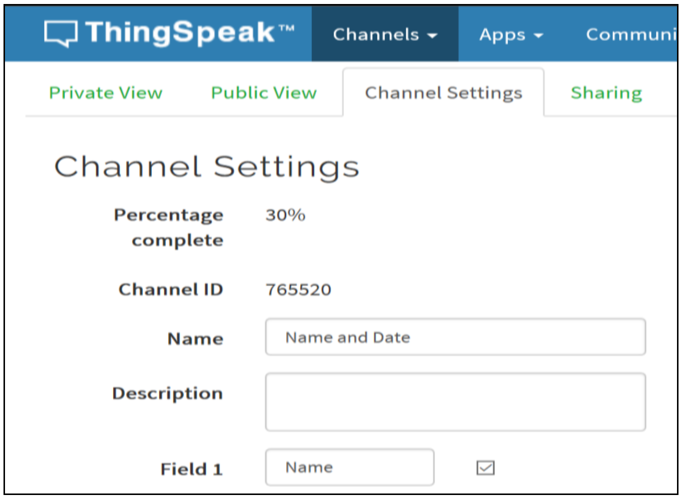
(a) (b)

Fig 4: (a) Accessing the webpage by using the URL provided by Serveo after port forwarding, (b) After entering the login credentials, the page to remotely access the door is displayed

Once access is granted to the unknown visitor, ‘Unknown’ string is uploaded on ThingSpeak.

### *C. Uploading the name of the visitor on ThingSpeak*

Whenever a visitor is given access, either by the system when visitor is an authorized person or by the owner when owner personally knows the visitor, we can upload the details of the visit, such as name of the visitor and at what time he arrives. ThingSpeak allows us to achieve this easily. It goes well with almost every IoT based applications. The only thing required to be done on Raspberry Pi to be able to upload names is importing a library called urllib2. All the necessary functions are available in this library to support name uploading. The steps to setup ThingSpeak for our application includes signing up, creating a new channel and defining a Field as ‘Name’.



Channel Settings

Percentage complete 30%

Channel ID 765520

Name: Name and Date

Description:

Field 1: Name ☑

Fig 5: Creating a channel on ThingSpeak to upload names of the visitors

Once a channel is created, API request to update a Channel feed can be used to successfully upload names on the desired channel under a desired field [16].

## V. Algorithm

The overall flowchart for the SMART DOOR is displayed in the Fig 6.

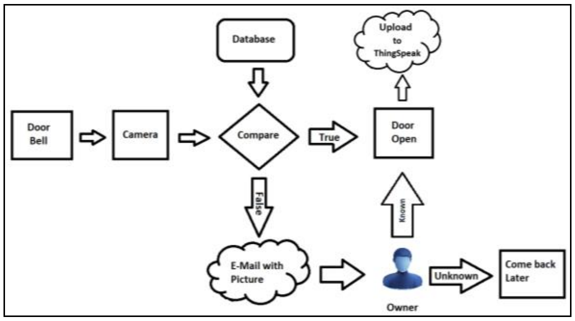


Fig 6: Flowchart of SMART DOOR from ringing the bell to granting/denying access to the visitor.

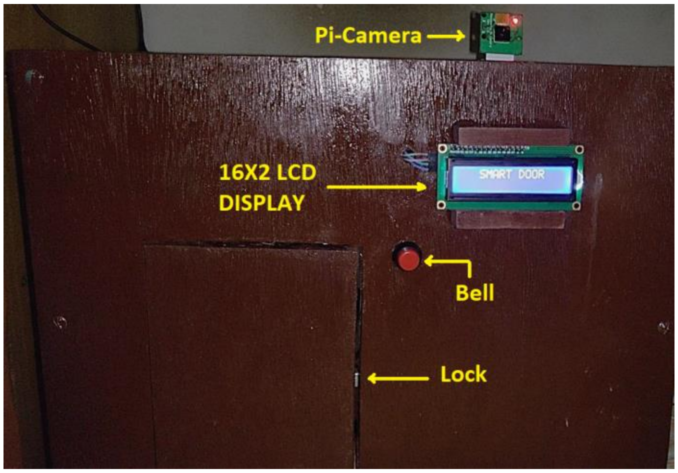


Fig 7: Relative positions of the peripherals connected to the raspberry pi, outside of the house and near the entrance.

Step 1: The visitor rings the bell.
Step 2: The LCD displays “Look towards the camera” so that the camera can acquire the face of the visitor. The visitor is

expected to look towards the camera until the system displays that the face either has been recognized or was not recognized.
Step 3: If the face is recognized, the name of the visitor is uploaded to the ThingSpeak channel.
Step 4: If the face was not recognized then an image of the visitor, taken by the camera is sent to owner via e-mail.
Step 5: The owner will have to login to the webpage which was globally hosted by the raspberry pi by using valid login credentials.
Step 6: The owner may then either grant access or deny access to the visitor.
Step 7: If the owner grants access to the visitor, a string ‘Unknown’ is uploaded to the ThingSpeak channel to mark the arrival of the visitor.
Step 8: If the owner does not grant access to the visitor, the LCD displays “Access denied”.
Step 9: If the owner fails to respond to the e-mail on time, the LCD displays “Come back later”.

## VI. Results

### *A. Actions taken by the system after successfully recognizing or failing to recognize the visitor*

After ringing the bell, the system starts recognizing the face of the visitor and displays the name of the visitor when recognized on the LCD display.

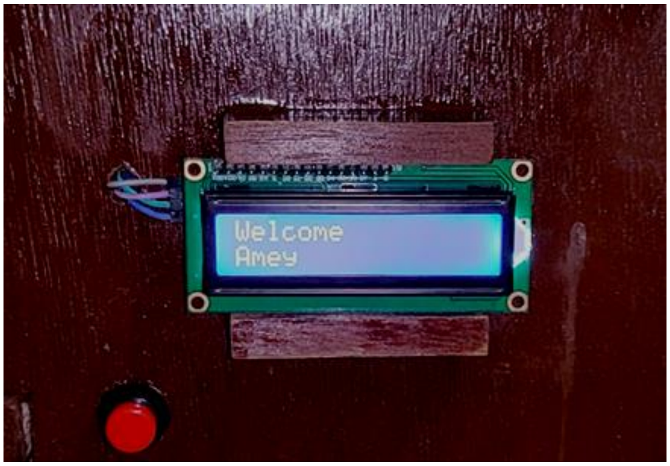
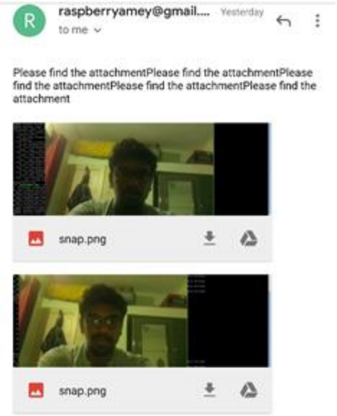


Fig 7: LCD displaying name of the recognized visitor.

When the system does not recognize the visitor the LCD displays the message “Unknown; Owner deciding” and sends an e-mail to the owner which consists the image of the visitor.



raspberryamey@gmail.... Yesterday
to me

Please find the attachmentPlease find the attachmentPlease find the attachmentPlease find the attachmentPlease find the attachment

snap.png

snap.png

Fig 8: Owner receives an e-mail containing the picture of the unrecognized visitor

### *B. On Granting Access or Denying Access through the webpage*

On granting access to the known visitor by the owner via webpage, the solenoid lock gets opened and the LCD displays “You may enter” (Fig 9 (a)). The visitor can then walk through the door. And if the owner does not grant access, the LCD displays “Access denied” and does not change the state of the lock (Fig 9 (b)).

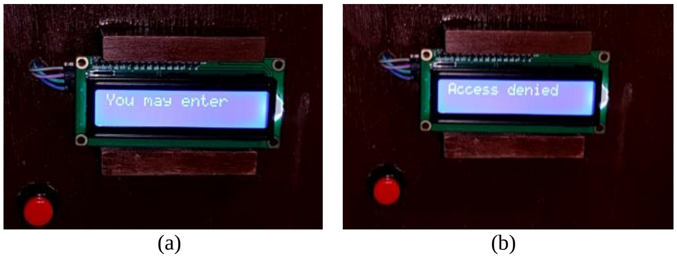
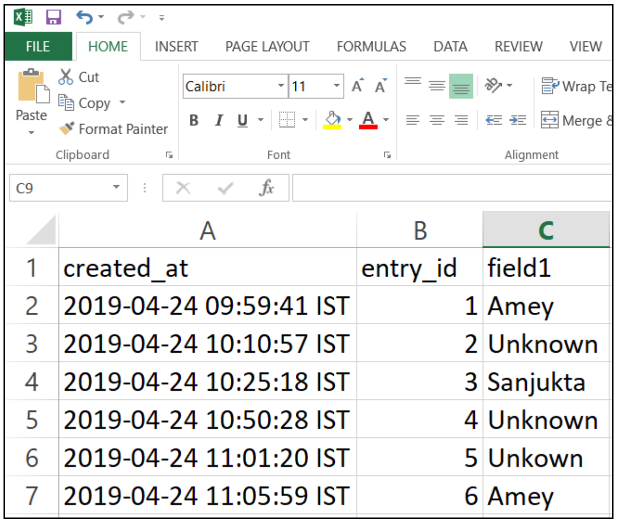


Fig 9: Message displayed on LCD when (a) access is granted (b) access is denied

### *C. Entries made on ThingSpeak*

The names of the visitors which were uploaded when access was granted by the system or the uploaded “Unknown” string when the owner granted them the permission, can be viewed by downloading the CSV file from the tab Data Import/Export. The exact time of occurrence of each event is also provided. The CSV file can be viewed by using Microsoft Excel as shown in Fig 10.



| | A | B | C |
|---|---|---|---|
| 1 | created_at | entry_id | field1 |
| 2 | 2019-04-24 09:59:41 IST | 1 | Amey |
| 3 | 2019-04-24 10:10:57 IST | 2 | Unknown |
| 4 | 2019-04-24 10:25:18 IST | 3 | Sanjukta |
| 5 | 2019-04-24 10:50:28 IST | 4 | Unknown |
| 6 | 2019-04-24 11:01:20 IST | 5 | Unkown |
| 7 | 2019-04-24 11:05:59 IST | 6 | Amey |

Fig 10: CSV file containing the list of visitors and time at which they arrived, viewed using Microsoft Excel.

## VII. Conclusion

The main idea of this paper was to provide a better way to deal with people seeking access to enter the premises employing this system. And we have done so by not just prioritizing security but also by providing a solution to deal with emergency situations as our linchpin aspect. The system proposed in this paper therefore uses the face as the key to open the door when the visitor is known to the system, thereby eliminating the need of a traditional lock and key, ID cards, etc. With extension to this, the system also lets the owner to decide whether an unknown visitor should be allowed to enter or not, even when the owner is on the other side of the world. The overall system hence, helps to tackle some more unavoidable situations. The proposed system is also cheap and makes use of the resources which the existing systems already are employing. The system is easy to setup and is also user-friendly.

This presented paper work can further be extended by adding a provision to let the system upload the name of the unknown visitors on the cloud.

An additional feature can be included to assist the delivery persons from various e-Commerce businesses i.e., if no one is home and a delivery executive arrives, a slot can be provided to allow him to drop the delivery.

## References

[1] SHU Chang, DING Xiaoqing, FANG Chi, “Histogram of the Oriented Gradient for Face Recognition”, TSINGHUA SCIENCE AND TECHNOLOGY ISSN 1007-0214 15/15 pp216-224 Volume 16, Number 2, April 2011

[2] Bhaumik Vaidya, Ankit Patel, Anand Panchal, Rangat Mehta, Krish Mehta, Parth Vaghasiya, “Smart home automation with a unique door monitoring system for old age people using Python, OpenCV, Android and Raspberry pi”, 2017 International Conference on Intelligent Computing and Control Systems (ICICCS)

[3] Naser Abbas Hussein, Inas Al Mansoori, “Smart Door System for Home Security Using Raspberry pi3”, 2017 International Conference on Computer and Applications (ICCA)

[4] Yong Tae Park, Pranesh Sthapit, Jae-Young Pyun, “Smart digital door lock for the home automation”, TENCON 2009 - 2009 IEEE Region 10 Conference

[5] Rishabh Kumar Gupta, Bala Murugan S, Aroul Kris, Marimuthu Ramakrishnan, “IoT Based Door Entry System”, Indian Journal of Science and Technology October 2016

[6] Faiz Aman, C Anitha, “Motion sensing and image capturing based smart door system on android platform”, 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS)

[7] Siddhi Kavde, Riddhi Kavde, Sonali Bodare, Gauri Bhagat, “Smart digital door lock system using Bluetooth technology”, 2017 International Conference on Information Communication and Embedded Systems (ICICES)

[8] Pradip Tilala, Anil K. Roy, Manik Lal Das, “Home access control through a smart digital locking-unlocking system”, TENCON 2017 - 2017 IEEE Region 10 Conference

[9] Abdallah Kaseem, Sami El Murr. Georges Jamous, Elie Saad, Marybelle Geagea, “A Smart Lock System using Wi-Fi Security”, 2016 3rd International Conference on Advances in Computational Tools for Engineering Applications (ACTEA)

[10] Ring, https://ring.com/

[11] Davis King, Adam Geitgey, Face Recognition module, https://github.com/ageitgey/face_recognition

[12] Adrian Rosebrock, Raspberry pi Face Recognition guide, https://www.pyimagesearch.com/2018/06/25/raspberry-pi-face-recognition/

[13] Serveo to host webpage globally or port forwarding, https://serveo.net/

[14] Guide for Flask Programming, http://mattrichardson.com/Raspberry-Pi-Flask/

[15] Using Raspberry Pi and Python to send e-mail alerts, https://makersportal.com/blog/2017/9/23/using-raspberry-pi-and-python-to-send-email-alerts

[16] ThingSpeak, https://thingspeak.com/