An IoT Based Intruder and Smoke Monitoring System

Vipa Thananant
Faculty of Administration Business and Information Technology, Rajamangala University of Technology Tawan-Ok, Bangkok, Thailand
vipa_th@rmutto.ac.th

Chumpol Mokarat
Faculty of Administration Business and Information Technology, Rajamangala University of Technology Tawan-Ok, Bangkok, Thailand
chumpol_mo@rmutto.ac.th

ABSTRACT

Burglar and fire are the two dangerous life events. An IoT based intruder and smoke monitoring system is proposed to protect us from these events. The proposed system is a low cost and energy efficient alternative to the existing systems. It is an integrated system that can detect and monitor theft and fire. The system consists of hardware and software parts. The hardware parts have two separated subsystems, smoke and intruder subsystem. The smoke subsystem uses an ESP8266 Arduino board with a smoke sensor (MQ2) and a buzzer. The intruder subsystem uses an ESPIno32CAM board with an ultrasonic sensor (HC-SR04). The software parts consist of three applications that are web, android, and LINE application. Both subsystems share the same web, android, and LINE application. When a smoke is detected by the sensor, the alarm sounds. The smoke subsystem sends a notification to the web and LINE application and the gas concentration level is sent to MySQL database. When an intruder is detected by the sensor, the intruder subsystem captures his photo and sends it to the LINE application for notification. The system also has the ability to recognize faces so that when a homeowner is detected by the sensor, no notifications are displayed in LINE and web application. Web application shows an alert message when there is an intruder and a smoke. It also shows a graph of gas level values. Android application can turn on and off the device in the system.

CCS CONCEPTS

• **Security and privacy**; • **Security services**; • **Authentication**; • **Biometrics**; • **Software and application security**; • **Web application security**;

KEYWORDS

IoT real time system, Intruder system, Smoke system, MySQL, Face detection, Face recognition, ESPIno32CAM

ACM Reference Format:
Vipa Thananant and Chumpol Mokarat. 2021. An IoT Based Intruder and Smoke Monitoring System. In *2021 7th International Conference on Computing and Artificial Intelligence (ICCAI '21), April 23–26, 2021, Tianjin, China.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3467707.3467778

ICCAI '21, April 23–26, 2021, Tianjin, China

ACM ISBN 978-1-4503-8950-1/21/04…$15.00
https://doi.org/10.1145/3467707.3467778

1 INTRODUCTION

An IoT based intruder and smoke monitoring system can protect us from theft and fire. In the major cities of Thailand, there are many housing estates from small (less than 50 detached houses) to big size (more than 200 detached houses). Most housing estates have a guard room in front of them for a security checkpoint of people in and out all day all night. The web application of the system can be installed in this guard room. When an intruder or a fire occurs in any house, the security guard gets the alert message which house has a problem in the web application. Therefore, he can give help immediately. The hardware parts can be installed in each house and the android and LINE application can be installed in the homeowner's mobile phone. We select LINE application for notification because LINE is very popular in Thailand. In 2020, LINE has 45 million users in Thailand out of 66 million people.

Nowadays, face recognition technology is available in many businesses such as airline, travel, hotel, and bank. It is applied in payment systems, access control systems, authentication systems, and others. Due to its complex computation, large storage, and processing capabilities, most of them are processed on high performance devices such as a computer. But face recognition on low performance devices such as embedded device [1] is used widely. The embedded system processor can be broken down into MPU (Microprocessor Unit) and MCU (Microcontroller Unit). MPU has more processing power than MCU but less than a computer. MCU integrated into a single chip has limited resources. MCU is cheaper and more power efficient than MPU. To make face recognition tasks perform efficiently on small MCU devices is challenging. MCU, ESP32Wrover-IB, with MTMN [2] and FRMN [3] model can perform the face detection and recognition efficiently. ESP32Wrover-IB is the powerful MCU of Espressif Systems. MTMN was developed based on mobile architecture mobileNetV2 [4] and multitask cascaded convolutional networks [5] to perform face detection and alignment. FRMN was developed based on the ArcFace algorithm [6] to perform face recognition. Face recognition steps based on MTMN and FRMN models are shown in Figure 1. First, the camera takes a photo. Second, the image enters the face detection and alignment process. Third, the aligned face image enters the face recognition process. Figure 2 shows the details of the face detection and alignment process. It starts with image scaled to various sizes to create overlapping images as an input to the P-Net process. Candidate frames are then created to go to the R-Net process where potential candidate frames are selected. The O-Net process gives five important landmarks on the face. Figure 3 shows the details of the face recognition process. The output of the O-Net process goes to the face recognition process that generates the face identity or face ID. This face ID is compared with all the faces registered in a

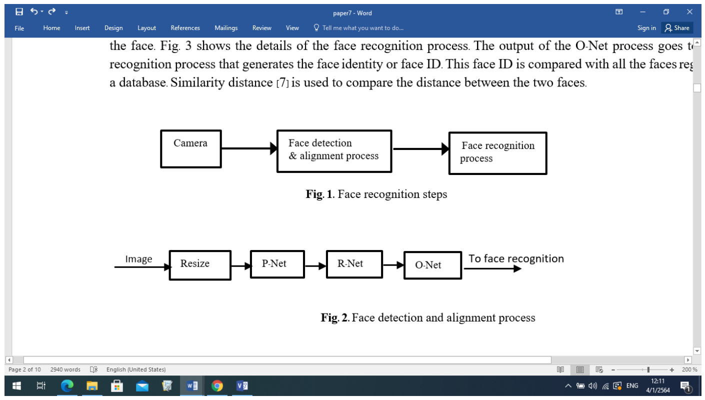

Figure 1: Face recognition steps.

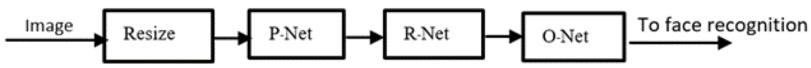

Figure 2: Face detection and alignment process.

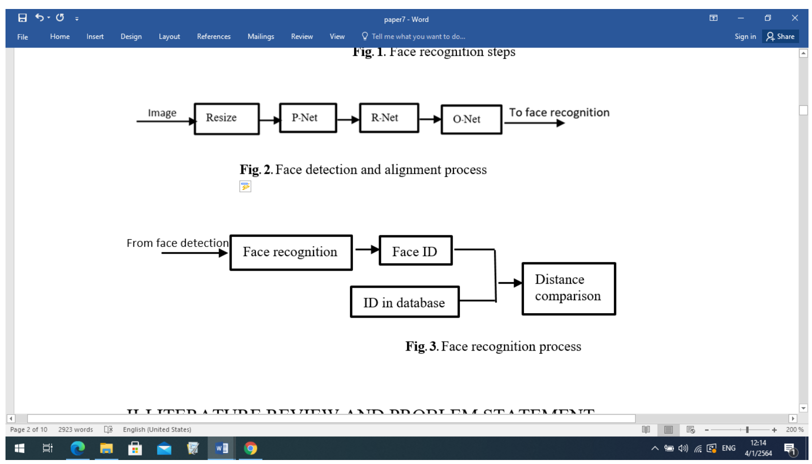

Figure 3: Face recognition process.

database. Similarity distance [7] is used to compare the distance between the two faces.

2 LITERATURE REVIEW AND PROBLEM STATEMENT

Many researchers discussed intruder and smoke monitoring system. Philip et al. [8] proposed the alarm system for gas, smoke, fire, and movement via SMS. The system had Arduino promini board with 4 sensors, IR flame, MQ4 gas, MQ2 smoke, and PIR. It used SIM 800 GSM module to send alerts via SMS. Kunal et al. [9] presented home security systems using 3 sensors, TMP36, MQ2 and PIR. When the temperature and smoke level were high, the system sent an SMS notification and made a voice call. Sajid et al. [10] proposed the home security systems that had Raspberry-pi with DHT11, MQ135, PIR, and GSM modem. The system sent the intruder and fire alert via SMS. The Pi-Camera was used to record intruder and fire incidents video that was stored in the Dropbox cloud storage. Su et al. [11] presented the home security system using Arduino Mega board connected with DHT11, Hall effect, MQ2, and PIR. The system used Arduino Ethernet shield to send all sensor reading values to MySQL database. These values were displayed on the web page in real time. Deepak et al. [12] proposed an intruder tracking and control system using a Raspberry-pi attached to a PIR sensor. The web camera took a picture when the sensor found an intruder and a warning email was sent. The owner could control the door lock anywhere. Ahmed et al. [13] presented a fire alarm using the Arduino Mega with TEMT6000 (light intensity) and MQ2 sensor. When the smoke level exceeded the threshold value, ArduCam took a photo. The photo

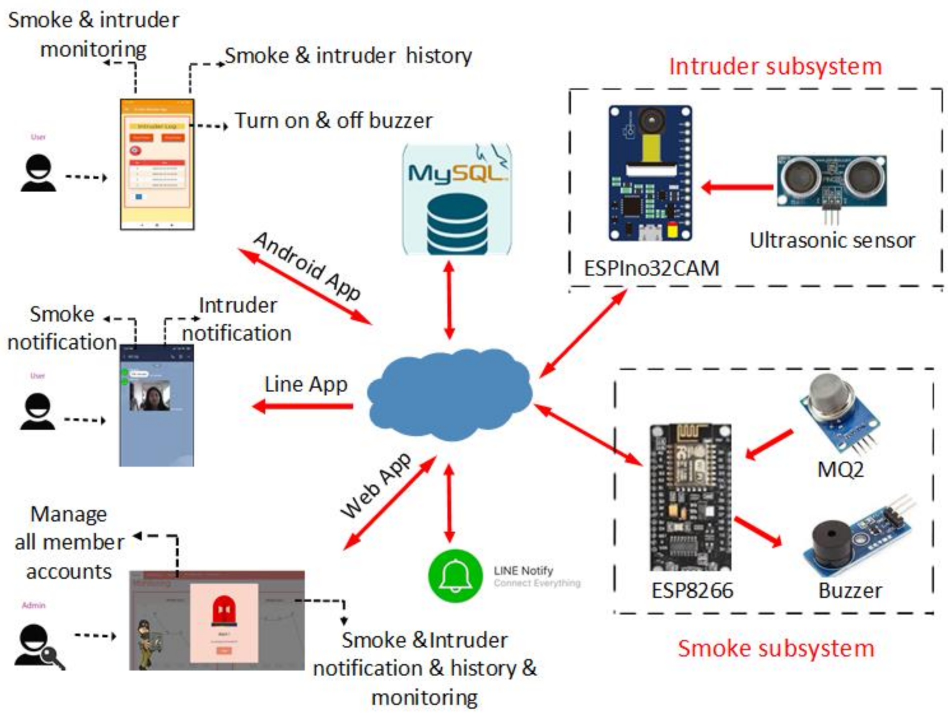

Figure 4: The overall framework of IoT based intruder and smoke monitoring system.

and place location were sent to the system administrator. Akash et al. [14] proposed the security control system using a Raspberry-pi attached to PIR, vibration sensor, air quality sensor, magnetic door lock, and camera. The system could send the email and control the door lock. Also, the system sent SMS when it could detect an intruder and fire. Akkasit and Naruepon [15] proposed an intruder security system that had Raspberry-pi attached to a camera, motions sensor, and siren. The system would turn on the light and siren and at the same time capture the intruder picture to send to LINE notification.

Most proposed systems mentioned above used Raspberry-pi (MPU) as a component in the system that it was not energy saving and costly. The authors propose low cost and energy saving ESPIno32CAM (MCU) as a component in IoT based intruder and smoke monitoring system to do face detection and recognition in intruder subsystem. The face recognition capability enables the system to distinguish the intruder's face from the homeowner's face. Thus the system can choose whether to send or not to send the notifications to homeowners and security guards.

3 PROPOSED SYSTEM

The overall framework of IoT based intruder and smoke monitoring system is shown in Figure 4. The system is divided into hardware and software parts. The network communication between hardware and software parts is Wi-Fi. The data transfer messaging protocol is HTTP. The hardware parts consist of two subsystems that are smoke and intruder subsystems. The software parts consist of three applications that are web, android, and LINE application. Both subsystems share the same web, android and LINE application.

The smoke subsystem has MCU (Esp8266) connected to passive buzzer and gas sensor (MQ2) that can capture flammable gas such as LPG, Methane, as well as smoke caused by gas combustion. This subsystem keeps monitoring the gas concentration level and storing this level into MySQL database. When the level is over the threshold value, the buzzer makes a loud sound and the notification message is sent to LINE API. The alert message is shown on a web and LINE application immediately. The homeowner can log in to the android application, see the gas concentration level at that time and turn off the buzzer sound.

The intruder subsystem has MCU (ESPIno32CAM) connected to the ultrasonic sensor (HC-SR04) that measures the intruder distance away from a sensor. ESPIno32CAM is a board with 32 bit dual core ESP32, WI-FI, Bluetooth, 240 MHz, 8 MB PSRAM, 16 MB flash memory, and CMOS Camera OV2640. MTMN [2] and FRMN [3] models perform the face detection and recognition in ESPIno32CAM board. The number of training images for face recognition is 6. The intruder subsystem keeps monitoring the intruder's distance. When the intruder walks pass the sensor, the camera captures the intruder's face. If this face has been registered in the system, the system takes no action because the system can recognize that the intruder's face is the same as the registered face. If the face doesn't match, data is stored in MySQL database and the notification message is sent to LINE API. Also, the alert message is shown on the same web and LINE application as the smoke subsystem.

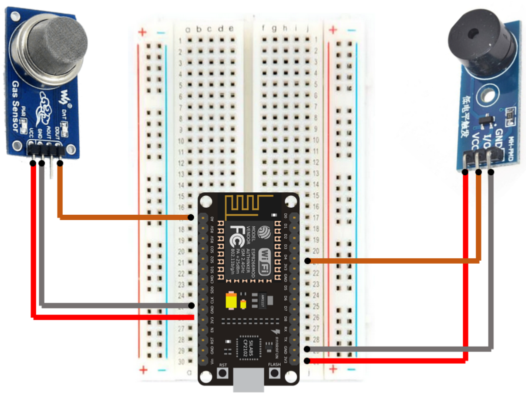

Figure 5: Smoke subsystem schematic diagram.

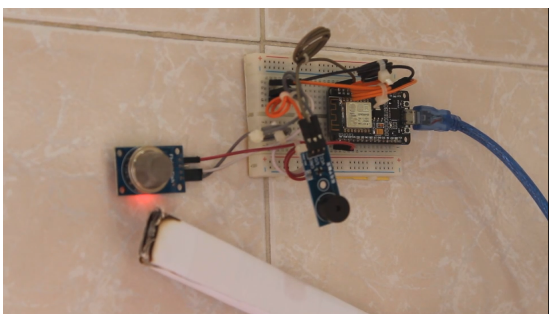

Figure 6: Burning paper to produce smoke.

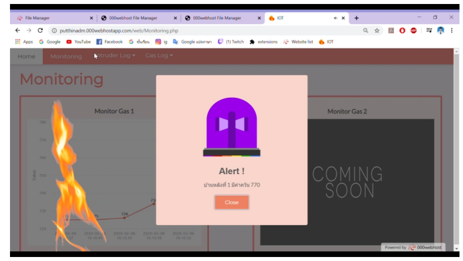

Figure 7: Smoke notification in web app.

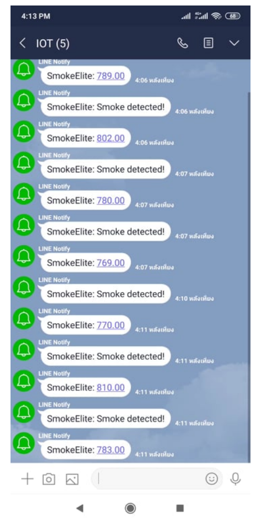

Figure 8: Line notification.

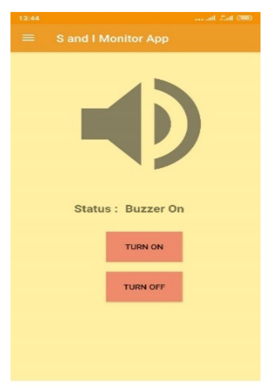

Figure 9: Turn on/off button in android.

4 EXPERIMENTS AND RESULTS

4.1 Smoke Subsystem

A schematic diagram of the smoke subsystem is shown in Figure 5. The smoke subsystem was installed in the kitchen. The experiment was conducted by burning paper and putting out the fire to produce smoke, as shown in Figure 6. When the gas concentration level reached the threshold values, the buzzer made a loud sound. At the same time, the web application that was installed in the guard control room automatically displayed a warning message with sound, as shown in Figure 7. The warning message could show which house was on a fire. Moreover, the homeowner immediately received the notification in the LINE application, as shown in Figure 8. Web and android application displayed graph of the gas concentration level in real time, as shown in Figure 7. After the homeowner signed in to the android application, he could turn off the sound, as shown in Figure 9

Four functions were performed after the gas concentration level reached the threshold values. They were 1) real time smoke notification 2) real time smoke monitoring, 3) smoke history and 4) real time turn on/off the sound. The experiment was performed 30 times. The results are shown in Table 1

It can be seen from Table 1 that the smoke subsystem has a 100% success rate for all system functions.

4.2 Intruder Subsystem

A schematic diagram of the intruder subsystem is shown in Figure 10. The intruder subsystem was installed in front of the door, as shown in Figure 11. The installed location could be anywhere that saw the intruder face as much as possible. To demonstrate that the system could detect any objects, we performed an experiment by holding a toy in front of the sensor, as shown in Figure 12.

Table 1: The experimental functional results of smoke subsystem

Functions	Success rate (%)
1. Real time smoke notification	
1.1 in web app (Figure 7)	100
1.2 in LINE app (Figure 8)	100
2. Real time smoke monitoring	
2.1 in web app (Figure 7)	100
2.2 in android app	100
3. Smoke history	
3.1 in android app	100
3.2 in web app	100
4. Real time turn on/off sound	
4.1 in android app (Figure 9)	100
Average	100

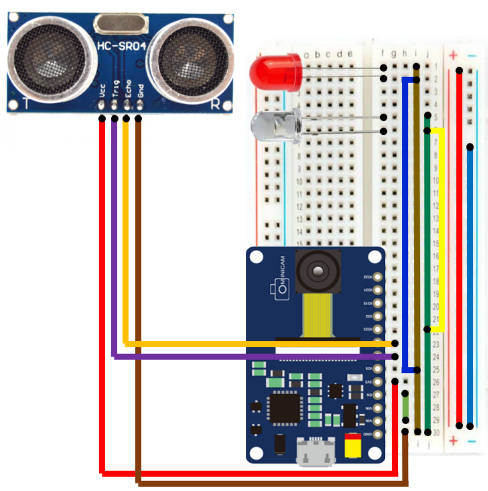

Figure 10: Intruder subsystem schematic diagram.

The system recognized that the object was not the homeowner face, therefore the same web application as the smoke subsystem automatically displayed an alert message, as shown in Figure 13. The alert message could show which house had an intruder. The homeowner immediately received the notification with the toy picture in LINE application, as shown in Figure 14. To show that the system could detect an intruder face, we conducted an experiment by letting the intruder walked pass the sensor. The system could detect the intruder face, as shown in Figure 15. The notification was displayed in both web and LINE application, as shown in Figure 13 and Figure 16.

Two functions were performed when an intruder was detected. They were 1) real time intruder notification and 2) intruder history. The third function is face recognition. It was performed when a homeowner, had already registered his face into the system, was detected. The experiment was performed 30 times. The results are shown in Table 2

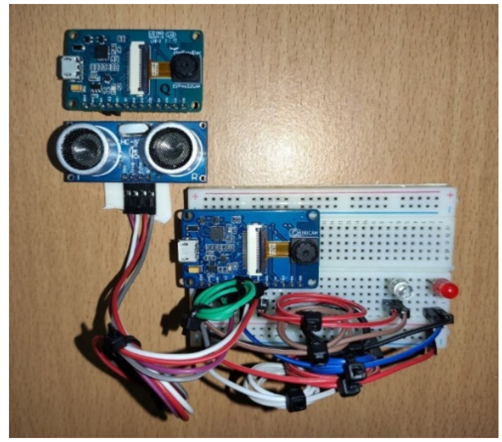

Figure 11: Intruder subsystem installation.

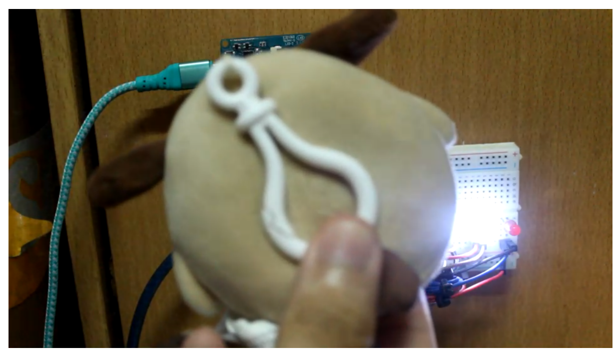

Figure 12: Holding a toy in front of the sensor.

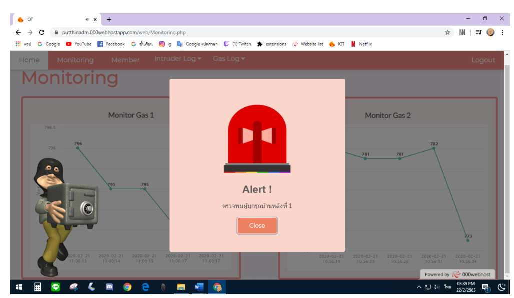

Figure 13: Intruder notification in web app.

It can be seen from Table 2 that the average success rate of the intruder subsystem is 97.3%. The intruder subsystem has a 100% success rate for all functions except for face recognition function (3.1 and 3.2). This success rate depends on the success rate of face detection and recognition in ESPIno32CAM board that is described in the next section.

Figure 14: Toy picture in Line app.

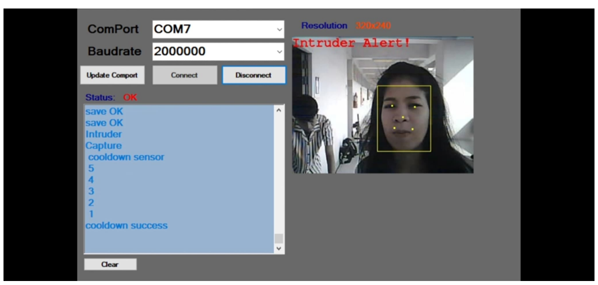

Figure 15: Detecting intruder face.

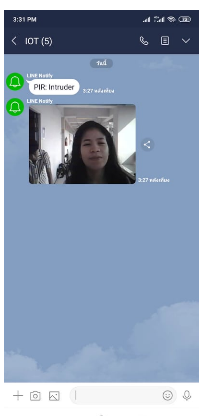

Figure 16: Intruder picture in Line.

4.3 Face Detection and Recognition in ESPIno32CAM

We performed an experiment to determine the number of faces the system should train for maximize result of detection and recognition process in ESPIno32CAM. The experiment was conducted by 20 people. The first step (training image) was to register individual face in a local database by taking one photo, as shown in Figure 17. The second step was to match the new image being displayed from the camera with the registered images of step 1 for ten times. If the new image matched the registered image, the message "Hello Subject + face ID" would be displayed on the screen, as shown in Figure 18. On the other hand, if the image did not match, an intruder alert message would be displayed, as shown in Figure 15. The third step was to count the number of recognition success that the system could recognize out of ten times. Then repeated steps 1-3 for the same person but increased the number of training images in step 1, by 1, until the number of training images was 10. Therefore, each one would repeat steps 1-3 ten times, and each time one additional photo was taken for training.

Table 2: The experimental functional results of intruder subsystem

Functions	Success rate (%)
1. Real time intruder notification	
1.1 in web app (Figure 13)	100
1.2 in LINE app (Figure 16)	100
2. Intruder history	
2.1 in android app	100
2.2 in web app	100
3. Face recognition	
3.1 in web app (No notification)	92
3.2 in LINE app (No notification)	92
Average	97.3

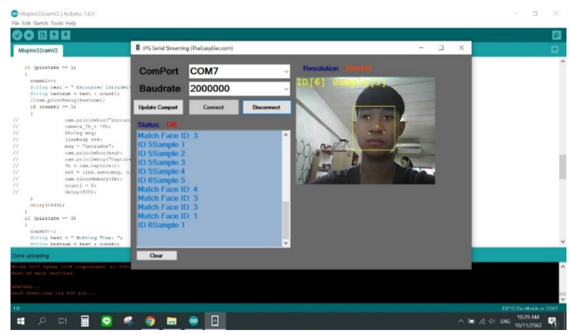

Figure 17: Face registration.

The recognition success rate (RSR) is shown in Table 3. The RSR is calculated from (1).

$$RSR = (number of recognition success) / (total number of trials) \times 100 \quad (1)$$

It can be concluded from Table 3 and Figure 19 that when the number of training images increases, so does the recognition success rate until it reaches the point (training images=6) that the rate maintains saturated. Therefore, we use 6 training images in the face recognition process of the experiment 4.2 to achieve the maximum recognition success rate.

Table 3: RSR of different number of training images

# Training images Person	1	2	3	4	5	6	7	8	9	10
# 1	40	40	50	50	40	70	70	80	80	90
# 2	50	60	70	80	80	100	90	100	90	90
# 3	50	60	60	80	80	100	90	90	100	100
# 4	60	70	80	100	90	100	100	100	90	90
# 5	70	70	80	90	100	90	90	100	100	90
# 6	60	60	70	80	90	80	90	100	90	80
# 7	60	70	80	90	100	100	100	100	100	100
# 8	60	60	60	60	70	90	90	90	90	100
# 9	40	50	50	60	60	100	100	100	100	100
# 10	60	60	70	70	100	100	100	80	90	90
# 11	50	50	60	60	60	100	90	90	100	100
# 12	60	60	70	70	80	100	100	100	90	90
# 13	50	60	70	70	70	90	90	80	80	80
# 14	60	70	80	80	100	90	100	100	90	100
# 15	70	70	80	80	90	80	80	80	100	90
# 16	60	60	70	80	80	100	100	90	90	100
# 17	70	70	80	90	100	80	100	80	80	100
# 18	60	60	60	60	70	100	90	90	100	90
# 19	60	60	70	70	70	90	80	100	90	80
# 20	70	70	70	70	80	90	90	90	100	100
Average	58	61.5	69	74.5	80.5	92.5	92	92	92.5	93

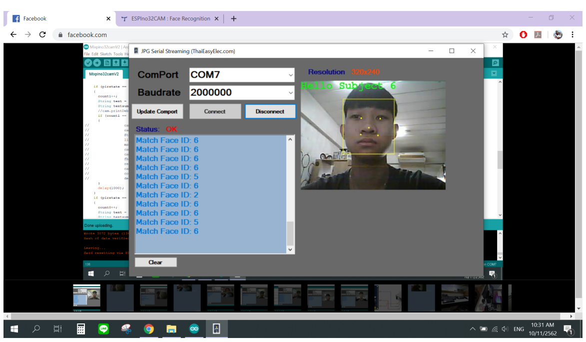

Figure 18: Face matching.

5 CONCLUSION AND FUTURE WORK

Our research goal is to design and implement a low cost and reliable intruder and smoke monitoring system. There are two separate subsystems, smoke and intruder. But they share the same web, android, and LINE application. The smoke subsystem has a 100% success rate for all system functions. The success rate of the intruder subsystem is 97.3%. The system can monitor several houses simultaneously. The system can alert which house has a fire or an intruder. The homeowners can log in to the android application and can control their own devices and see the information that only belongs to them. In the intruder subsystem that needs to perform demanding tasks such as face recognition, usually MPU (Raspberry Pi) is proposed. However, we propose MCU (ESPIno32CAM) that is smaller and cheaper. Most of the existing systems use PIR sensor to detect the intruder's motion. We find out that an ultrasonic sensor can detect an intruder better than PIR. For future work, we can increase the success rate of the intruder subsystem by replacing MTMN [2] and FRMN [3] model with other models such as genetic algorithm, Haar-Like face detection algorithm, Eigenface recognition algorithm. Also, we can add more functions to the system such as controlling more home devices.

REFERENCES

[1] Abbas Bigdeli, Colin Sim, Morteza Biglari-Abhari, and Brian C. Lovell. 2007. Face detection on embedded systems. In Proceedings of the Third International Conference on Embedded Software and Systems, ICESS, Daegu, Korea, 295-308.
[2] Yuan Jiong and Ye Hang Yang . 2020. Retrieved Dec 1, 2020 from https://github.com/espressif/esp-face/tree/master/face_detection
[3] Yuan Jiong and Ye Hang Yang . 2020. Retrieved Dec 1, 2020 from https://github.com/espressif/esp-face/tree/master/face_recognition
[4] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chidh Chen. 2018. MobileNetV2: Inverted residuals and linear bottlenecks. In Proceedings of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, 4510-4520. https://doi.org/10.1109/CVPR.2018.00474
[5] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. 2016. Joint face detection and alignment using multitask cascaded convolutional networks. IEEE Signal Processing Letters. 23, 10 (October 2016), 1499-1503. https://doi.org/10.1109/LSP.2016.2603342
[6] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. 2019. ArcFace: Additive angular margin loss for deep face recognition. In Proceedings of 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, 4685-4694. https://doi.org/10.1109/CVPR.2019.00482
[7] Arjun Mane, Ramesh Manza, and Karbhari Kale. 2010. The role of similarity measures in face recognition. International Journal of Computer Science and Application IJCSA 1 (January 2010), 62-65.
[8] Philip F.O. Kpae, Macmammah Michael, Ezekiel D. Nsikak, Woji T Chigozinum., Abubakar Abdullahi, and Kpaa O.N. Friday .2018. Design and implementation of a smart home (smoke, fire, gas and motion detector). International Research Journal of Engineering and Technology (IRJET), 5, 8 (August 2018), 1-34.
[9] Kunal Kumar, Navneet Sen, Sheikh Azid, and Utkal Mehta. 2016. A fuzzy decision in smart fire and home security system. Procedia Computer Science, 105, 1 (December 2017), 93-98. https://doi.org/10.1016/j.procs.2017.01.207

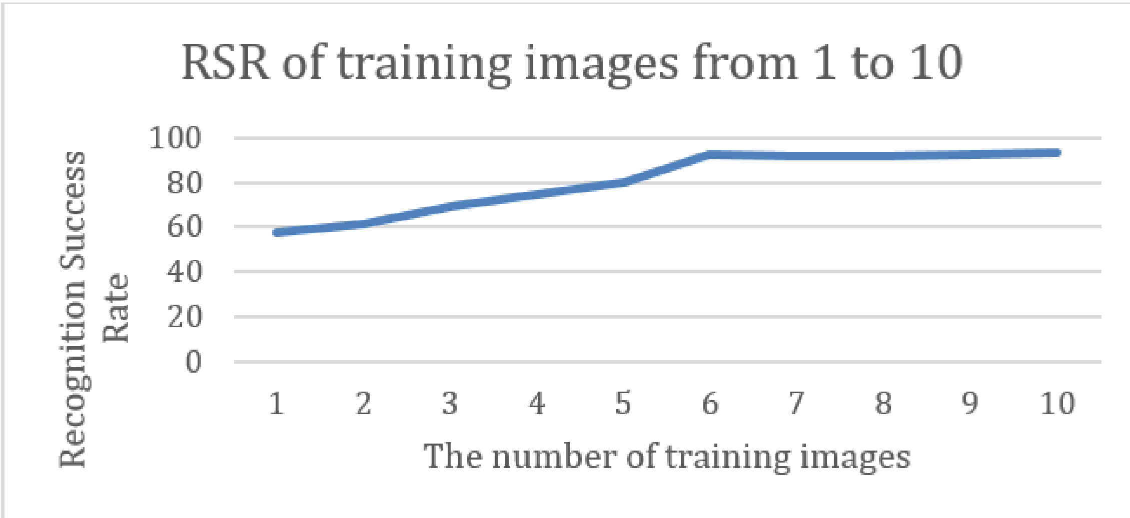

Figure 19: RSR Graphical representation for different number of training images.

[10] Sajid M. Sheikh, Modise K. Neiso, and Fatma Ellouze. 2019. Design and implementation of a raspberryPI based home security and fire safety system. International Journal of Computer Science and Information Technology (IJCSIT), 11, 3 (June 2019), 13-27. https://doi.org/10.5121/IJCSIT.2019.11302

[11] Su Z. Z. Win, Zaw M. M. Htun, and Hla M. Tun. 2016. Smart security system for home appliances control based on internet of things. International Journal of Scientific and Technology Research, 5, 6 (June 2016), 102-107.

[12] Deepak K. Nauak, Prabakaran N., Nitish Kumar, and Madhav P. Khanal. 2017. IoT based smart surveillance monitoring and controlling system. International Journal of Pure and Applied Mathematics, 117, 18 (December 2017), 239-242.

[13] Ahmed Imteaj, Tanveer Rahman, Muhammad K. Hossain, Mohammed S. Alam, and Saad A. Rahat. 2017. An IoT based fire alarming and authentication system for workhouse using Raspberry Pi 3. In Proceedings of 2017 International Conference on Electrical, Computer and Communication Engineering (ECCE), Cox's Bazar, 899-904. https://doi.org/10.1109/ECACE.2017.7913031

[14] Akash V. Bhatkule, Ulhas B. Shinde, and Shrinivas Zanwar. 2016. Home based security control system using raspberry Pi and GSM. International Journal of Innovative Research in Computer and Communication Engineering, 4, 9(September 2016), 16259-16264.

[15] Akkasit Sittisaman and Naruepon Panawong. 2019. A Development of the Internet of Things based intruder detection and security alarm system. International Journal of Engineering and Advanced Technology (IJEAT), 8, 6(August 2019), 2307 – 2312.