Smart Home Security System Using Biometric Recognition

Phan Van Vinh[1(✉)], Phan Xuan Dung[2], Pham Thuy Tien[1], Tran Thi Thuy Hang[1], Truong Hong Duc[1], and Tran Duy Nhat[2]

[1] School of Computing and Information Technology, Eastern International University, Binh Duong City, Vietnam
{vinh.phan, tien.pham.setl5, hang.tran.setl5, duc.truong.setl5}@eiu.edu.vn

[2] School of Engineering, Eastern International University, Binh Duong City, Vietnam
{dung.phan, nhat.tran}@eiu.edu.vn

Abstract. A security system is one of the most important applications of a smart home that protects our home from thieves or potential risks. However, a traditional home security system usually suffers from high costs or does not satisfy the user's needs. Therefore, in this research, we design and implement an IoT-based smart home security system, which not only protects our home from unauthorized access but also saves our life from dangerous situations. In our proposed system, biometric recognition based on the combination of fingerprint and face image is used to identify the homeowners who have permission to access the home. The main door will be opened if the input biometric image matches the one stored in the database. Otherwise, the system will raise an alarm with a doorbell and/or send a notification message to the homeowner. Besides, the system also collects environmental data in the home and notifies the homeowner in case of a dangerous situation, e.g. there was a fire or gas leak. The homeowner can monitor and control their home remotely via a friendly Web-based user interface. All activities happening in the home are recorded in a logging system for further analysis.

Keywords: Smart home · Security system · Fingerprint recognition · Face recognition

1 Introduction

Nowadays, a smart home and its applications becomes more realistic with many advantages for a more convenient and better life [1]. A security system is one of the most essential applications that need to be considered when building a smart home. People usually install a security system in their home to keep safe valuables from burglars and thieves and to protect their family safe from potential risks by fires or leakage of poisonous gas. In a traditional home security system, people usually use a key, password or security cards to unlock the door for entering their home. However, these security methods are easy to break by a thief and do not keep track the user's information of whom got accessed into the home. To tackle these problems, one

© ICST Institute for Computer Sciences, Social Informatics and Telecommunications Engineering 2021
Published by Springer Nature Switzerland AG 2021. All Rights Reserved
B. Li et al. (Eds.): IoTaaS 2020, LNICST 346, pp. 405–420, 2021.
https://doi.org/10.1007/978-3-030-67514-1_33

approach for the security system is to use human biometrics to identify a person, in which a fingerprint is widely used since the human fingerprint is unique and never changes [2]. However, it takes time to identify multiple people at the same time within closed distance. Moreover, it is not easy for human to recognize a person based on his fingerprint data. Recently, another approach in the security system is face recognition [3]. This technique usually applies computer vision and machine learning to recognize a person based on training data. At this time, the face recognition has significant improvement in accuracy and processing time with the support of IoT edge devices and deep learning technology. Recently, many deep learning algorithms with high performance of accuracy and speed have been proposed, such as Region Convolutional Neural Network (RCNN) [4], Fast RCNN [5], Multi-scale Deep Convolutional Neural Network (MS CNN) [6], Faster RCNN [7], and YOLO [8]. MobileNet-V2 [9] is the state-of-the-art object detection algorithm with the best performance in accuracy. The disadvantage of these algorithms is the hardware cost, requires more processing units than the others. However, face recognition should not be implemented alone in the security system due to the unreliability of the system. User photos or videos can be used to spoof facial recognition system. The face anti-spoofing is an open research and therefore still needs more contributions to provide a concrete solution with better performance. Some approaches take the advantages of both fingerprint and face recognition by using a combinational system [10, 11] to improve the accuracy and reliability of a security system. However, they did not provide a complete system with a friendly user interface that can be applied in practical scenarios.

In this paper, we aim at design and implement of a smart home security system with a two-stage verification process of two biometric recognitions to provide higher security level. Face recognition is used in the first phase for fast and fairly reliable verification and then fingerprint recognition can be used to improve the accuracy of the system. Besides, the proposed system can be able to provide a live video stream of the monitoring area, detect human presence and other environmental related information such as humidity, temperature, carbon monoxide (CO), poisonous gas, etc. All activities happening in the home will be kept in a logging system for further analysis. The proposed system can be controlled and managed remotely via a friendly Web-based user interface. In summary, the contribution of this paper is as follows: (1) Design and implement a smart home security system prototype and test it in practical scenarios. (2) Implement a friendly Web-based user interface to control and manage the proposed smart home security system.

In what follows, we present the principle design of the proposed approach in Sect. 2. Section 3 covers the implementation results. Finally, we make concluding remarks in Sect. 4.

2 Smart Home Security System Design

In this section, we present the main design of the proposed smart home security system, including the system architecture, the hardware and software design, and the multi-mode verification process.

2.1 System Architecture

The proposed security system architecture consists of four layers: Sensors/Actuators layer, Controller/Gateway layer, Analytic/Processing layer and User/Application layer, as shown in Fig. 1:

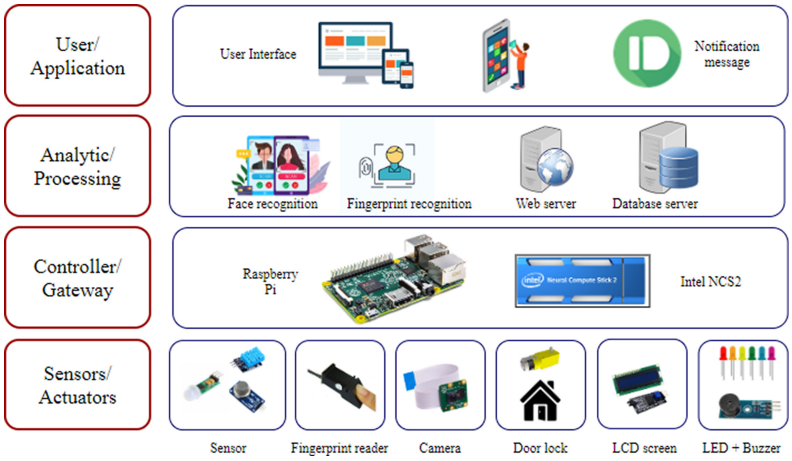


Fig. 1. The architecture of the proposed security system

- *Sensors/Actuators layer*: This layer consists of many types of input devices such as sensors, fingerprint reader, camera, and output devices such as door lock, LCD screen, LED and doorbell (buzzer). Sensors, including temperature and humidity sensor (DHT11), gas sensor (MQ2), motion detection sensor (PIR), are used to get the ambient data in the home and then send these data to the center server for further processing.
- *Controller/Gateway layer*: This is the main layer of the proposed system in which IoT edge devices are used for system control. At this time, there are several kinds of IoT edge devices with different capabilities and performance, such as Raspberry Pi family, Nvidia Jetson family, Google Edge TPU Dev Board. Moreover, Intel Neural Compute Stick 2 (NCS2), a deep learning processor on a USB stick, can be used to accelerate the performance of IoT edge device in image processing.
- *Analytic/Processing layer*: At this layer, biometric images captured from a camera and fingerprint scanner are used to detect the registered user by using face recognition and fingerprint recognition algorithms. The door will be opened or an alarm will be raised, depending on whether the detected person has the right to access or not. While the ambient data from sensors are analyzed by using a fuzzy logic control algorithm to judge whether the target environment is safe or not. All activities happening in the home will be kept in a logging system for further processing.

- *User/Application layer*: The homeowner or admin user can monitor and control all activities in the home via a web-based interface. An alert message will be displayed on the screen monitor or a notification message will be sent to mobile phone to notify the homeowner about an abnormal or dangerous situation.

Currently, there are many types of IoT-enabled devices available in the market. They can have differences in size, performance and cost. Because of project requirements, we are considering to use the Raspberry Pi model, a Linux-based high performance computer with low cost and powerful features. The specifications of some well-known Raspberry Pi models are given in Table 1.

Table 1. The specifications of some selected Raspberry Pi models

Raspberry Pi Platform	Raspberry Pi Zero Wireless	Raspberry Pi 3B	Raspberry Pi 3B+	Raspberry Pi 4B
Processor	1 GHz single-core ARM11	1.2 GHz, Quad-Core Cortex A53	1.4 GHz, Quad-Core Cortex A53	1.5 GHz, Quad-Core Cortex A72
RAM	512 MB	1 GB	1 GB	1/2/4 GB
USB	1x micro USB	4x USB 2.0	4x USB 2.0	2x USB 3.0 +2x USB 2.0
Ethernet	–	10/100 Mbps	10/100 Mbps	1 Gbps
Wi-Fi	802.11n	802.11n	2.4 GHz and 5 GHz 802.11 b/g/n/ac	2.4 GHz and 5 GHz 802.11 b/g/n/ac
Bluetooth	4.1	4.1	4.2	5.0
HDMI	Mini-HDMI	Yes	Yes	2x micro HDMI
GPIO	40 pins	40 pins	40 pins	40 pins

As we can see from Table 1, the Raspberry Pi models with small-sized, Wifi & Bluetooth enabled, and GPIO support can be one of the best options for IoT applications. However, due to the hardware limitation, the Raspberry does not perform well in some real-time image processing applications which require high speed and accuracy. To overcome this problem, one solution is to combine the Raspberry with Intel Neural Compute Stick 2 (NCS2), a deep learning processor on a USB stick (as shown in Fig. 2). With the hardware-optimized performance of the newest Intel Movidius Myriad X Vision Processing Unit (VPU), NCS2 is one of the best combinations with Raspberry Pi in image processing tasks, especially in deep learning applications.

Fig. 2. Raspberry Pi 4 with Intel Neural Compute Stick 2

2.2 Hardware Design

In this section, we describe the hardware design of the proposed security system. The Raspberry Pi model is used as a main controller to connect to all other hardware components, including sensors, LED, buzzer, DC motor, fingerprint module and Pi camera module. The following Table 2 presents all required hardware components of our proposed system.

Table 2. The hardware components of the designed system

Hardware components	Description
Raspberry Pi (3B+, 4B)	Main controller of the system
Intel neural compute stick 2	Deep learning USB device to accelerate edge devices
MQ2 Gas/Smoke sensors	Detecting LPG, Smoke, Alcohol, CH_4
DHT11 sensor	Measuring temperature and humidity
PIR sensor	Detecting a person moving in a given area
Buzzer module	Playing an alarm as doorbell
Pi camera module	Taking video or photo
DC motor	To open/close the main door
L298N motor controller	Driving DC motors to open/close a door
R305 fingerprint module	Sensor for fingerprint recognition
LED	Lights in the house
ADS1115	Analog to digital converter module
CP2102 UART	Connect Fingerprint module to Raspberry
LCD (16 x 4) with I2C module	Display information on the screen

Before going to make the printed circuit board (PCB), hardware and functionality testing need to be performed to make sure that all required components can work well in a concrete model. Figure 3 shows the connection diagram of the required components connected via a breadboard for testing purpose.

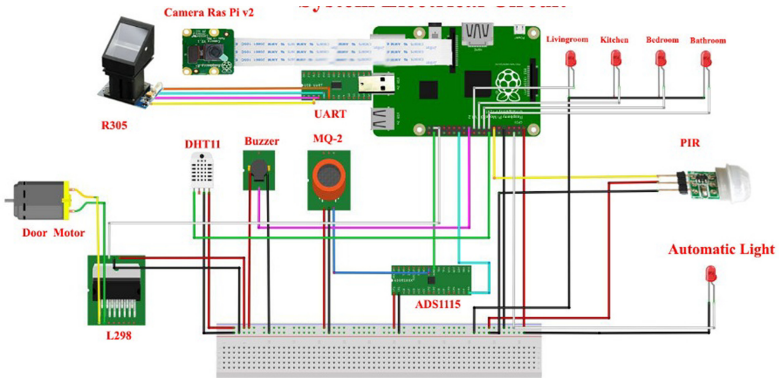


Fig. 3. The connection diagram of required parts of the proposed system

After finishing the hardware testing, we go to the hardware circuit design phase to make the printed circuit board (PCB) of the proposed system. Figure 4 illustrates the schematic diagram of all required component as follows:

- **Power block**: include the voltage reducer circuit with pulse power supply which can have maximum output of 5 V–3A from the input voltage of 5–24 V DC to supply power for other components (e.g. Raspberry Pi, sensors and actuator)
- **Motor driver block**: include Dual H-bridge driver chip L298N which can drive DC motors, control the speed and direction of each DC motor independently through PWM (Pulse Width Modulation). Input voltage is 5–30 V DC and maximum output is 2A for each bridge.
- **Controller block**: Raspberry Pi 3B+/4 is used for control system operations
- **Actuator block**: include DC motor, LCD module, buzzer and LED. The camera module attaches to the CSI interface and the R305 fingerprint module is connected to USB interface of the Raspberry

– **Sensor block**: include DHT11, PIR and MQ2 sensor for environmental data collection

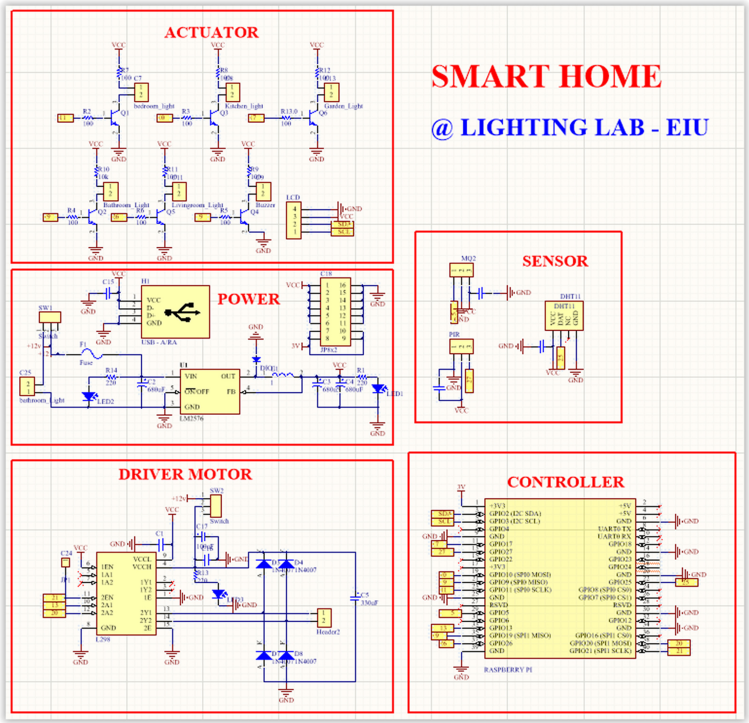


Fig. 4. The schematic view of electrical circuit diagram

The next step is to transfer the schematic diagram into a drawing of PCB (Fig. 5). The schematic will serve as a blueprint for laying out the traces and placing the components on the PCB. In addition, the PCB editing software can import all of the components, footprints, and wires into the PCB file, which will make the design process easier.

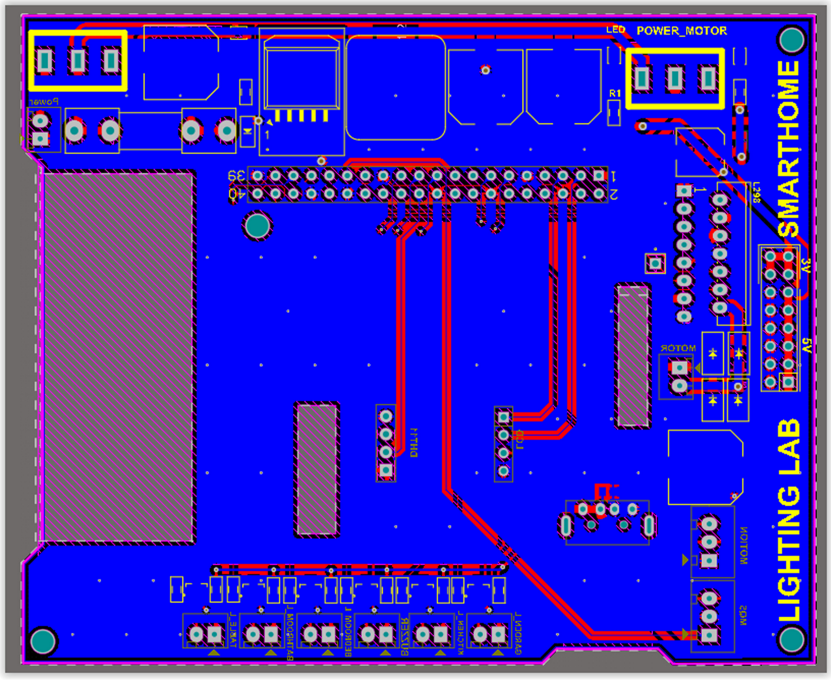
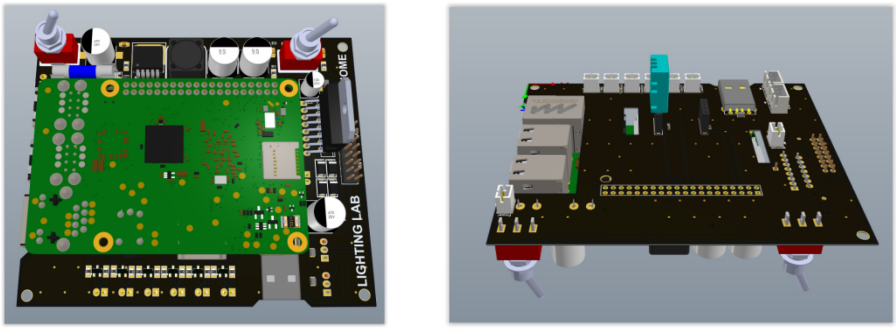


Fig. 5. The PCB layout of the designed system (2D view)

In this case, we use double layer PCB, where the entire bottom layer is covered with a copper plane connected to ground. The thickness of power wires is 35 mil while the one of data wires is 10 mil (0.254 mm). The positive traces are routed on top and connections to ground are made through holes or vias. Ground layers are good for circuits that are prone to interference, because the large area of copper acts as a shield against electromagnetic fields. They also help dissipate the heat generated by the components (Fig. 6).



(a) Front-view (b) Bottom-view

Fig. 6. The PCB layout of the designed system (3D view)

2.3 User Interface Design

In this project, we design a friendly web-based user interface for monitoring and control all operations of the proposed system efficiently. The web-based user interface is mainly developed by the PHP language and MySQL database. Figure 7 shows the structure of the designed web interface. Users must have an account to access to the system. User's registration can be done via REGISTER page. There are two types of users: *Normal users* who can only see the monitoring data and live-stream image while *Admin users* who have full privileges to manage the whole system. The functionalities of some main pages in the web application are as follows:

- CONTROL page: Users can control all activities in the home at this page, such as: switch the lights ON/OFF, enable/disable camera, video live-streaming, biometric recognition and object detection features; take photo or record video;
- DATA page: Including three subpages, display the recorded videos and photos captured from camera; visualize real-time environmental data acquisition in graph view and tabular view.

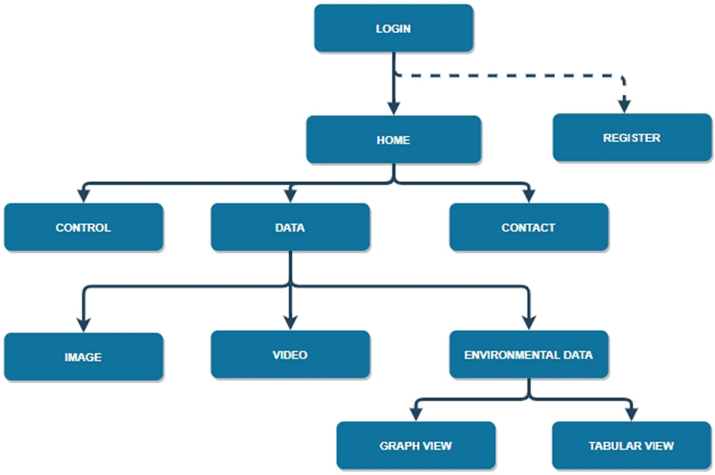


Fig. 7. The structure of the Web-based user interface of the proposed system

2.4 Multimode Verification Process

Only face recognition or fingerprint recognition may not satisfy the needs of security system in term of accuracy and processing time. Therefore, in this project, we take the advantage of using the combination of both biometric recognitions to provide a more accurate and reliable security system. There are two modes of security level for flexible usage as follows:

- *Basic security mode*: This mode provides a fast verification process with an acceptable level of security; therefore either a face recognition method or fingerprint recognition method can be used in this mode.
- *Enhanced security mode*: This mode provides high precision of security by using both biometric recognitions. In the first phase, the face recognition method is used to verify whether a person is authorized or not. This verification is not completely perfect but the achieved results are fairly good with low processing time. If the first phase is passed, the next phase of verification with fingerprint recognition is used to provide more accuracy level of the security system.

a) **The verification process with face recognition**

In this process, we apply a face recognition algorithm with a deep learning technique. Firstly, we need to build a dataset of users who are permitted to access the home. The pre-trained Caffe face detection model is used for face detector. After building the dataset, face embeddings are extracted with a pre-trained OpenFace PyTorch model. The face embeddings consist of a 128-d vector for each face in the dataset. We train a machine learning model on the set of embedding with the support of Support Vector Machines (SVM). Finally, we use the trained model with OpenCV for face recognition in the real scenarios. The detailed verification process with face recognition is illustrated in Fig. 8.

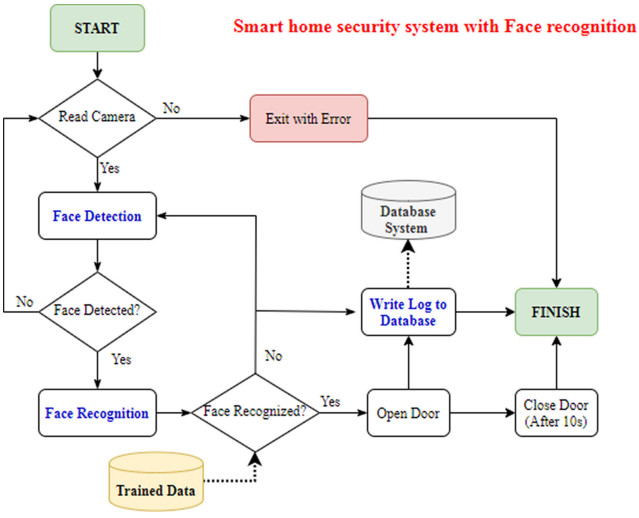


Fig. 8. The verification process with face recognition

b) **The verification process with fingerprint recognition**

The verification with fingerprint recognition in a security system is one of the most reliable and accurate methods. Firstly, fingerprint enrollment is performed to collect a dataset of user's fingerprint. The main step of this process is minutiae extraction in which the image of user's fingerprint is utilized for extracting minutiae points and

stored to create a user's fingerprint dataset. In the fingerprint recognition process, the minutiae matching technique is used to compare the input minutiae set with the one in feature set. If a fingerprint match found, the door will be opened. Otherwise, a notification message will be sent to the homeowner to notify the unauthorized access. The detailed verification process with fingerprint recognition is illustrated in Fig. 9.

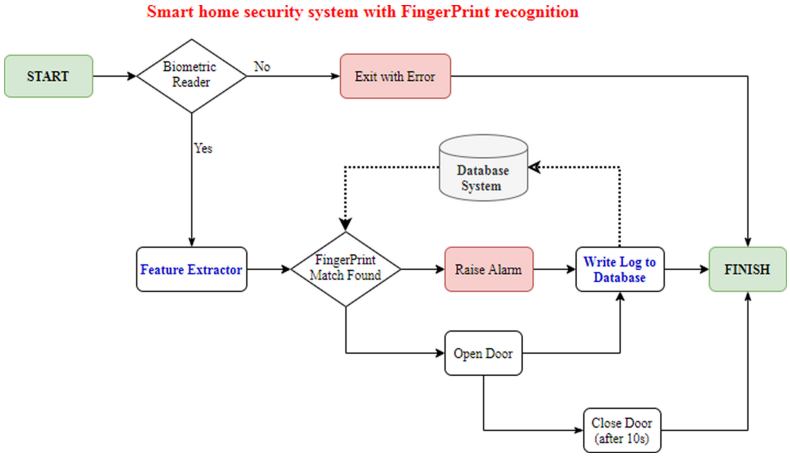


Fig. 9. The verification process with fingerprint recognition

For the notification system, we used the pushbullet application [12], a useful application to send or receive SMS and multimedia message from a computer, mobile phone or embedded device. The homeowner can receive notification messages with images captured by the attached camera. By this way, they can know what happening in their home immediately and then take the right action.

3 Implementation Results

3.1 Inference Benchmarks

In this part, to verify how the Intel NCS2 module can improve the performance of the edge devices in image processing tasks, we conduct some experiments with the inference benchmark testing. The experiments use the SSD Mobilenet-V2 network model and TensorFlow framework, frame size of 400 x 400 with object detection application.

Figure 10 shows the results from deep learning inference benchmarks of some selected Raspberry models, and laptop or desktop computers with or without the Intel NCS2 module attached. As we can see that with the Intel NCS2, the performance of all Raspberry Pi models increase rapidly, especially for the Raspberry Pi 4 with high speed of CPU and USB3.0 port supported. The Intel NCS2 also improves the performance of other hardware platforms, such as laptop or desktop computers which do not have a graphic card inside. However, the upper bound benchmark of the Intel NCS2 is about 20 FPS according to this experiment.

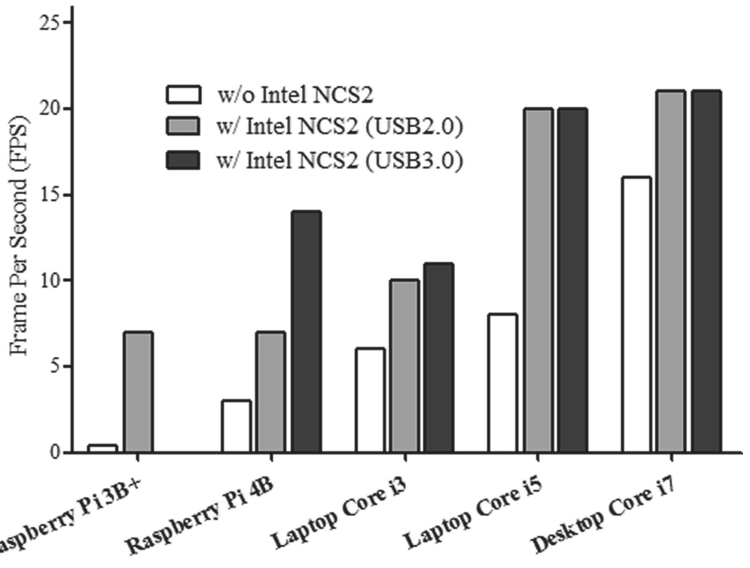


Fig. 10. The comparison of inference benchmarks of some experimental platforms

The Table 3 shows the detailed inference benchmarks and hardware utilization of some selected platforms when running the object detection algorithm. From these results, we can see that the Intel NCS2 not only improves the inference benchmark but also reduce the CPU usage when running deep neural network applications. This allows IoT-enable devices can adapt well to a variety of wide range of IoT applications.

Table 3. Inference benchmarks of some experimental platforms

Platform	FPS	Memory usage	CPU usage
Raspberry Pi Zero	0.03	N/A	N/A
Raspberry Pi 3B+	0.3–0.5	83 MB	342% (*)
Raspberry Pi 4B	3.1–3.3	78 MB	343% (*)
Laptop Core i3	5.6–6.3	103 MB	68%
Laptop Core i5	7.3–8.1	100 MB	77%
Desktop Core i7	14.1–16.3	170 MB	60%
Raspberry Pi Zero with Intel NCS2	DNR	DNR	DNR
Raspberry Pi 3B+with Intel NCS2	6.5–7.5	80 MB	106% (*)
Raspberry Pi 4B with Intel NCS2 (USB 2.0)	8.8–9.7	103 MB	148% (*)
Raspberry Pi 4B with Intel NCS2 (USB 3.0)	13.1–14.3	104 MB	167% (*)
Laptop Core i3 with Intel NCS2 (USB 2.0)	9.4–10.6	107 MB	7%
Laptop Core i3 with Intel NCS2 (USB 3.0)	9.8–10.9	106 MB	6%
Laptop Core i5 with Intel NCS2 (USB 2.0)	18.2–20.1	143 MB	7%

(*continued*)

Table 3. (*continued*)

Platform	FPS	Memory usage	CPU usage
Laptop Core i5 with Intel NCS2 (USB 3.0)	18.9–20.2	140 MB	7%
Desktop Core i7 with Intel NCS2 (USB 2.0)	19.8–21.3	196 MB	3%
Desktop Core i7 with Intel NCS2(USB 3.0)	19.5–21.2	193 MB	3%

(*): The total CPU usage of multi-core platform in a Linux-based system
DNR (did not run) or N/A: The results occurred due to limited memory capacity or hardware/software limitations.

3.2 Implementation Results

1) Smart Home Security System Prototype
First of all, we build a smart home model for testing purpose. As we can see in Fig. 11, the camera, fingerprint module, LCD module are installed in front of the home, to display user information and open/close the main door with biometric recognitions. People must do security verification before getting into the home. MQ2 – Gas/Smoke sensor is installed in a kitchen room to detect gas/smoke leakage.

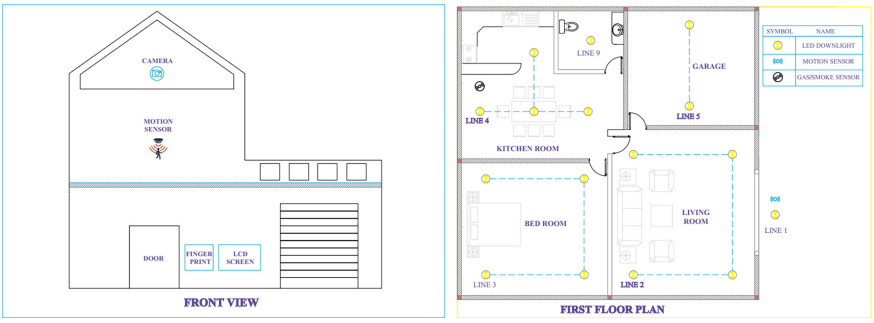


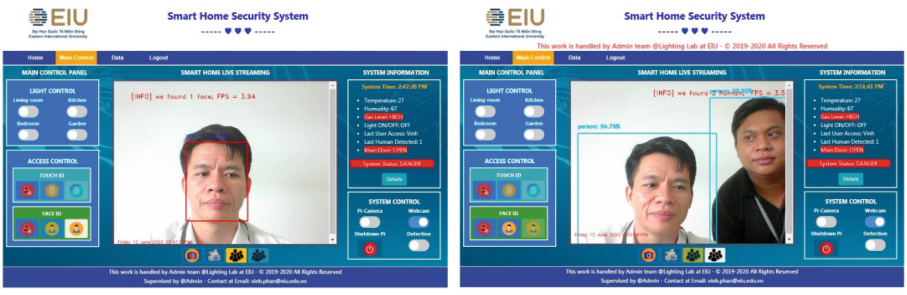
Fig. 11. The layout design of smart home prototype

Figure 12 presents the prototype of our proposed smart home model which is made of 5 mm high quality acrylic sheet with the support of a laser machine in our lab. The home model looks very nice and firmly.

Fig. 12. The prototype of our designed smart home model

2) Web-based User Interface

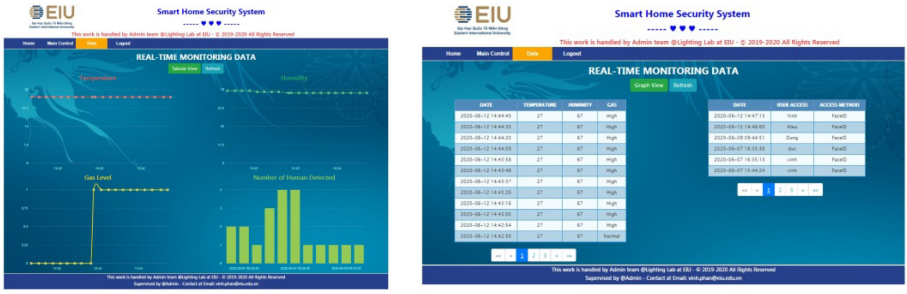
When accessing to the smart home system via the web-based user interface, the homeowner can take control of the system operation with the full rights. At the MAIN CONTROL page (Fig. 13), users can see much useful system information, such as live streaming data, environmental data, status of lights, status of door, who got accessed to the home, etc. The alarm message can be displayed when the monitoring data reach the threshold value. Besides, the homeowner can perform many kinds of system control, such as switch the lights ON/OFF, enable/disable camera, video live-streaming, enable biometric recognition and object detection features, take photos or record videos.



(a) Face recognition (b) Object detection

Fig. 13. A screenshot of security check with (a) Face recognition and (b) Object detection

At the DATA page, all collected information from the environment can be observed in real-time. Figure 14 shows the screenshot of the DATA page with the real-time monitoring data in a graph view and tabular view.



(a) Graph view of collected data (b) Tabular view of collected data

Fig. 14. A screenshot of the DATA page: monitoring data in graphs

Moreover, at the DATA page, the users can review all photos or record videos as shown in Fig. 15. In some special cases, this information is very helpful to identify what happened in the home or who got accessed to the home without permission.

Fig. 15. A screenshot of the DATA page: the recorded videos or photographs

4 Concluding Remarks

In this paper, we have proposed and implemented a smart home security system with high level of security by using the combination of both biometric recognitions. The real-time face recognition and fingerprint recognition are used to grant access for authorized people while object detection is used to detect intruders who intend to get into the home without permission. Notification messages are sent to the homeowner via

SMS or multimedia application when an intruder tries to access the home or critical situation happens in the home. The smart home control and management activities can be remotely performed via the web-based user interface. Experimental results have shown that the proposed system satisfies requirements of the smart home security system. However, the hardware design of the smart home model needs to be improved when making a realistic product.

References

1. Kumar, P., Pati, U.C.: IoT based monitoring and control of appliances for smart home. In: IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology (RTEICT), pp. 1145–1150 (2016)
2. Jose, A.C., Malekian, R., Ye, N.: Improving home automation security; integrating device fingerprinting into smart home. IEEE Access **4**, 5776–5787 (2016)
3. Pawar, S., Kithani, V., Ahuja, S., Sahu, S.: Smart home security using IoT and face recognition. In: Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), pp. 1–6 (2018)
4. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 580–587 (2014)
5. Girshick, R.: Fast R-CNN. In: IEEE International Conference on Computer Vision (ICCV), pp. 1440–1448 (2015)
6. Cai, Z., Fan, Q., Feris, R., Vasconcelos, N.: A Unified Multi-scale Deep Convolutional Neural Network for Fast Object Detection (2016)
7. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. IEEE Trans. Pattern Anal. Mach. Intell. **39**, 1137–1149 (2017)
8. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: unified, real-time object detection. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 779–788 (2016)
9. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.: MobileNetV2: inverted residuals and linear bottlenecks. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4510–4520 (2018)
10. Priyadarsini, M.J.P., et al.: Human identification using face and fingerprint. In: International Conference on Intelligent Sustainable Systems (ICISS), pp. 325–329 (2017)
11. Thakre, S., Gupta, A.K., Sharma, S.: Secure reliable multimodel biometric fingerprint and face recognition. In: International Conference on Computer Communication and Informatics (ICCCI), pp. 1–4 (2017)
12. https://www.pushbullet.com/