ORIGINAL ARTICLE

Smart house system for safety of elderly living alone based on camera and PIR sensor

Yichen Wang[1] · Yutian Wu[2] · Shuwei Zhang[1] · Harutoshi Ogai[1] · Katsumi Hirai[3] · Shigeyuki Tateno[1]

Abstract

With the improvement of human life quality, life expectancy generally increases. As a result, more and more elderly people living alone appear. Recently, the safety problems of the elderly living alone have attracted more and more attention from the public. Due to living alone, the elderly cannot be found at the first time when an accident occurs indoors or out, and the rescue time is delayed. This article proposes a way to use the speed up module to realize real-time face detection on the Raspberry Pi and optimize the processing of PIR sensor signals and write a logic system based on the camera and PIR signals to record and analyze the life of the elderly living alone and warning system to their family members.

Keywords Smart house · Elderly's safety · Face detection · PIR sensor · Raspberry Pi

1 Introduction

With the improvement of human life quality, life expectancy generally increases. As a result, more and more elderly people living alone appear. Recently, the safety problems of the elderly living alone have attracted more and more attention from the public. Due to living alone, the elderly cannot be found at the first time when an accident occurs indoors or out, and the rescue time is delayed. On the one hand, due to work and other reasons, family members such as children cannot pay attention to the activities of the elderly at home

all the time. On the other hand, existing monitoring equipment in the market also requires high price and complicated operation and installation. These reasons make busy family members worry about the elderly while working, which increases a great burden of life.

There are already many smart house monitoring devices on the market, the more famous ones are Switchbot and TP-Link series products like TL-IPC433M-4-W10 and DB52C DB52C. Among them, TL-IPC433M-4-W10 has high-definition video quality and movement recognition function and can use Wi-Fi to store video in the cloud, but it cannot recognize the identity of the people who make the movement. The TP-Link smart doorbell can only identify the people in the porch outside the door but cannot continuously monitor the residents' indoor behavior. Switchbot has many sensors, including open/close sensor, motion sensor, and security camera modules for residents to choose and assemble. But

✉ Yichen Wang
  wangyc-tony@ruri.waseda.jp

  Yutian Wu
  wuyutian@ustb.edu.cn

  Shuwei Zhang
  ZHANGSHUWEI@toki.waseda.jp

  Harutoshi Ogai
  ogai@waseda.jp

  Katsumi Hirai
  khirai@hakutsu-tech.co.jp

  Shigeyuki Tateno
  tateno@waseda.jp

[1] Graduate School of Information, Production and Systems, Waseda University, Fukuoka, Japan

[2] Key Laboratory of Knowledge Automation for Industrial Processes of Ministry of Education, School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing, China

[3] Hakutsu Technology, Fukuoka, Japan

in fact, the assembly and operation are complicated for residents and the price of a complete set of products is expensive. These reasons have led to the inapplicability of the systems sold on the platforms above to the safety monitoring of elderly people living alone.

According to the above problems, this paper designs a safety system for the elderly living alone based on low-cost sensors and Raspberry Pi, to help alarming the danger of the elderly and reduce the concerns of their relatives. We first use the acceleration module to enable the common two-stage facial recognition system to run in real time on the relatively weak Raspberry Pi. Second, we use the time interval to optimize the signal of the PIR sensor to better record the resident's actions indoors. Finally, we wrote a set of logic for analyzing the resident's activities based on the signals of the above two sensors, to help record the life of the elderly living alone and warn of possible dangers.

The structure of this article is as follows: Sect. 1 is the introduction. Section 2 details the various parts of the system. Section 3 introduces the performance of various parts of the improved system and the results of real-time use. Finally, Sect. 4 gives a summary.

2 System composition

The system proposed in this paper is mainly composed of four parts, facial recognition, motion recognition (PIR), information judgment system, and message sending system, which is shown in Fig. 1. Facial recognition and motion recognition, as the perception of the whole system, are used to accept changes in the home situation, and the information judgment system can detect the number of residents and tenants in the room, resident activity signs, etc., based on the feedback analysis of the two, and finally the information system passes through the server and sends it to the private mobile phone of the family members. The principle of each part will be explained in detail below.
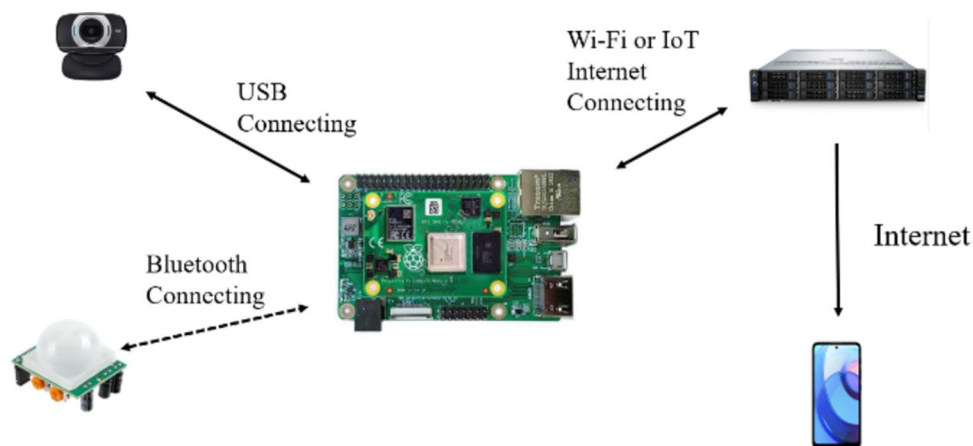
2.1 Camera facial recognition

The baseline structure of the facial recognition part mainly contains two parts, and the structure diagram is shown in Fig. 2(a).

Its tasks include finding the presence or absence of faces in incoming frames, and then comparing the recognized faces to the faces of residents stored in the database to determine whether the person recognized in the hall is a registered resident. We will briefly explain it in Sect. 2.1.1.

However, when this baseline is running on the Raspberry Pi, due to the poor computing power of low-priced hardware, the running speed is very limited and cannot meet the real-time detection requirements required by the system. Therefore, we proposed a structure using tracking accelerate module on this baseline, as shown in Fig. 2(b), so that the facial recognition module can achieve real-time detection while maintaining accuracy.

2.1.1 Related work: two-stage detection structure

In the face detection part, we choose the YuNet [1] model, which is used by the face detection library libfacedetection. The anchor-based face detector generates square anchors on feature maps of four scales. The smallest detectable face is 10×10 and the largest face is 256×256. In addition to the positioning of the face, the original author also marked 5 key points of the face (left and right eyes, nose tip, left and right mouth corners), which added the multi-task ability of key point detection to the model.

For the face recognition part, we use the SFace [2] model to extract features, which is a MobileNet [3] lightweight model supervised by a hypersphere loss function (SFace) and trained on a dataset of millions of celebrities. The dataset of millions of celebrities [4] is used to train the face recognition model after data cleaning using the
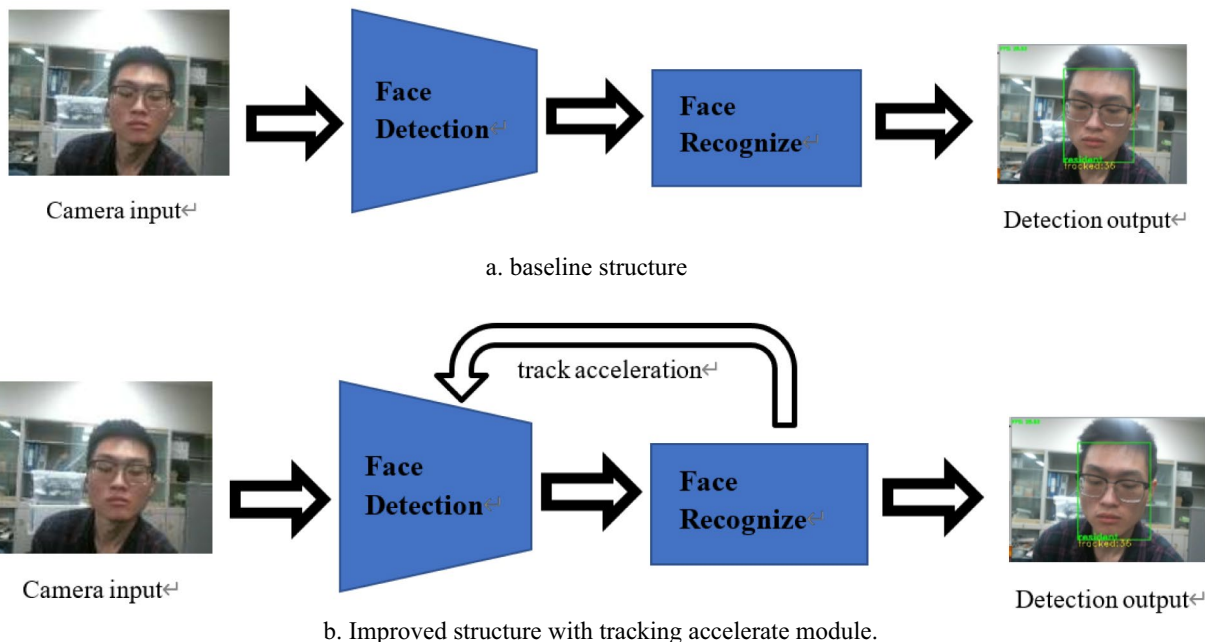
Fig. 1 System composition, including the sensor part: camera and PIR sensor, system carrier Raspberry Pi, server

a. baseline structure

b. Improved structure with tracking accelerate module.

Fig. 2 Facial recognition model structure

meta-learning-based adaptive label noise cleaning algorithm [5], which improves the performance of the face recognize model.

We first set the input frame of the camera into YuNet [1] to find out whether there is a human face and its existing area, and then use the detected YuNet [1] output face area as input and using SFace [2] extracts the features and matches them with the resident's face stored in the system. Face feature comparison calculates the distance between features to determine whether different face images belong to the same identity, to identify whether the person in the current video frame is a registered ID resident.

2.1.2 Improvement: speed up module

From the above, we have trained two models to form a two-stage face detecting and recognizing structure with the related work, but it still cannot be used in real-life condition. Even these several modules can be called from OpenCV directly, the structure cannot achieve the running speed of real-time (around 30fps) deployment on the Raspberry Pi. The size of YuNet [1] detected as the first step is only 337 K, and when the input image size is 160×120, it can reach 160FPS on the Raspberry Pi 4B. Meanwhile SFace [2] still needs a lot of computing resources. When the input size is 112×112, SFace's performance on the Raspberry Pi is only 10FPS, which cannot achieve the real-time requirements.

To make the entire detection meet the real-time requirements, we added a tracking acceleration module to reduce the calculation time of SFace [2], as shown in Fig. 2(b).

The return value of YuNet [1] is a quaternion array (x, y, w, h), where (x, y) is the upper left anchor point of the bounding box of the detecting face, and (w, h) is the length and width of the bounding box. The example of YuNet [1] output parameters is shown in Fig. 3.

From this, we can calculate the center position of the bounding box of the adjacent input two frames of faces:

$$(c_x, c_y) = \left(x + \frac{w}{2}, y - \frac{h}{2} \right) \tag{1}$$
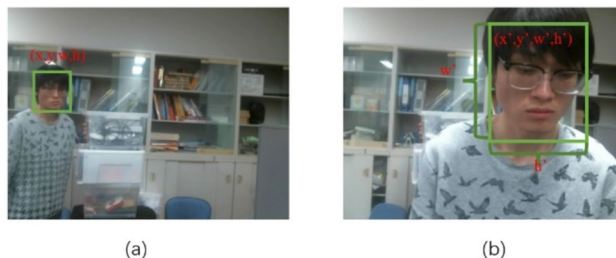

(a)

(b)

Fig. 3 YuNet [1] output parameters used in tracking accelerate model, **a** shows the time t output and **b** shows the time t' output as example

Further, we can calculate the absolute distance square in the bounding box of the face at time t and the bounding box at time $t + 1$, and when

$$F_{distance} = \left(c_x^t - c_x^{t+1}\right)^2 + \left(c_y^t - c_y^{t+1}\right) < SF_d^2, \tag{2}$$

we consider that the face bounding boxes in these two frames belong to the same face. When there are multiple faces in two adjacent input frames, and the bounding boxes of different faces are relatively close, we regard the bounding box with a smaller $F_{distance}$ in the next frame as the same face recognition. Compared with SFace's large amount of calculation per frame, the calculation of this tracking module is so small that it will not have a significant impact on the number of frames (only fluctuating at $1 \sim 2$FPS).

2.2 Sensing with PIR sensor

We use a cheap sensor, HC-SR501 PIR Sensor, as a detection sensor for residents' indoor movement, HC-SR501 is based on infrared technology, automatic control module, using Germany imported LHI778 probe design, high sensitivity, high reliability, ultra-low-voltage operating mode, widely used in various auto-sensing electrical equipment, especially for battery-powered automatic controlled products.

2.2.1 Principle and assembly with Arduino

PIR sensors are specifically designed to detect levels of infrared radiation. At temperatures above absolute zero (0 Kelvin/-273.15 °C), the human body releases heat energy in the form of infrared radiation. The hotter an object is, the more radiation it emits. Therefore, when a human resident is active in the house, the PIR sensor will change the reading due to the proximity of the human body. We use this output to detect the resident's actions.

A PIR sensor consists of two main parts (Fig. 4):

1. A pyroelectric sensor, which you can see in the image below as a round metal with a rectangular crystal in the center.
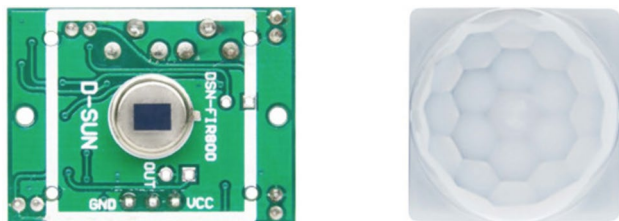
2. A special lens called a fresnel lens, which focuses the infrared signals on the pyroelectric sensor.

The principle of PIR conversion signal is not the focus of this article. You can check in the PIR sensor website [6].

2.2.2 Processing PIR sensor signals

When the PIR sensor detects the motion of the object, there are two trigger modes to determine its output, which are:

One-trigger mode: sustained motion will result in a one trigger.

Multiple-trigger mode: constant movement will cause a series of triggers.

In one-trigger mode, the output goes HIGH as soon as motion is detected and remains HIGH for a period determined by the time-delay potentiometer. Further detection is blocked until the output returns to LOW at the end of the time delay. If there is still motion, the output will go HIGH again, so Movement 3 is completely ignored. In model (b), the output goes HIGH as soon as motion is detected and remains HIGH for a period determined by the time-delay potentiometer. Unlike single trigger mode, further detection is not blocked, so the time delay is reset each time motion is detected. Once the motion stops, the output returns to LOW only after a time delay [4].

The board comes with a berg jumper (some modules have a solder bridge jumper) allowing you to choose one of two modes: L and H, details shown in Fig. 5.

But in actual use, whichever mode of signal output we use, the signal of the PIR sensor will still be interrupted to varying degrees due to the length of time the resident stops.
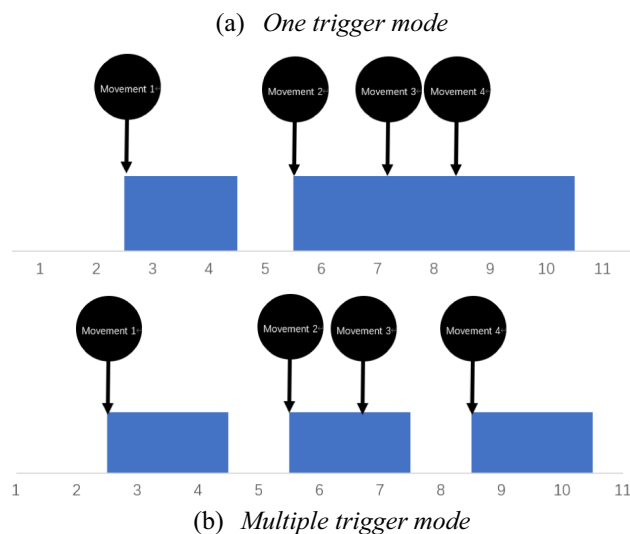

Fig. 5 Two trigger modes of PIR sensor output, the x-axis represents the time axis, the signal trigger time is assumed to be 2 s, a black circle stands for an independent movement

Fig. 4 PIR sensor composition

Therefore, we set up a default time while receiving the signal on the Raspberry Pi. Taking the test site as the toilet as an example, we set the default time for the resident to go to the toilet as a maximum of 30 min. Within 30 min of the start of the signal, we will default to the same movement for subsequent simple signals. Since the purpose of PIR is only to detect whether the resident has not moved for a long time when he is alone at home, the error of the number of actions is negligible in signal processing.

2.3 Judgment system

When the system on the Raspberry Pi receives the real-time information from the camera and PIR sensor, it will make a judgment based on the current status of the registrants (resident, family, guest) in the system and make a corresponding status change of the registrant. This section mainly describes how the system judges it.

It should be emphasized that the primary task of our system is not to monitor the elderly people living completely and accurately alone at home, but to analyze potential dangers that require early warning through sensor information. The camera is only used in the hall area, focusing on people's face information to record their identify and in and out status.

Meanwhile, the PIR sensor is set in places where there must be activity for people in a period. PIR sensor can cover a small space, so places such as kitchens, toilets, and balconies are suitable for placing it to detect the resident's movements. The toilet above is just an example. In actual application scenarios, it will be updated in additional rooms according to the conditions of the house.

It should also be emphasized that the PIR sensor will only be used by the system to predict and record the user's behavior when the elderly user is at home alone. This is because the main purpose of the PIR sensor is to predict the situation when an elderly person accidentally loses their mobility when they are alone at home, so as to give early warning of potential danger to their family members. When there are multiple people in the room, visitors can help the user better than our system in such accidents.

2.3.1 Installation of camera

The camera is set on the right side of the resident's exit direction Fig. 6(a).

The camera we use is Logitech C615N, which is customized for convenience and can be modified at any time as needed. The viewing angle is 78 degrees, and the effective distance is $1 \sim 5$ m, so it can actually reach the house entrance 6 m, which shown in Fig. 6(b).

Because the system has a higher demand for identity confirmation of people entering the home, we appropriately tilt the camera toward the door to obtain more and more accurate facial information of people entering the home. In the test, $5 \sim 10°$ can better verify the identity of the person entering the door while ensuring that the facial information of the person leaving the door is detected.

The walking speed of the elderly we assumed is 1 m/s for normal people, so the walking time during this area is $2 \sim 6$ s, or $60 \sim 180$ frames. Since the camera is placed in
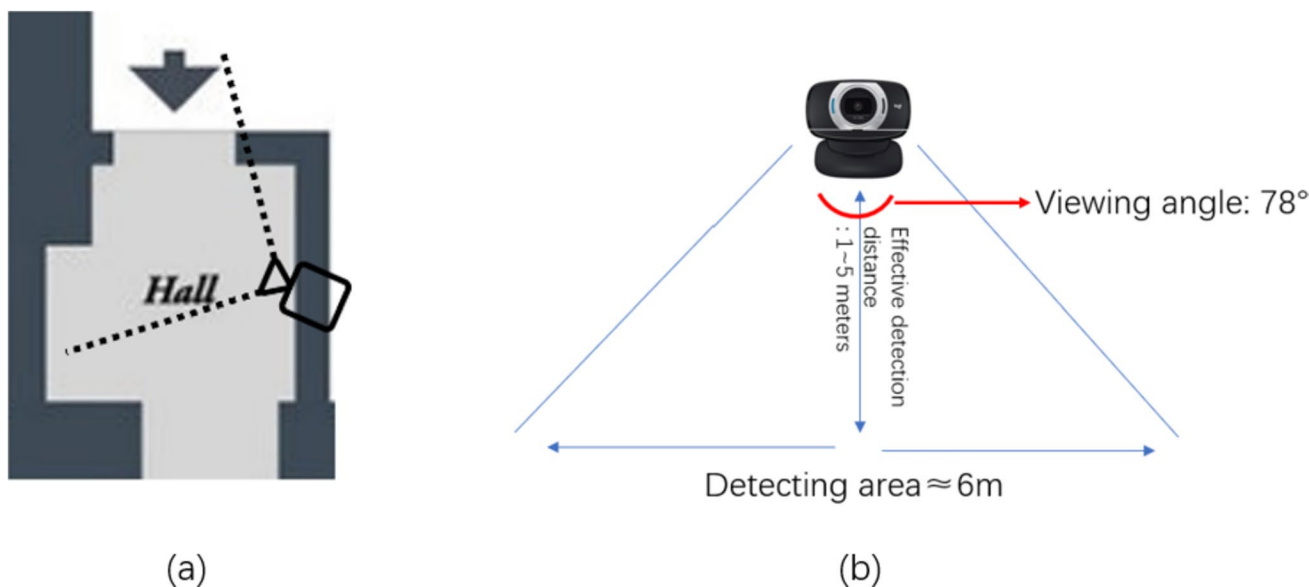

Fig. 6 A Schematic diagram of setting the direction of the camera, the dotted line range is the view field of the camera. **B** Detailed data regarding the camera's angle of view, effective detection distance, and detection area

the indoor hall area, we default that when the user enters the detection area, there is sufficient daylight or sufficient lighting in the hall.

2.3.2 Processing of each sensor signal

This section mainly explains the basis for judging whether the registrant is going out or not.

We assume that when the registered resident's face appears in the frame of the camera, the output parameters are $(X_{begin}, Y_{begin}, W_{begin}, H_{begin})$, and the last output parameter when resident leaving the camera's field of view is $(X_{end}, Y_{end}, W_{end}, H_{end})$, then its displacement value in the horizontal direction is:

$$D_{level} = \left(X_{begin} + \frac{W_{begin}}{2} \right) - \left(X_{end} + \frac{W_{end}}{2} \right) \qquad (3)$$

When the value of $D_{level} < 0$, it indicates that the detected person moves from the right to the left of the camera, which means entering the house. Otherwise, it means going out.

2.3.3 Processing of PIR signal

PIR sensor needs a minute or so to initialize. During initializing time, it will output 0–3 times. One minute later it comes into standby. Therefore, when the PIR sensor is just started in 1 min (Bluetooth connection), the system will ignore the signal of the PIR sensor.

2.3.4 The messages generation

The messages generated by the system according to the resident status and acquisition are shown in Table 1. Among it, resident is the person who uses the system, and visitors are the family members and guests of the resident. The corresponding situation of the message number is as follows:

n0_000 means normal situation, resident has movement at home.

n1_000 means normal situation, visitors come home.

u0_000 and u1_000 stand for resident changes their status: coming home or going out.

a0 is for resident at home when something needs to alarm.

a1 is for resident outside when something needs to alarm.

The description part describes the reporting type of the information label, the resident's situation (at home or not), and a brief description of the event, among which:

Timing report is report when normal things happen.

Emergency report is warning for resident's family members.

Status (Before) is the resident's situation before the event happens. Detecting Results is what the camera and PIR sensor detected. The two combined determine the type of event that occurs. Status (After) is the status change for the resident and others after the event.

The working StateFlow of the judging system is shown in Appendix A (Fig. 11). When the process starts, the system will first check the resident's status, and then judge whether there is a registered person (resident, family member and

Table 1 System messages and corresponding description

Message number	Description (kind of report, resident at home or not, details of what happen)	Status (before)	Detecting results	Status (After)
n0_000	Timing report, presence, no abnormal situation	Resident: at home no other people	PIR sensor has signal	No change
n1_000	Timing report, presence, has visitor, no abnormality	Resident: at home no other people	Camera detects family or guest	Visitor's status changes to at home
u0_000	Timing report, presence, status changing	Resident: outside	Camera detects resident	Resident's status changes to at home
u1_000	Timing report, going out, status changing	Resident: at home	Camera detects resident	Resident's status changes to outside
a0_001	Emergency report, presence, abnormal situation: resident no movement	Resident: at home	PIR sensor has no signal for a long time	No change
a0_003	Emergency report, presence, abnormal situation: strangers break in	Resident: at home	Camera detects stranger	No change
a1_002	Emergency report, going out, abnormal situation: resident stayed out for too long	Resident: outside	Camera does not detect resident for a long time	No change
a1_003	Emergency report, going out, abnormal situation: strangers break in	Resident: outside	Camera detects stranger	No change

guest) whose status needs to be changed according to the information detected by the camera. Then make a timing report according to the situation. When the resident has not been detected by the PIR sensor for 30 min at home or has not returned after going out for 3 h, or someone unregistered break into the home, the system will issue a relevant emergency report. The speed of the message generation circulation is same with the camera fps (about 30 rounds per second).

2.4 Message server and smart phone

Different from conventional systems on the market, this system is built based on Raspberry Pi. It can not only transmit data through Wi-Fi and the server, but also through the SIM card on the Raspberry Pi. When the Wi-Fi signal is unavailable, the IoT network sends out the messages. When the server receives the messages from the Raspberry Pi, it will send an alarm to the mobile app of the relatives of the corresponding resident according to the system number.

3 Experiment and result

This section will introduce the general average performance of the final running system under long-running conditions and the comparison of the modified model network in each case, including the running speed, output signal value, and comparison with related products.

3.1 Face recognition comparison

Table 2 shows the performance of state-of-the-art face detection and recognition models on Raspberry Pi platforms and the performance of our proposed model.

In addition, the FPS will also be affected depending on the output result. This is due to the error caused by the distance of facial recognition. When the face recognized in YuNet [1] is relatively far away, SFace [2] cannot extract enough facial features to match with the facial information registered in the library and will identify the resident as an unknown person. When the resident moves closer to the camera, SFace [2] updates the features to match them.

Table 2 Performance of different models

Method	Face detection	Face recognition	FPS
①	HOG [7]	ResNet [9]	< 1
②	SSD [8]	ResNet [9]	1.5
③	YuNet [1]	SFace [2]	5
④	Method③ + speed up module (our proposed)		27

Therefore, when an unknown person is identified, the system designed in this paper will not use tracking acceleration. As a result, the Fps of our proposed method detecting unknown person will keep the same with method③ at 5.

Apart from these, robust to face rotate is also improved in our method. As shown in Fig. 7, it is the recognition result of the method③ and our proposed method for the face of different angle.

It can be easily seen from the figure that compared with the baseline, our algorithm can identify user's face from more angles.

The reason for this result is that the recognize operation will be performed for each recognized face in the method③, but the facial features of the side face are not enough for SFace [2] to recognize whether it is the resident. And our algorithm uses the tracking module, so that the resident can be recognized even when the face changes from the front
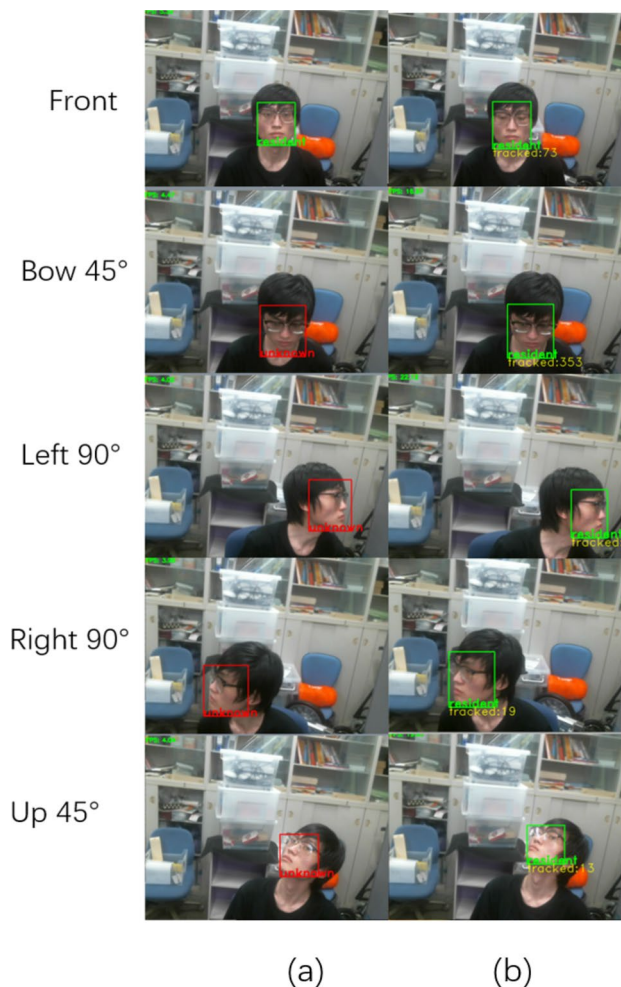

Fig. 7 Testing in the same test video, the two algorithms identify the user's face from different angles. **a** Represents the results of method ③, **b** represents our method

Table 3 Accuracy of the two methods

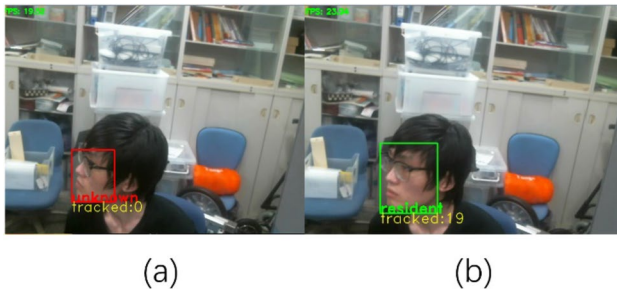| Method	Structure	Accuracy %
③	YuNet [1] + SFace [2]	87.81
④	Method③ + speed up module (our proposed)	94.15

(a)   (b)

Fig. 8 When at time (**A**), the user's face has just been recognized, but there are not enough features to be identified; at time (**B**), when the user's face can be identified, the previously detected face is updated simultaneously

face to the side face, to achieve the effect of increasing robust to rotate.

After testing the two methods in our test set, the accuracy for resident face recognition is shown in Table 3. The accuracy rate is calculated as the number of correctly identified faces divided by the total number of faces that appeared.

As shown in Table 3, the reason why our method has a higher accuracy rate than the baseline in facial identification is also shown in Fig. 8.

Even when the user's face first appears, facial information cannot be correctly identified due to insufficient feature points. When there are enough features to identify the identity later, the tracking module will supplement the number of frames of the previous unknown face, increasing the accuracy of identification. As shown in Fig. 8 (b), the number of tracking frames corresponding to the user's face inherits the previous unknown face's 19.

3.2 PIR sensor signal

In this section, we will show the difference between the signal of the PIR sensor after adding the time interval and the signal of the original basic setting. For the convenience of comparison, in the result display, we set the default time to 15 s. The result is shown in Fig. 9(a).

In the test, we applied actions at 1 s, 24 s, 41 s, and 57 s respectively. We have completed the signal initialization of the PIR sensor before the start of the test. The 1 s on the X-axis in the figure is the first second of the first PIR sensor at the start of the test. Different peak lines represent the PIR

sensor producing different signals for the same effect on a person under different triggering modes.

When the PIR sensor detects the signal, you will see the change of the peak in the graph. It is not difficult to see from the figure whether it is one-trigger mode or multiple-trigger mode, some erroneous noise will be generated. When the same action is detected by the PIR sensor, based on its two original trigger modes that are too sensitive to the action, it will react to the same action 2 to 3 times to change the signal and notify the system that the user's action has been detected. This will lead to an increase in the frequency of sending information, increase the burden on the system and server, and reduce the system's judgment on the credibility of the PIR sensor signal.

When the default time (15 s here) is added to the signal processing, the signal will only be triggered once within 15 s of the detected action, greatly reducing the frequency of false alarms.

In addition, we also attached the PIR sensor signal record for a long time during the actual use test. During the 12 h that were recorded, we performed activities within the detection range of the PIR sensor at 4 min, 134 min, 207 min, 310 min, 461 min, and 699 min. As can be seen from Fig. 9(b), the signal of the PIR sensor remains stable for a long time, and in the mode with default time, there is no multiple fluctuations for a single activity.

So, the addition of default time is helpful for signal processing of PIR sensors.

3.3 Results displayed on the Raspberry Pi

In this section, we will show the results that will appear on the output of the Raspberry Pi when there is a situation that needs to be reported.

As shown in Fig. 10, each piece of information sent by our system to the server is separated by a dotted line. Between the two dotted lines is a complete message, which includes the following details:

The first line records the signal of the current PIR sensor. 1 means someone is detected within the range, 0 means there is no activity within the range.

The second line records the current time and the specific content of the event.

The third to fourth lines are the current system number, the message number sent to the server, the time sent to the server, and the event number.

The last line is the current indoor personnel situation.

3.4 Related product comparison

In this section, we compare our system at the current stage with the three products mentioned in the introduction and analyze the advantages and disadvantages of each product.

PIR sensor signal

□ one trigger   □ Multiple trigger   □ default time
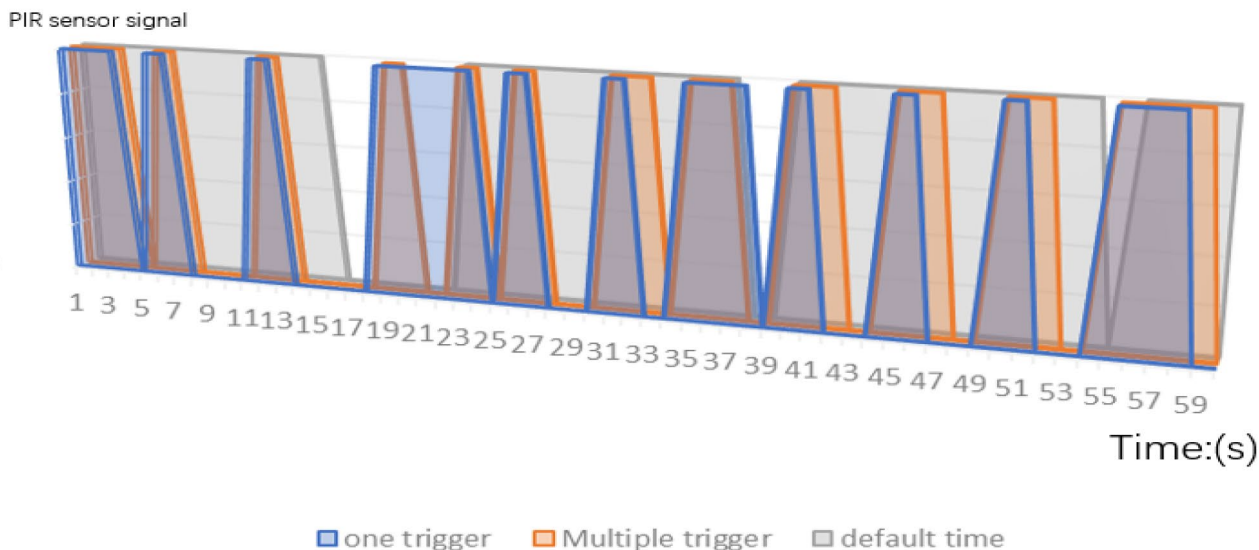
(a) When the default time is set to 15s, the waveforms generated by the PIR sensor according to the signal in different modes in 1 minute

PIR sensor signal with default time lasting for 12hours
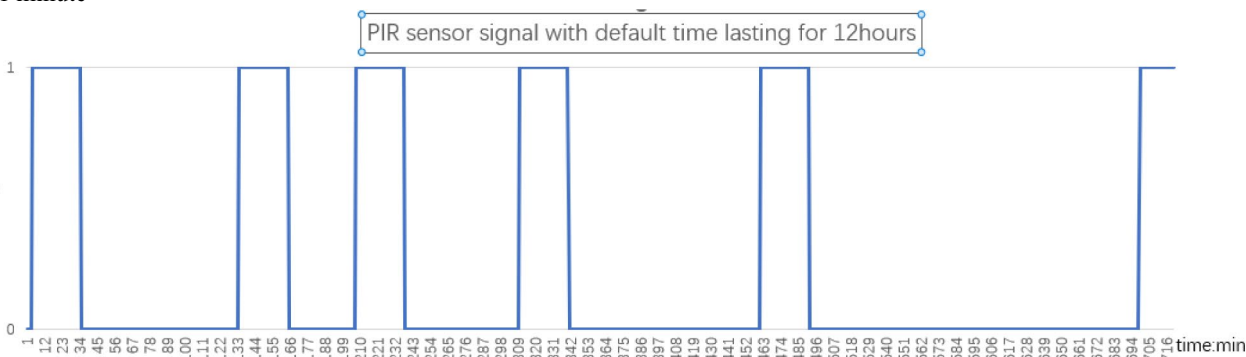
(b) In actual use, the signal fluctuation of the PIR sensor within 12 hours

Fig. 9 Signal feedback of PIR sensor: when resident's movement is detected by PIR sensor, a crest will be generated. In **a**, blue represents the result of the one-trigger mode, orange represents the multiple-trigger mode, and gray is the feedback result after we added the default time. The x-axis represents time(s). In **b**, blue represents the fluctuation of the PIR sensor signal in actual use of our system The x-axis represents time(min) (color figure online)

E': '2022-11-17 23:18:18', 'REPORT_TYPE': 's0', 'GROUND_FOR_ALERT': '000'}
the resident is in, family(s): 0 at home, guest(s): 0 at home.

PIR sensor sent signal 1 at 2022-11-17 23:19:41 ------------
At 2022-11-17 23:19:41 Resident went to toilet.
AI Terminal sent message: {'TERMINAL_ID': '0001', 'MESSAGE_NO': '403', 'DATA_TIM
E': '2022-11-17 23:19:41', 'REPORT_TYPE': 'n0', 'GROUND_FOR_ALERT': '000'}
the resident is in, family(s): 0 at home, guest(s): 0 at home.

Fig. 10 Analysis results displayed on the Raspberry Pi

Table 4 Products' comparison

Product name	Image size	Detection content	Installation	Price (JPY)
TL-IPC433M-4-W10	640	Infrared humanoid detection within 5 m, mobile detection, no indoor detection	Wireless interface	4500
DB52C	2 k	3-m humanoid detection, no indoor detection	Wireless interface	3700
Switchbot	1080	Multi-sensor free choice	Free combination, complex	50,000 (basically meet requirement)
Our system	640	3 m visit record, indoor activity record, smart alarm	Bluetooth, USB connection, wireless interface	35,000 (among them, the demand for cameras is only ¥3000 webcams.)

Table 4 lists the differences between the comparison product and our system.

The first two rows represent the monitoring system and visual doorbell system that are leading sales in e-commerce platforms. Compared with our system, TL-IPC433M-4-W10 can only detect whether there is someone within 5 m outside the door but cannot identify the person's identity. Although the DB52C can identify visitors, it requires higher resolution images for identification, which requires higher camera costs. At the same time, like the TL-IPC433M-4-W10, it can only detect outside the door, and cannot detect the things that may exist indoors. Warning of danger. Both of them also need to be connected to indoor Wi-Fi to interact with the mobile phone signal, but our product can rely on the SIM card provided on the Raspberry Pi to independently interact with the server. In addition, only comparing the camera part, since our system has lower image size requirements, we can use lower cost cameras to save the whole system costs.

The third row is the Switchbot series of products. Although the functions are very complete, each part of this series of products is separated and sold independently. As a result, users need to select the combination modules themselves and connect them to their mobile phones one by one, which increases the number of elderly users, difficulty of use, and cost of sales. In comparison, our products have an advantage in price Fig. 11.

4 Conclusion

As a conclusion, to deal with the increasingly prominent safety issues of elderly people living alone, this paper proposes an early warning system based on Raspberry Pi with two sensors: camera and PIR sensor. First, the system can detect and record the identity of people who visited the house through the camera installed in the hall. We propose an improved two-stage structure that uses tracking accelerate module to speed up face identification to achieve real-time identify on the Raspberry Pi. This allows the system to use cheaper cameras with slightly lower image quality, thereby reducing system costs. Second, the system proposed in this paper uses the PIR sensor to detect the user's necessary behaviors to predict the potential danger of the elderly living alone being unable to move due to sudden accidents. It uses the default time to reduce the frequency of the PIR sensor reporting the same action to reduce the false alarm rate, and improve the credibility of the alert to the user's family. Finally, this article proposes a judgment logic based on time and two sensor signals to achieve in and out records for the elderly living alone and early warning of potential dangers. Ultimately, we came up with a system that is cheaper and easier to install than similar products in the market.

However, there are still some parts that need to be improved when the system is deployed. At present, the angle of camera deployment has a great impact on the detection results of resident access. But when the detection object of the image is multiple people, we still need a more accurate and less time-consuming tracking algorithm to maintain the accuracy of recognition. At the same time, the PIR sensor is based on Bluetooth communication and Raspberry Pi for data transmission. The communication quality of the Bluetooth channel (distance, interference from obstacles between terminals) is also a part that needs to be considered and guaranteed in the future.

Appendix

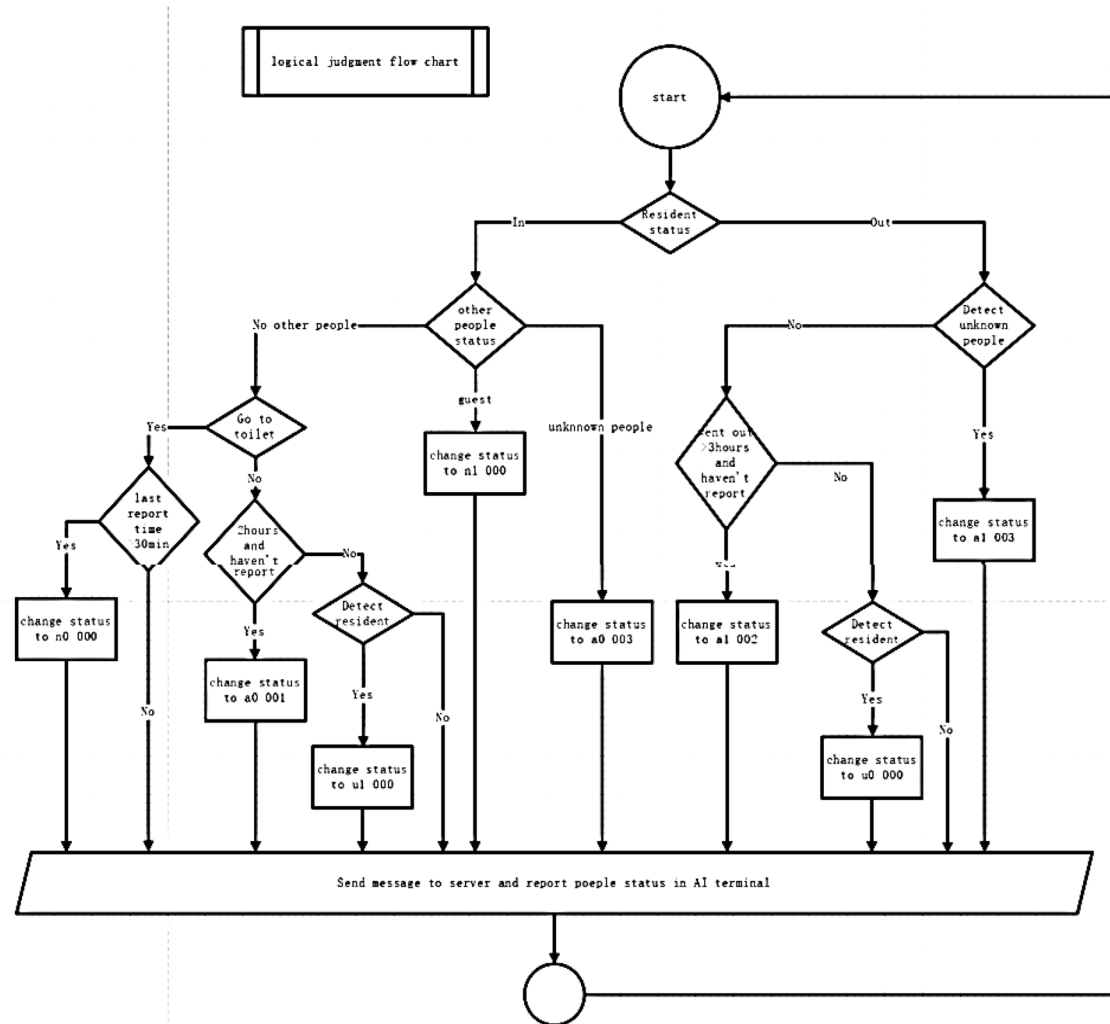See Fig. 11.

Judging system flow chart

Fig. 11 Judging system flow chart

References

1. Feng Y et al (2021) Detect faces efficiently: a survey and evaluations. IEEE Trans Biomet Behav Ident Sci 4(1):1–18
2. Zhong Y et al (2021) SFace: Sigmoid-constrained hypersphere loss for robust face recognition. IEEE Trans Image Process 30:2587–2598
3. Howard AG, et al ((2017) "Mobilenets: Efficient convolutional neural networks for mobile vision applications." arXiv preprint arXiv:1704.04861
4. Guo Y et al (2016) "Ms-celeb-1m: A dataset and benchmark for large-scale face recognition". Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part III 14. Springer International Publishing
5. Zhang Y-F et al (2022) Focal and efficient IOU loss for accurate bounding box regression. Neurocomputing 506:146–157
6. PIR sensor tutorial website, https://lastminuteengineers.com/PIR-sensor-arduino-tutorial/"

7. Mallick S (2016) Histogram of oriented gradients explained using opencv. Retrieved from LearnOpenCV. com/histogram-of-oriented-gradients

8. Liu W, Anguelov D, Erhan D et al (2016) Ssd: Single shot multibox detector[C] Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14. Springer International Publishing: 21-37

9. Targ S, Almeida D, Lyman K. Resnet in resnet: Generalizing residual architectures[J]. arXiv preprint arXiv:1603.08029, 2016.