Progressive Report upon Face-Traceable Fever Detection System

Z5286111(Wenqian Zhang), Z5325156(Zeal Liang), Z5379786(Jiacheng Wei), Z5286095(Minxin Shi)

1. Introduction:

Our project focuses on developing a low-power portable system for fever detection. In our initial proposal, we planned to use a Raspberry Pi as the main system and a thermal camera and an RGB camera for temperature measurement and face detection. In particular, we intended to employ a micro-analogue servo with a small driving force to aid in face tracking.

2. Plan in Changes:

As the development processes, in order to improve the tracking for faces, we extended the number of servos to two with stronger driving forces to provide 2 DoF of tracking. We have also introduced a new compact screen, SSD1315, to help visualise inputs users can give, making the system more user-friendly.

3. Challenges in Software

a. Challenge 1: Thermal Camera Tuning and Face Detection

In the early stage of development, when tuning the thermal sensor, we found the pixel-based temperature measurement was inaccurate if directly averaged from the face readings. We found that the inaccurate readings were due to the ineffective landmark points we captured. The model we used captures 468 landmark points from faces, where some of the points located on the edges are from the ambient environment, which is behind human faces.

Our final solution was that we first remove those edge landmark points. Then, we average some sets of 5-highest data from multiple frames to increase the reliability. This solution has been tested and verified to have stable outcomes throughout our development experiments.

b. Challenge 2: Aligning thermal and RGB images

- First attempt:

We extract the feature points in both images by ORB algorithm and then match the key points using BFMatcher provided by the OpenCV library. The best 10 matches were selected to compute the perspective matrix. But the problem is that the faces are not of one type due to the nature of details in thermal graphics and rgb. So the extracted features can't match.

- Second attempt:

We manually select a matrix between two images by using the selectROI function in OpenCV. After that, a perspective transformation matrix is returned. After testing, the alignment is very good after applying this matrix with a warp perspective.

c. Challenge 3: Correcting temperature offsets based on face distance

Without adding any offset, the acquired face temperature decreases as the distance gets larger.

- First attempt:

Obtain the maximum and minimum x-values for all the key points generated by face recognition. The width of the face is calculated from these two values. The temperature offset is then calculated as an empirical value based on the ratio of this width to the width of the whole image. Add this offset to the raw temperature of the thermal data to get the result. As a result, the calculated temperature is relatively stable only in the range of 50-100cm, while the error is larger for longer distances.

The formula is as follows (a and b are experience values, and will continue to be revised subsequently):

$$Temp_{final} = Temp_{avg} + a + Ratio * b, \ where\ a = 3,\ b = 0.0006$$

- Second attempt:

After aligning the thermal and rgb images, there is the added benefit of being able to align the depth image as well in a similar way. After having the depth data for the face, we were able to maintain relatively stable temperature data for distances ranging from 20-200cm. The formula is as follows (a, b and k are experience values, and will continue to be revised subsequently):

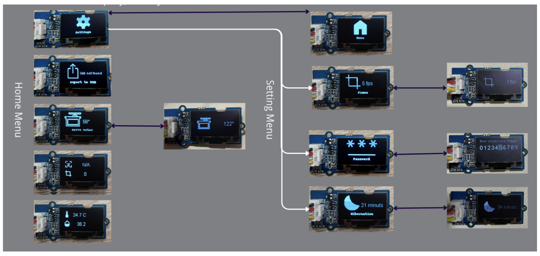
$$Temp_{final} = Temp_{avg} + a + Depth_{avg} * (b + k * (T_{env} - T_{ref})),\ where\ a = 3,\ b = 0.02,\ \{k \mid 0.001 \le k \le 0.01\}$$

4. Hardware Progress

2D gimbal:

Completed the 2D gimbal driver, realised the x-axis range, 0°-180° degrees, y-axis range, 40°~150° degrees range of rapid movement, using two servos for the implementation, due to the Raspberry Pi's voltage at the same time to supply two servos and other hardware will lead to insufficient voltage can not be turned on, so here we use a 2 * 4 * AA battery pack adds extra power to the servos.

Human-Machine Interaction:



We've used a screen to enhance the information display, for interaction using a Rotary Position Sensor and Button. Our storyboard includes the following pages and features:
On the home menu, we have 5 pages which are Setting, Export to USB, Servo Angle, Face and Environmental Temperature.

- Export to USB: By clicking the button, users can export images, timestamps, and temperatures of people with potential fever. If a USB storage device is connected to the Raspberry Pi, the data will be written to it. Otherwise, an error page will be displayed.
- Servo Angle: By clicking the button, Users can adjust the current gimbal's x-axis rotation position (0-180) using the Rotary Position Sensor. Clicking the button confirms the setting and automatically returns to the previous menu (home menu).
- Face Temperature: shows the current face temperature and the number of frames tracked.
- Environmental Temperature: displays current environmental temperature and humidity.

Click the button on the setting page to switch to the settings menu. There are Frame, Password and Hibernation in the menu.

- Frame: By clicking the button, users can maximise the number of frames for a single person. Confirming the setting will automatically return them to the setting menu.
- Password: Upon clicking the button, users need to enter the old password using the Rotary Position Sensor and button. They will then be prompted to enter the new password twice in the same manner. After completion, they'll return to the previous menu.
- Hibernation: Set the machine to enter sleep mode after a specified period of inactivity.
- Home Page: Clicking on the homepage will take users back to the home menu.

Physical Alarm:

If fever is detected, the buzzer will start to sound and the screen will start to flash the warning, click the button and then enter the password to cancel the warning, as shown in the picture.

5. Future Plan

Pending Plans:

- Alarm Email: If the temperature exceeds the threshold, the camera will capture an RGB photo and send it, along with the temperature, to the administrator's email.
- Network Prevention: All RGB photos and temperature data will be stored on the Raspberry Pi and can be accessed via the USB interface.

Project Expansion:

- Face Tracking: Further adjustments are needed for rotation angles and accuracy.
- Casing: The plan is to utilise 3D printing technology to create a box that achieves a professional appearance and neatness, ensuring a product-level look and cleanliness.

Testing: Utilising an infrared thermometer for non-contact temperature measurements

1. Due to the distance-dependent nature of the thermal imaging camera readings, more accurate real-world data is required to calibrate the impact of distance.
   Plan: Further temperature adjustments will be made based on the data obtained from the infrared thermometer and depth camera.
2. Temperature Testing under Varying Room Conditions:
   Plan: A comparative analysis will be conducted using the infrared thermometer and a temperature sensor to test the temperature under different room conditions.

Leveraging a broader dataset will enhance the project's precision, allowing thorough testing to discern and mitigate scenarios where the infrared camera's performance may falter, thereby preventing potential misunderstandings and accidents due to inaccuracies.

Advantages:

- Strong Reusability: New functionalities can be realised by adding new hardware.
- Good Code Style: The code structure and hierarchy are clear and easy to understand.
- High Productivity: The device is simple, easy to replicate, and manufactured.