IoT for Fever Surveillance

Joel Braganza- z5283268

Nehal Yatham - z5274224

Raghavendran Kasinathan - z5284284

Suharsh Nagasampagi - z5087281

Abstract

COVID-19 has affected our daily lives in a number of ways, the most significant being the efforts to avoid contracting it in the first place. As part of these efforts, we have witnessed the implementation of various methods to ensure the general good health of the population, like social distancing, self-isolation, proper sanitization etc. The current measures to track the infection and check it's spread, while effective, could be more efficient. While the pharmaceutical world continues to grapple with the challenge of developing an efficacious vaccine, using precautionary measures to control the spread of the virus remains to be our most effective strategy. This can be achieved only via the detection and monitoring of early symptoms, and implementing some or all of the aforementioned preventive measures. This report proposes a real-time COVID-19 detection and monitoring system to measure the temperatures of individuals in a crowded space and report instances of high temperatures. The proposed system would use an Internet of Things framework to collect real-time symptom data using a combination of a thermal and an RGB camera connected to a Raspberry Pi, integrated with a web-admin interface to facilitate the detection and reporting of high-fever cases. Based on the results of this project, it is believed that this system is capable of identifying the temperature of an individual from a live video feed, and thus can be a useful tool in the fight against the spread of COVID-19.

Introduction

Since its discovery in late November 2019, there have been more than 55 million confirmed cases and over a million reported deaths owing to the virus, in over 185 countries. Despite a year of rigorous testing and an exhaustive search for a vaccine and treatment, the best medication is still months away from being available to the population of the world. As a result, preventive efforts such as the tracking and monitoring of early symptoms, self-isolation, contact tracing, proper sanitization among others, are paramount.

The use of cameras in places of public gathering to inspect whether measures of social distancing and the wearing of masks are being practiced leads us to the conclusion that Computer Vision (CV) based applications also have an important role in crowd monitoring.

Based on this premise, our proposed system attempts to provide a real-time system that detects an individual with a high temperature among others in a crowd. While the world is slowly returning to normalcy, the need for monitoring symptoms is still far from gratuitous. In Australia specifically, where the general population was only subjected to a partial lock-down for as little as 90 days, in some states, yet a second wave of COVID-19 cases was experienced in Melbourne, the need for such an application that surveys a crowd and identifies individuals with potential early symptoms, is ever-present.

Literature Review

The existing literature lacks a considerable amount of work pertaining to the implementation of Internet of Things (IoT) in the applications of fever surveillance. However, this is some data available from research conducted during the SARS epidemic. A study conducted by Wang et. al [1] over 734 subjects found that syndromic surveillance of fever was the single most effective indicator of a positive SARS test among several other less significant factors proposed by the World Health Organization. While citing geographical restrictions due to data sources as a limitation, the models in the 2012 article by Racloz et. al [2] predicted that Dengue Fever outbreaks could be reduced by up to 73% by implementing proper surveillance strategies.

While the above literature is indicative of the fact that fever is a prominent indicator of COVID-19, despite recommending it, the United States Center for Disease Control presents it as an imperfect litmus test for COVID-19, resulting in up to 44% false-negative [3] readings.

This report acknowledges this limitation and proceeds with the notion that while fever is an inadequate metric, it is still the most pragmatic indicator of the presence of the virus in individuals

Existing Solutions and their Drawbacks

The existing solutions to monitor fever are limited both in scale and feasibility.

A popular method to monitor temperature is using connected thermometers that are linked via a network and notify authorities of any anomalous temperature readings. These are not pragmatic as they require regular manual temperature readings and dedicated thermometers which would impose a significant financial burden for very limited effectiveness in stopping the spread.

Of the other temperature monitoring applications, connected thermometer straps are effective for continuous body temperature monitoring. They do however have a few features that render them sub-optimal for our specific application. Having to be worn on an individuals skin, is a useful feature that can help in monitoring the temperature of infants and the elderly, or any patient unable to continuously monitor their own body temperature, however, handing out wearable body temperature monitors to every individual entering a crowded place is a grossly impractical affair, financially, logistically and economically. Furthermore, they would require actual physical contact thereby failing to limit the chances of spread of the infection.

In addition, contact-based solutions require the user's participation causing an unnecessary dependency.

The most widespread method for fever surveillance currently, is the use of Infrared non-contact forehead thermometers to measure the temperature of each person entering a crowded area like a shopping mall. This is impractical as it causes a bottleneck at points of entry, involves the risk of infection for the individual conducting the surveillance, as well as incurs additional unnecessary labor overheads.

These solutions do not provide an automated system that tracks people’s temperatures in a large crowd.

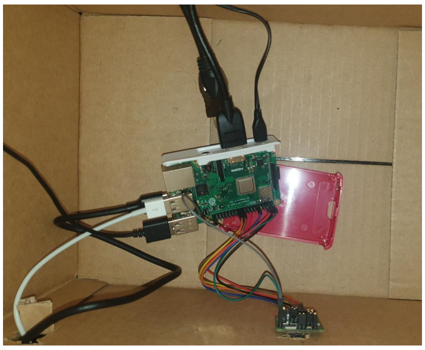
Introduction to our methodology

Continuous body temperature measurement (CBTM) systems play an important role in the medical field today in detecting who has high temperatures. Since high temperature/Fever is a vital symptom of COVID, it is really effective to track down all individuals in a crowd with high temperatures and keep the society safe. The traditional contact body temperature measurement has the problem of wasting consumables and causing discomfort. There is a huge problem in tracking down the individual and alerting the authorities in case of a mall or any crowded place. To address these issues, we propose a non-contact, automated CBTM system using a thermal camera and an RGB camera.

This method implemented to detect the temperature of individuals in a crowd involved the conjunctive use of two cameras, an RGB camera and a thermal camera, connected to a Raspberry Pi. The Raspberry pi used in our system was the Raspberry Pi version 3, the thermal camera was the FLIR Lepton Thermal Camera and the RGB camera was the Intel Realsense D435 Depth Camera.

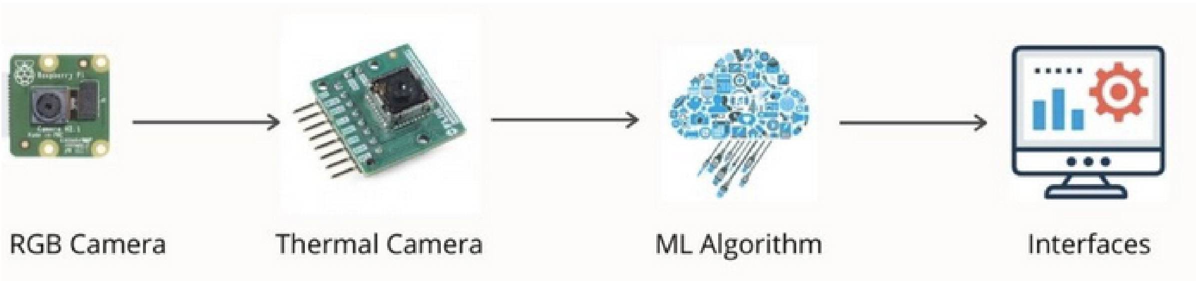
Hardware

Prototype Setup



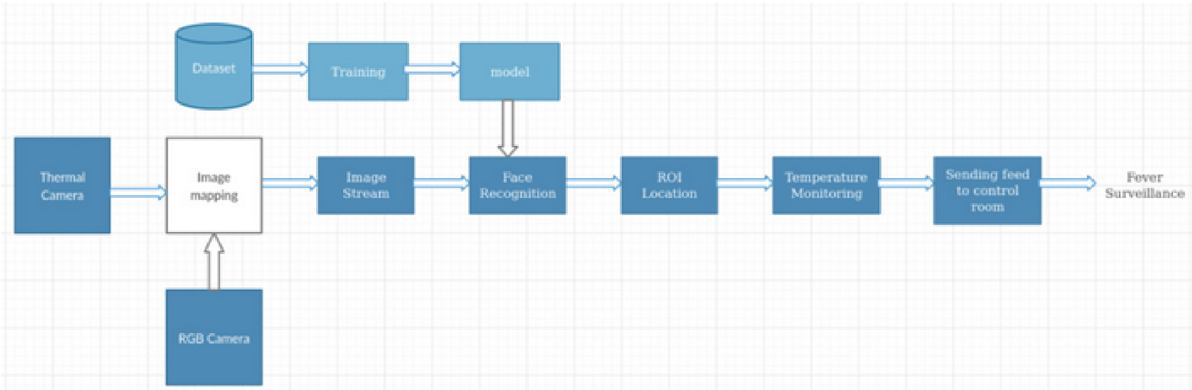
Cameras and Raspberry Pi Connections

Hardware Flow Diagram



Flow Diagram

Block Diagram / Workflow



Block Diagram

Continuous body temperature measurement of people in a crowded place will benefit a large set of people specifically the elderly and children.

Continuous body temperature measurement plays a pivotal role in stopping the spread of the COVID-19 virus. Identifying people in a crowded place with a high temperature can vastly affect the number of people affected as research shows that even one infected person in a crowded mall, has the potential to infect up to 11 people within a 1.5m radius of themselves who can propagate this exponentially. It is thus paramount to devise an effective technical strategy to combat the spread by identifying potentially infected individuals.

Our system uses the FLIR Lepton InfraRed camera to capture thermal images using the Long-range Wave InfraRed (LWIR). Parallel to the FLIR, for alerts and faced detection purposes, we have an RGB camera along with the FLIR.

Once captured and mapped with the detected temperatures of each person, an alert is raised for if the system finds individuals with high temperatures i.e. if the value is above a certain prescribed threshold.

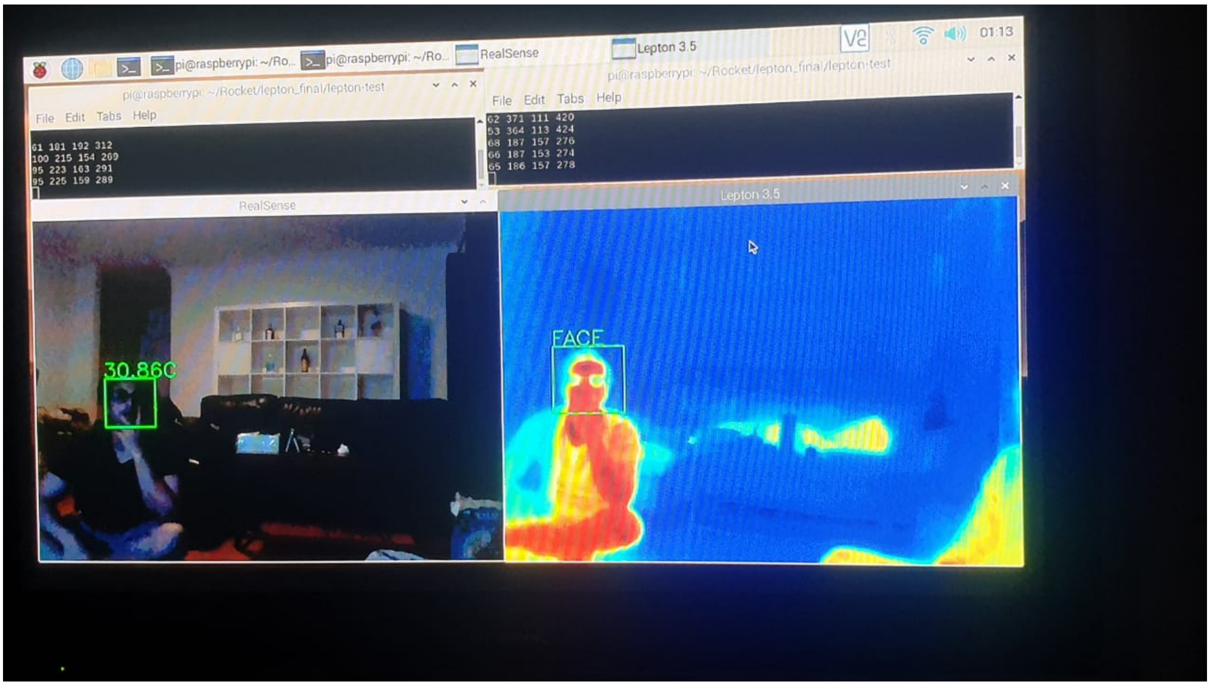
The alert includes a picture of the individual, their temperature, and the last location in terms of which camera identified them last. (Depicted Live Location).

Face Detection

For face detection on the RGB images, we have used a simple cascade classifier, a pre-trained model that uses the HAAR CASCADE Frontal Face Dataset. We chose this because of its relatively high accuracy with very little demand for resources since the raspberry pi has a low RAM memory. The cascade classifier recognizes features of a human face like eyes, nose, forehead etc. and if it identifies a face in the image pixels. we use the coordinates of these pixels to draw a bounding box to visualize that a face has indeed been detected.

Thermal Image Capture

The FLIR captures thermal images at a resolution of (160,120). These images have been resized to (480, 360) to match the output of the Realsense RGB camera. We normalized the data and also applied a color map to convert the raw pixel thermal data into something which can be understood by the human eye when streamed.



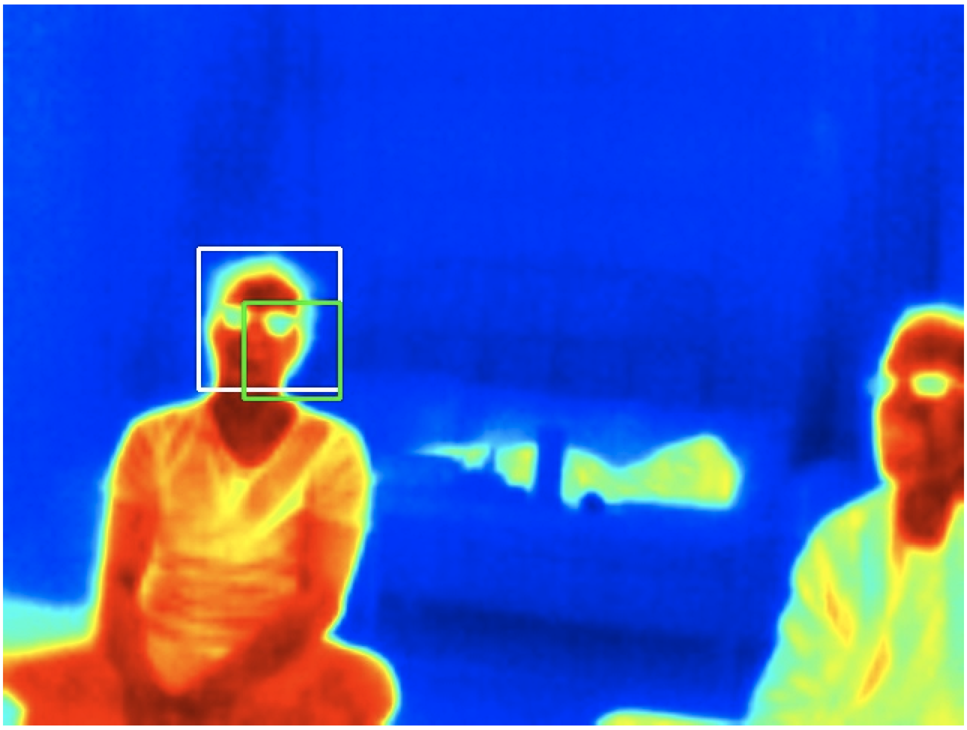
Thermal and RGB Camera Output

The image above has the output of the thermal and RGB cameras with the face detection algorithm running on the both of them. The bounding boxes, as mentioned above, indicate that a face has been location in that region of the image stream.

The number, "30.86C" in the RGB output is the temperature of the detected face.

Image Mapping and Temperature Calibration on RGB stream

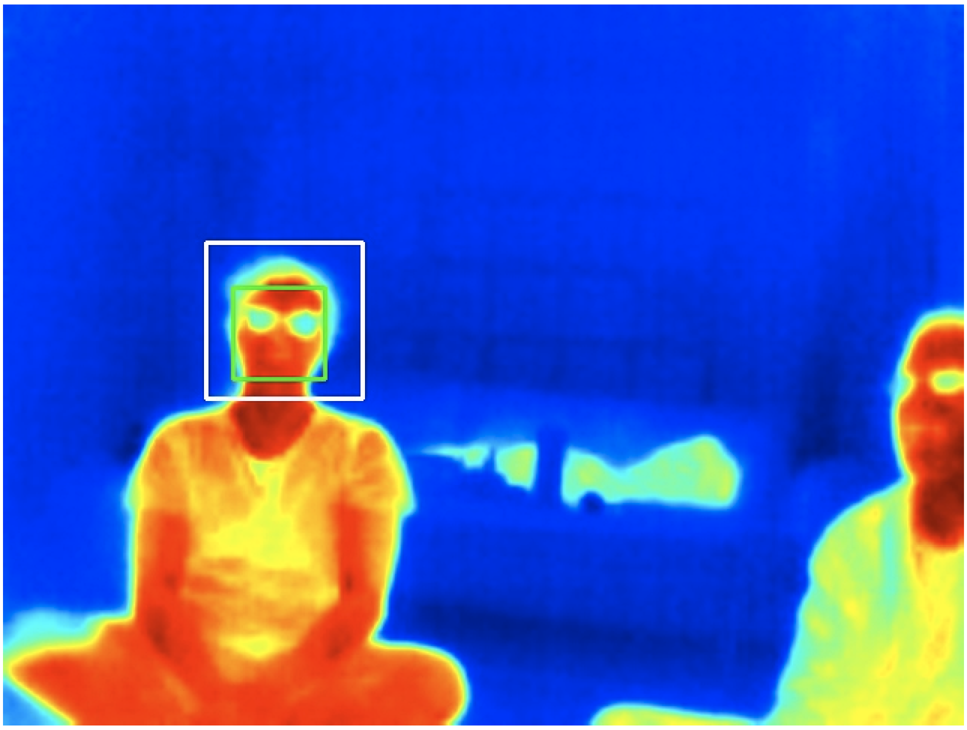
Since the placement of the two cameras differs slightly, the relative positions of faces differs in RGB and thermal image.



Thermal image ROI before calibration

The image above is the thermal image output. The face detection is run on both the thermal stream and on the RGB stream. The algorithm gives us the coordinates of the face in both the thermal image and the RGB image. We drew out boxes of both face detection to help visualize the difference in the relative placements of the face. The green box is basically where the face is located on the RGB image stream and the white, on the thermal stream.

To make up for the discrepancy, we manually adjusted for the difference to give us the following output.



Thermal image ROI after calibration

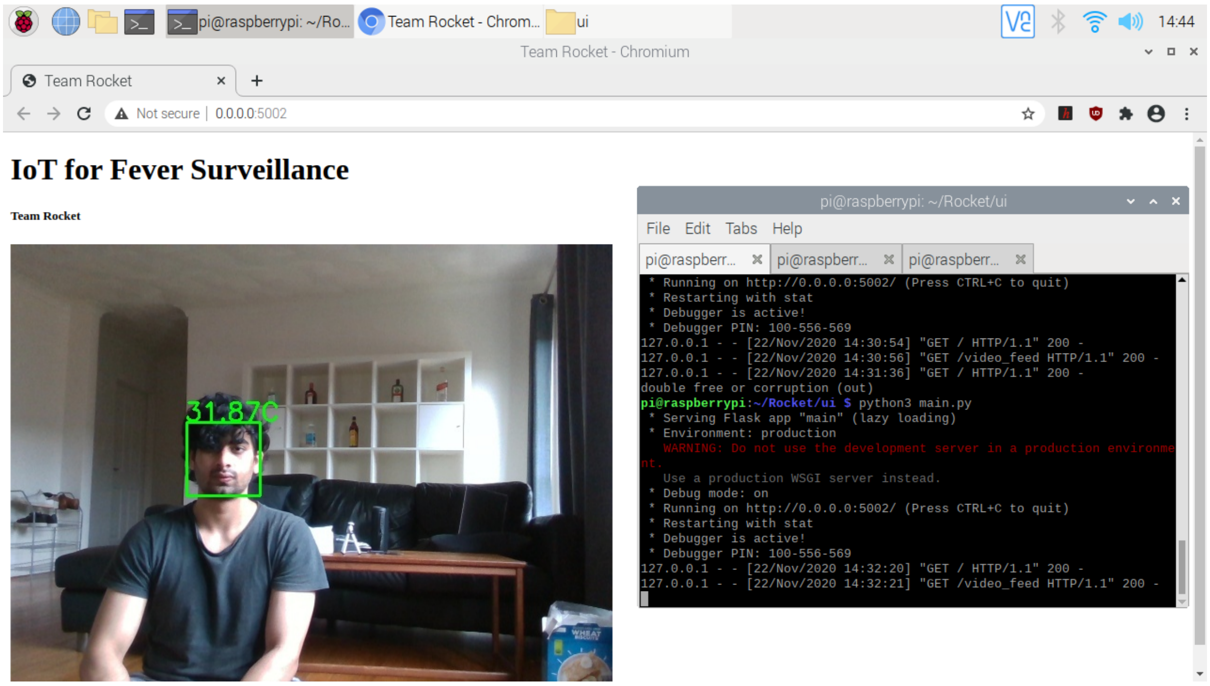
Now we have mapped the faces together.

This is important because need to send the raw thermal values of that exact region to find max temperature of that region, which is the face.

So every time a face is detected in the RGB stream, not only do we draw out a bounding box in-place on the RGB stream to visualize the location of the face, but also, we adjust these coordinates to fit the face on the thermal stream and calculate the max temperature in that region.

Web Interface

The final system streams the overlay over a very basic interface through flask.



Output on web interface

Evaluation

The project is meant to solve the current COVID problems of not having a person with high temperature enter the hall. Maintaining COVID safe measures demands a dependable system that is connected and monitored. For this the system detects all the faces and gives out the temperature of the face instantly, highlighting the face with the higher temperature and all of this is streamed on a web interface.

Evaluation hence is done in 3 stages. Starting with the Hardware evaluation, it is made sure that the cameras are hard calibrated accurately so as to give the right temperature from the lepton camera for the faces detected in the RGB camera.

It was noted that the Lepton camera is subjective to the distance of the face from the camera, when it calculates temperature. i.e. Farther the face, lesser the temperature detected. Hence, upon detecting the distance using the Depth RGB Camera, this was solved and calibrated as well. For real time it is not implemented for the reason that Rpi3 becomes slower to detect faces and the distance at the same time. This limitation can be solved if Raspberry Pi 4 is used for an onboard processing.

After Calibrations, proceeding the software evaluation for the RGB camera, Haar Cascade classifier is used to detect the face in the RGB Camera with an accuracy of almost 98%. The next evaluation was done on the Lepton camera. On detecting a face, when asked for a thermal image from the Lepton camera, it is super efficient to simply click a frame and compute the temperature instantly. The temperature has an error rate of +/-3 degree Celsius that is reduced at a great rate by making the rgb take a pipeline of frames rather than a single frame and then compute the net image to detect the face, after which the lepton will capture the thermal image with the perfect temperatures detected.

On an overall when tested with different scenarios of face temperatures, such as after a hot water face wash the skin temperature rose to 34-35 degree Celsius, when face was rubbed with ice for a long time it showed a skin temperature of 30-31 degrees Celsius while normally the skin temperature was steady at 32-33 degrees Celsius.

We also made it so that the when a face with a high temperature is detected, the system saves the frame and turns the green bounding box to red for reference.

Conclusion and Future Work

Thus, a prototype model for what a temperature surveillance system could look like has been achieved.

The following are the drawbacks of the system:

- The system fails in part or completely when a significant portion of the face is covered because facial obstruction like face mask or a covered eye wear can prevent the thermal camera from recording temperature values and if a significant portion of the face is covered, can mess with the face detection algorithm.
- The thermal camera has an error rate close to 3°C and the error seems to increase with an increase in temperature. Although we did try to incorporate this aspect to the system, the Raspberry pi could not handle the load leading to us ditching the plan to correlate and test the distance-temperature interaction.
- Upon research, we discovered the surface skin temperature of an average healthy human is between 33°C - 35°C. This variability in temperature can cause difficulties in monitoring temperature.

Future Work

- Correlating depth with error to adjust for temperature for individuals at a greater distance from the camera on a heavier processing board with improved accuracy.

Acknowledgement

We would like to thank the other team working on the same project, "Make Moodle Great Again" and Chris for helping us debug and fix the thermal camera output and the lepton module. They were great help and played an integral role in helping make this project a great success.

References

[1] Wang, L-M et al. “The rationale of fever surveillance to identify patients with severe acute respiratory syndrome in Taiwan.” *Emergency medicine journal: EMJ* vol. 23,3 (2006): 202-5. doi:10.1136/emj.2005.027037

[2] Racloz, Vanessa et al. “Surveillance of dengue fever virus: a review of epidemiological models and early warning systems.” *PLoS neglected tropical diseases* vol. 6,5 (2012): e1648. doi:10.1371/journal.pntd.0001648.

[3] Sumathi Reddy “Temperature isn’t a good litmus test for COVID-19” *Wall Street Journal 2020.*

[4] *Yamaguchi, Masahiro & Truong, Phuc & Mori, Shohei & Nozick, Vincent & Saito, Hideo & YACHIDA, Shoji & SATO, Hideaki. (2018). Superimposing Thermal-Infrared Data on 3D Structure Reconstructed by RGB Visual Odometry. IEICE Transactions on Information and Systems. E101.D. 1296-1307. 10.1587/transinf.2017MVP0023.*

[5] A Thermal Camera based Continuous Body Temperature Measurement System Jia-Wei Lin, Ming-Hung Lu, Yuan-Hsiang Lin

[6] An IoT-based framework for early identification and monitoring of COVID-19 cases Mwaffaq Otoom, Nesreen Otoum, Mohammad A. Alzubaidi, Yousef Etoom, Rudaina Banihani

[7] An Overview of Thermal FaceRecognitionMethodsM.Krišto, M. Ivašić-Kos

[8] https://github.com/groupgets/pylepton

[9] https://github.com/akmamun/live-stream-zeromq