Privacy-aware, Low-power, Occupancy detection of Multiple People using a Thermophile Sensor

Kilian van Berlo
MSc Embedded Systems
Delft University of Technology
Delft, the Netherlands
S5436737

***Abstract*—Privacy-aware, low-power, and device-free indoor occupancy detection is highly sought after in order to enable smart environments. IR-based solutions seem to obey to these demands, however, are quite limited in terms of real-life application due to several obstacles. So, another, more appropriate solution, is to be found. For this, Ranga Rao and Sujay from the Networked and Embedded Systems group of the TU Delft already developed a device that integrates, among others, a thermophile sensor. For me, it is now the task to explore the sensor usage possibilities. This is done so I can get more familiar with three topics:**

1) **Acquiring sensor data appropriately**
2) **Applying computer vision techniques to detect people based on IR information**
3) **Sending information extracted from the sensor, using the MQTT protocol, to an Azure IoT Hub.**

For the MQTT connection with the IoT Hub, also a separate code is created for testing and can be found in the folder standalone_MQTT_IoT of the GitHub repository of this project [1]. In this repository, all code used for the project can be found.

***Index Terms*—Privacy, Low-power, Thermophile, Infrared, Occupancy detection, MLX90640, Image processing, Contour detection, MQTT**

I. Introduction

These days technology is becoming more and more apt in collecting data about each and every individual. With increasing capabilities come increasing threats which have to be considered. With everyone's data out there, privacy will become a myth from the past. Hence, it is important to maintain rules to ensure the protection of all personal data so that privacy does not cease to exist. These rules in Europe fall since May 2018 under the General Data Protection Regulations (GDPR) [2]. With these regulations new challenges appeared such as tracking people within indoor environments without gathering any individual data. This cannot be done with the standard cameras since these would breach the GDPR, given that people are recognizable through them. Next to camera's, also techniques like Ultra-Wideband, RFID and Bluetooth fall short since these also need personal data in order to properly operate in the task of detecting people [3].

In the meantime, several other options have been considered in literature to still successfully complete this feat that before GDPR was relatively easy to do. Approaches that comply with GDPR often include something with either LiDARs, Ultrasound or Infrared [3] [4] [5]. The challenge with these techniques is that one has to consider what it desires most, either deployment cost or efficient energy consumption in exchange for the level of accuracy. On the one hand, LiDARs for example, are in general more accurate than Infrared, however a LiDAR also uses more energy in order to be more accurate.

In order to consider the right technique it is also important to consider what it is generally used for the most. One can imagine this technique is mostly desired for safety-critical systems in company buildings where one desires to know who enters the building and who leaves, or if someone is behaving suspiciously. In case of an emergency this system should then work independently from the already existing infrastructure since especially in these cases one ought to utilize this technique as an aid. This logically results in a need of the device to be independently operating, GDPR compliant and preferably with a low power consumption.

Knowing these demands, and comparing the different options, eventually a thermophile sensor (IR) seems to be the best choice for this person detection project. Since the project is only a sub-part of the entire person tracking system, high complexity is not needed. If one, for example, would need to measure the position of people in a room, one might say that a LiDAR might be more appropriate given its ability to both detect a person in a room and derive its position. In our case, however, only detection is needed. Therefore, IR seems to be the best fitting option, utilizing little power while maintaining reasonable accuracy [6] and being a relatively low-complexity solution. Another plus for the use of IR sensors is that it can also be used for other applications such as being a component for smart HVAC or lightning systems that help in saving power consumption [9]. Since both these applications can use the IR sensor, this could save costs as well.

So, with costs, complexity, accuracy and power consumption the IR sensors seem to be the right solution. However, as every advantage has its disadvantage, so too has a thermophile sensor. Challenges that are faced, also in this project, do not always have a simple solution. For example, how to make a distinction between hot objects and people since both look the same in the face of heat emission. Another question one might ask is what happens if someone is wearing a thick sweater or jacket? The IR radiation that someone emits is diminished with every layer of clothing someone is wearing since layers do not emit radiation, they block it.

Decided on using an IR sensor, in this project the first challenge is to acquire the data from the MLX90640 IR sensor so it is ready for use in the image processing that is needed in order to detect people. Knowing this, it makes sense that finding a good approach for counting people in a frame and subsequently implementing this approach is marked as the second challenge in this project. To do this, multiple techniques are possible, some especially helpful for thermophile data, others work, but need a little adjustment from its standard camera applications. For this project promising person detection algorithms have been explored, such as YOLO [10], HOG [11], R-CNN [12], contour [13], and canny edge [14]. By comparing these techniques a final approach is decided on.

To make a proper choice on the person detection technique to use, a Design Space Exploration (DSE) on the different approaches is done where attention is paid to the aspects deemed most important. These aspects are accuracy, processing complexity, and implementation complexity. An overview of the results of the DSE can be found in Table I, where a + means it has a relatively positive effect on the choice and a – means it has a relatively negative effect on the choice. The right outermost column adds up all values and shows the eventual best choice for this project. Since both processing and implementation complexity are good for canny edge and contour, while not performing much worse on accuracy, these are the preferred options of choice [10] [11] [12] [13] [14]. The YOLO and R-CNN also are techniques which became less interesting when it was realized that a huge test-set was needed, which was going to take too much time to create within the time limits of this project, hence making those solutions even more complex. Canny edge detection, as well as contour detection techniques, are widely applied in various computer vision systems [15] [16] [17], making the choice between these two approaches not significant for now. After exploring both options and experimenting with them in the application, eventually the choice has fallen on the contour detection technique.

TABLE I
DESIGN SPACE EXPLORATION FOR MULTI-ROBOT COMMUNICATION

	Accuracy	*Processing*	*Complexity*	**Total**
YOLO	+	+	-	1
HOG	+	+/-	+/-	1
R-CNN	++	-	- -	-2
Contour	+/-	++	+	3
Canny-edge	+/-	++	+	3

After the person detection has been implemented successfully, the third and last challenge of this project has to be addressed. This challenge has to do with the fact that the system should be able to operate independently at any place while still providing information to the stakeholders. In order to do this the last and third challenge is therefore to establish a MQTT connection with an Azure IoT hub so the data of how many people are detected is send to a cloud platform without any cables required for the device to be connected. This way the entire system can be dynamically moved wherever it is needed. Hence, in total three challenges are addressed in this project. All of them on very different aspects of an embedded system (sensor data acquisition, image processing and networking systems).

II. METHOD

A. Design

To determine how well people can be detected using the software developed for the thermophile sensor, several experiments have been conducted in two comparable environments. One setup (setup 1, Figure 1) is at my parents' place and the other setup (setup 2, Figure 2) is in my student room in Delft.

Fig. 1. Setup 1 at parents

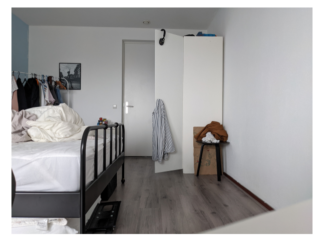

Fig. 2. Setup 2 in the student room in Delft

These environments were used since halfway the past period I moved back to my room in Delft, making a new environment necessary. Throughout the experiments people and objects have been standing in these rooms for GDPR compliant detection by the thermophile sensor. Next to the setup, also the choice for experiment hardware is important. For the experiment I used the MLX90640 32x24 IR array thermophile sensor. This sensor is used since it covers a decent room size, which it is able to do due to its Field of View (FoV) of 110° x 75° and a good resolution of 32x24 pixels (768 pixels in total). The maximum refresh rate of this sensor is also 64Hz, which should be sufficient for the use case of identifying people since people don't usually move through an indoor environment at high speeds. Now it is true that this thermal camera is not unique in what it does, there are other sensors out there that do approximately the same thing. However, this option provides the perfect mix of specifications for the use case of this project.

B. Participants and procedure

Since the experiments are not dependable on differences between people, only few participants (7 in total) have been used to create the desired test environments. These participants were informed about the fact that their privacy would not be infringed and after agreeing, instructions were given on what to do before entering the room. Once they were positioned correctly, the thermophile sensor was activated and started gathering information.

C. Measurements

To answer the research questions mentioned in the introduction, several situations have to be tested in order to get a good impression of the possibilities of the developed software. First of all, the background in both situations has to be evaluated to see whether there are any disturbances that can limit the performance of the other tests. As you can see in Figure 3 in the lower left corner, other objects can be limiting since they also emit heat, which the sensor might identify as a person.

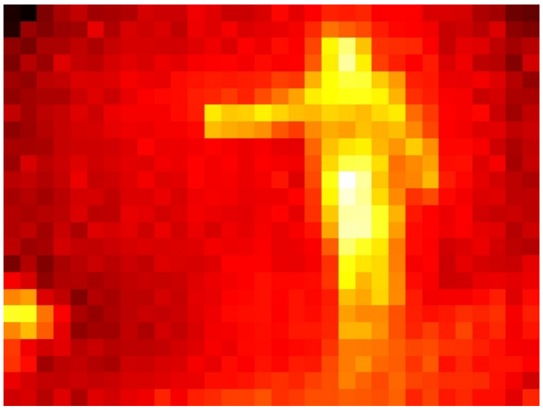

Fig. 3. Background noise (lamp) clearly visible

Second, it is important to see how well the sensor can detect participants (e.g. clothing types, obstacles between person and camera) and what characteristics can be identified based on solely the heat emitted (hottest parts in the body). This could help later on in identifying people more accurately. Once all these measurements have been done, the next, more complicated, stage in the person detection can be attempted to be solved. This stage is about detecting multiple people, either clearly separated or clustered together (behind or directly next to each other).

III. Development

The first step in the development of the occupancy detection through a thermophile sensor is about the data acquisition from the sensor. After the data is acquired, the next step is to count the number of people in the Field of View (FoV) based on the data and finally to send this with the MQTT protocol to a server.

A. Data acquisition

Acquiring the data from the MLX90640 is done through a rather complicated process since all data it sends is encoded in 16-bit raw values instead of just directly giving the absolute values. The first action to perform in order to start a successful data acquisition is to extract calibration data from the EEPROM of the sensor. This EEPROM data consists of calibrated constants specific to the sensor and is of utmost importance for a successful computation of the eventual temperature values of each pixel. In order to acquire this information first an 8-bit value of 0x01 has to be sent to the sensor before it can be read back. Based on the 16-bit EEPROM values extracted, the calibrated constants of the MLX90640 can be calculated. All constants and their descriptions and calculations can be seen in the data sheet [6] and my implementation for these values can be found in the file 'calibration_restoration_EEPROM.py'.

Once the calibration data is extracted, the actual temperature calculation for each pixel can take place. This process is started by sending the sensor an 8-bit instruction with the value of 0x02, after which the data can be read in a continuous stream at a rate of 4Hz. A rate of 4Hz means that each 250 ms, data for a new sub-page is available. Since each frame of 32x24 pixels consists of two sub-pages, this means that after each half a second a complete new frame is received. These sub-pages can be sent in two ways, either in a chess or a TV interleave pattern as can be seen in Figures 4 and 5 [6]. After a sub-page is received, first it is checked which of the two sub-pages it represents after which the raw values in the sub-page are converted to the measured temperatures for each pixel using Equation 1, which is based on instructions in the data-sheet [6].

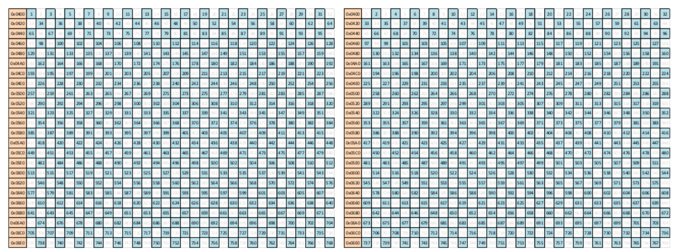

Fig. 4. Chess pattern

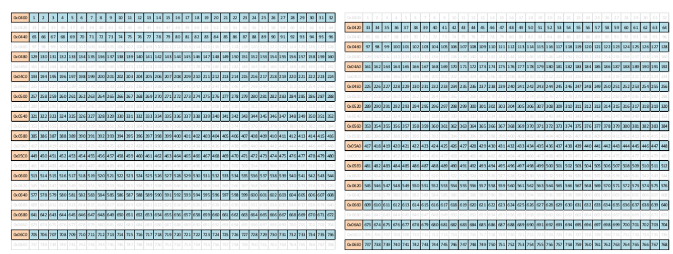

Fig. 5. TV interleaved pattern

$$T_{o(i,j)} = \sqrt[4]{\frac{V_{IR(i,j)_{COMP}}}{\alpha_{comp(i,j)} * (1 - K_{sTo2} * 273.15) + S_{x(i,j)}} + T_{a-r}} - 273.15 \quad (1)$$

$$S_{x(i,j)} = K_{sTo2} \sqrt[4]{\alpha^3_{comp(i,j)} * V_{IR(i,j)_{COMP}} + \alpha^4_{comp(i,j)} * T_{a-r}} \quad (2)$$

In the equation To(i,j) is the temperature reading for each pixel, $S_{x(i,j)}$ is shown in Equation 2 [6], $V_{IR(i,j)_{COMP}}$ is the offset compensated raw value for each pixel, $\alpha_{comp(i,j)}$ and K_{sTo2} are constants corresponding to each pixel, and T_{a-r} varies with the ambient temperature. For all of these values it holds that $i \leq 32$, and $j \leq 24$, with i and j both being Natural numbers $\mathbb{N}$. These parameters are calculated using several of the constants and pixel offset values that are extracted from the EEPROM calibration data. In equation 1, however, the absolute temperature is calculated for each pixel, which takes more energy than when calculating relative temperatures. Since the project seeks to minimize power consumption, the calculation used in the project is simplified by working with relative pixel data difference rather than absolute data. This is done by using Equation 3 instead of 1 after it was found that K_{sTo2} and $S_{x(i,j)}$ in that equation did not seem to be of added value.

$$T_{o(i,j)_{comp}} = \left(\frac{V_{IR(i,j)_{COMP}}}{\alpha_{comp(i,j)}} + T_{a-r} \right) * 10^{-9} \tag{3}$$

In the equation $T_{o(i,j)_{comp}}$ is the compensated (for offset and irregularities in zones) raw value which is of lower complexity and can be run on MCUs which run on much lower power. The implementation for both these calculations can be found in the file 'temperature_calculation.py' and the difference in result is shown in Figures 6 and 7, where both equation results are compared. In this figure one can see that the difference in result between both equations is barely observable, indicating an effective simplification of the calculation while preserving the gradients.

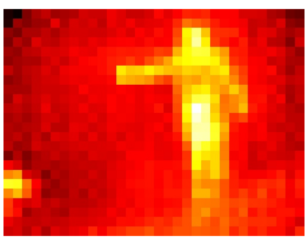

Fig. 6. Absolute temperature values used

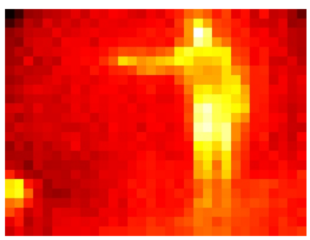

Fig. 7. Relative temperature values used

Note that with both equations the value calculation for each pixel also takes into account the accuracy zones of a frame and compensates for them in the calculation. There are three accuracy zones in a MLX90640ESF-BAA frame (Figure 8 [6]), Zone 1 with an accuracy of $\pm 0.5°C$, Zone 2 with an accuracy of $\pm 1°C$, and Zone 3 with an accuracy of at least $\pm 2°C$ (Melexis, 2019). Knowing these differences in zones, the values in zones can be corrected for through offset calculations for ambient temperatures between $0°C$ and $100°C$ Any higher or lower temperatures have different accuracies, but are not deemed important for this project given that the project is about detecting people in an indoor environment.

B. Image Processing

Once all data is correctly acquired from the sensor it is time to start extracting useful information from that thermophile data. In this case, the data is considered useful if it aids in detecting the number of people in the frame. In order to do this three steps are taken consecutively. First, the background

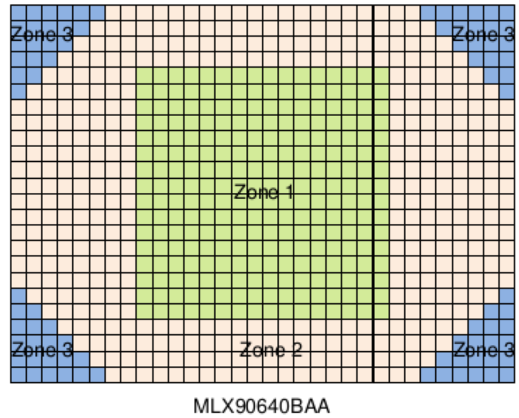

Fig. 8. Three accuracy zones of the sensor

noise is removed from each frame making it easier to detect changes in the environment (aka people walking around). After the background is removed, the next step is to interpolate the image in order to gain a better accuracy to work with in the final step, the contour detection of people. For this contour detection multiple approaches are possible with for example using a Sobel filter [7] or applying morphological transformations like the morphological gradient [8]. These techniques are mainly used to emphasize the edges in the image so it eventually easier to recognize these when trying to detect objects in a frame. Often times, when the picture received some preparatory treatments like the ones mentioned, it is then easy to determine the contours. In the case of this project, there are already little details present in the image after interpolation which makes pre-processing techniques less crucial. Therefore, after background filtering and interpolation, applying any other step did not matter for the performance of the contour detection. Since the other steps did not show to be important, and given our application has to be as light as possible, it was chosen to not add them. The eventual image processing approach is therefore rather simple and is shown in the flowchart in Figure 9.

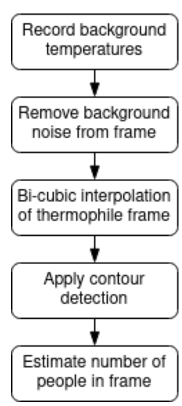

Fig. 9. Flowchart of the image processing approach

1) Background filtering: The first part, the background noise filtering, is done by first taking the temperature data from the initial frame of the thermophile data. This initial frame is a frame without any people in there, so only with static objects. The example shown in Figure 10 contains a background frame of setup 2, with a warm light and a small part of the tabletop the sensor is standing on. As you can see these static objects normally emit a bit of heat and should be eliminated from the eventual image.

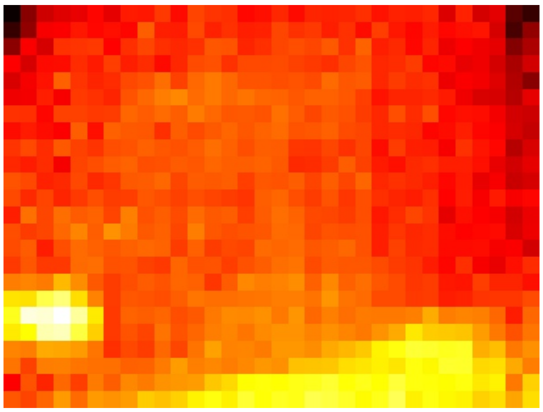

Fig. 10. Thermophile image of the background in Delft

The background temperature data is used as the baseline deduction for each of the thermophile frames that follow. By doing this, the temperature of all static background is canceled out, making the difference between the background and people in the frames as big as possible. Figures 11 and 12 show the difference between a frame containing a person, without background subtraction, and a frame containing a person, with background subtraction.

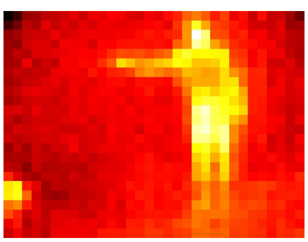

Fig. 11. Original thermophile image

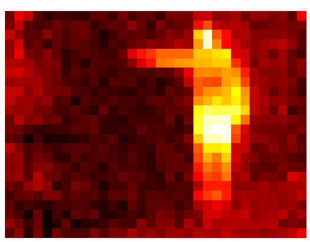

Fig. 12. Thermophile image with background heat subtracted from the frame

2) Data interpolation: For the interpolation of the image there are multiple options. Since the goal is to make it deal with the data in real-time we should also keep in mind that the processing time should not be arduous and that the data provided is continuous. Hence, multiple well-known interpolation options are considered such as Nearest-Neighbour, Bi-linear interpolation, and Bi-cubic interpolation. As one might expect Nearest-Neighbour did not result in any useful improvement compared to the original image (Figure 13), leaving the data rather ”blocky” (Figure 14). This is because it uses only the nearest cell center on the input frame.

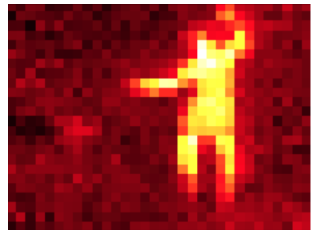

Fig. 13. No interpolation

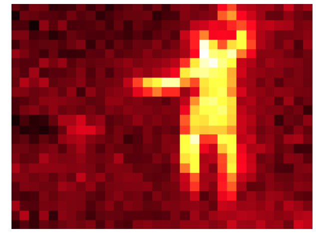

Fig. 14. Nearest-Neighbour interpolation

Bi-linear interpolation on its turn uses the four closest cell centers to compute a weighted average for the new pixel value. Through this approach, Bi-linear interpolation already shows significant improvements compared to Nearest-Neighbour (Figure 15) and might already be useful for the person detection to be achieved. However, even here, still some slopes seem to be jagged. To smoothen the image further, a more accurate interpolation method is also considered. This method is called bi-cubic interpolation (Figure 16) and takes even more neighbouring pixel values into account, sixteen to be precise. As you can see in the Figures, eventually the most promising results were achieved using the bi-cubic interpolation method.

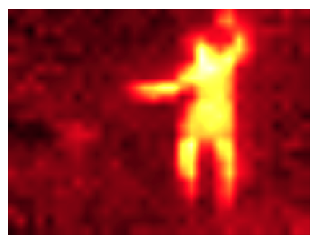

Fig. 15. Bi-linear interpolation

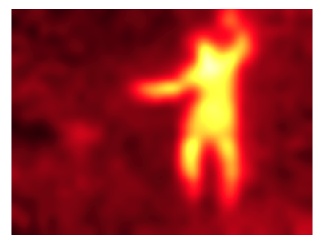

Fig. 16. Bi-cubic interpolation

3) Contour detection: Contours are a useful tool for object detection and recognition. A contour is a curve joining all the continuous points with the same intensity along a boundary. In the original image provided, it might still be rather difficult to recognize the contour so in order to make it easier threshold or canny edge detection can be used to make the contours more clearly visible. For this project the chosen approach is to utilize threshold detection for obtaining the goal. This means that every time a frame is received from the MLX sensor and processed, the image is saved as JPEG and converted from BGR color format to an HSV color format. Then, consecutively, on this HSV frame a filter is applied which selects only the pixels that, according to a given range of HSV values, are supposed to be detected, in other words: the threshold (17). Once the desired pixels are known, a contour is created around all sets of pixels (18). By counting the number of contours in the frame, the number of people can be determined.

Multiple approaches are definitely possible, as already discussed in the introduction. However, eventually this approach turned out to be the best one for this project after experi-

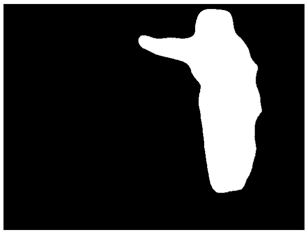

Fig. 17. Black and white threshold for HSV values

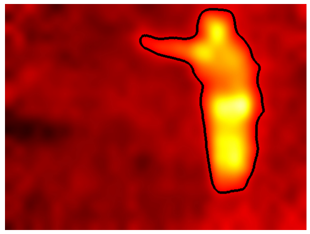

Fig. 18. Contour for person detected through contour detection

menting with several options. For normal person detection, detection based on a threshold might not have been the best option. However, given that in this case the colours do not depend on the actual colours, but only on relative temperature colours with the background as a baseline, it is easy to set boundaries for the detection of people. In this way, people always show to be the brightest part of the image and are always yellow/orange (assuming that in general people are the dynamic objects that emit the most heat in the frame).

C. MQTT

MQTT is a protocol that is standard for IoT messaging where it is designed as a lightweight publish/subscribe approach for connecting smart devices over minimal network bandwidth [18]. Since the LOCI is also such a device, it speaks for itself that MQTT is ideal for use in this situation. In order to establish a testing environment it was decided to make use of the service provided by Wapice IoT-TICKET. They have an Azure IoT Hub that will receive the sensor data of this program using the MQTT protocol. Once this data is received at the Azure IoT Hub, the MQTT trigger function of Wapice will forward the data to try.iot-ticket.com, where the data and the device sending the data then appear. On the IoT-TICKET system it is then easy to create simple, lightweight dashboards for showing real-time data originating from the sensor.

For the development of this solution a basic understanding of the IoT-TICKET system is needed. However, the most important part is getting acquainted with the Azure IoT SDK that aids in connecting to the IoT Hub while making use of the MQTT protocol. Development for establishing this connection can be found integrated in the *project_main.py* file or separately in the folder *standalone_MQTT_IoT* where a test integration in both Python and C# is present [1].

IV. Results

A. Data acquisition

The first results to be discussed are related to the first challenge, the data acquisition. In order to properly test this, two things can be done. The first one is more subjective where one looks at a generated image and looks whether it seems right. However, this of course does not give any guarantees, but merely an initial idea of its correctness. Therefore, the only proof of work approach would be by having an example of the data to compare the results with. Fortunately, on GitHub, Melexis provides one set of example data [19] and when comparing our results to the example everything matches exactly. Take in mind that this is with the exact restoration formulas described earlier. After the optimizations I applied to the code, the results are different for the eventual temperature outcomes since these values are relative and not absolute. This difference, however, does not affect the eventual image result. Knowing all these values match well and show a logical image compared to the test data provided, it can be concluded that the data acquisition from the IR sensor has been performed successfully and the next (bigger) experiments are ready to be performed. These experiments are with regards to the image processing.

B. Image processing

With image processing there are several cases to be tested for, for which results are to be obtained. Next to the background noise, as already discussed in the earlier section, it is important to know the heat emission from a person him/herself. As we can see in the Figures shown before, our sensor detects the hottest parts of a person around the head and below the hips. In between these two areas the IR sensor detects less heat which is due to someone wearing thicker clothing (Figure 18), thereby cancelling heat emission towards the sensor. This is an important characteristic to take into account when designing an occupancy detection technique. The next logical step is that one asks that if clothing already have this effect, what effect can other obstacles have on the identification of people within a captured frame.

1) Impact of obstacles: To analyze the impact of obstacles, two cases that could occur are presented. Given the limits with objects available in the room a chair was used in both setups. One is when someone is standing behind a chair (in, for example, office spaces) as depicted in Figure 19, and the other one is when someone wraps his arms around the back of the chair while sitting on it (Figure 20. With the first setting it becomes clear that when an object blocks part of a body, it sees the disconnected parts as different people in the frame. This same challenge holds for the second setting again, where one can see that the head and the arms are considered separate entities. To tackle this problem, multiple solutions are be proposed, however, all of them seem to eventually fail in some situation or another, these are discussed in the next section, the discussion. What became clear from this is that it is not possible to eliminate the effects of the obstacles, as they cannot be detected by the sensor. The sensor needs all people in the frame in line of sight for it to be able to detect the number of people with a good accuracy (if they are even detected at all).

2) Impact of multiple people present: Another interesting topic to analyze is the impact of multiple people in the frame. How well does the image processing function with two people present in the frame, perhaps even more. As can be seen in Figures 21 and 22, the image processing works well. These are the images of one of the many tests with multiple people, where detection accuracy turned out to be around 96%

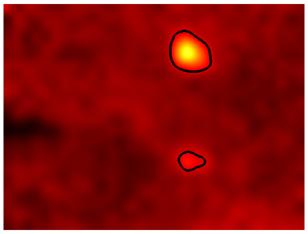

Fig. 19. Standing behind chair

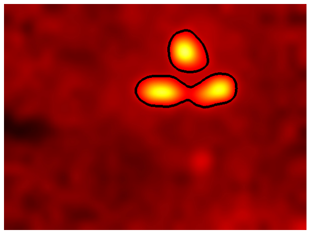

Fig. 20. Arms wrapped around back of chair while sitting

accuracy. Note however, that in these cases it was made sure that the people held a decent distance from one another. Once the participants were instructed to get closer together (Figure 23) or get behind each other from the viewpoint of the sensor as in Figure 24, detection difficulty considerably increased. Detecting multiple people that were intermingled did not work for the current image processing as none of the experiments successfully recognized the amount of people present in these cases. Looking at the images it is not unexpected of the algorithm to have troubles making a distinction between the people. For the two people holding each other's shoulder in Figure 23, adjustments to the approach can be imagined (e.g. based on the gap in the middle). For the two people behind each other (Figure 24), however, it is not clear what should hold them apart in the image, at least not in a still image. Some of the possible improvements for the approach are discussed in the next section, the discussion.

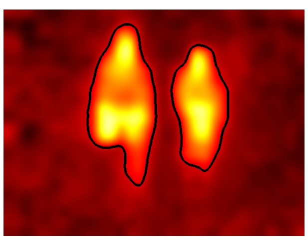

Fig. 21. Two people with space between them

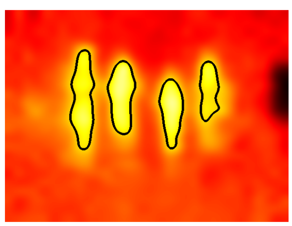

Fig. 22. Multiple people with space between them

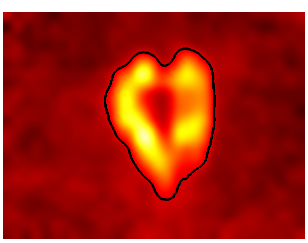

Fig. 23. Two people holding each other's shoulder

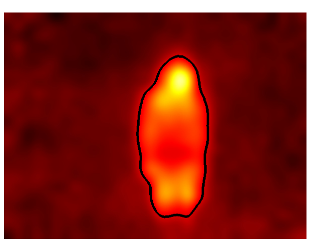

Fig. 24. Two people, one standing, the other one sitting on the ground in front

C. MQTT

For the establishing the MQTT connection and sending data over to the IoT Hub, there are basically two goals to check to make sure that everything works as expected. Firstly, the data has to be successfully send to the Hub. Secondly, the data has to be successfully received by the Wapice IoT TICKET server. Connecting to the Azure Hub proved to be working very well once I understood the SDK well enough. When a message is successfully sent from the application, a string with the message is output in the terminal, showing the message that has been sent (Figure 25).

After some struggles with Wapice also the connection from the Azure Hub to IoT TICKET showed to be working flawlessly, thereby completing the communication channel from the sensor to the data server. At the data server a simple dashboard was build to show the information that was being received in real time (Figure 26).

Not that a message is being send to the IoT Hub that does not necessarily make sense for this project since it is a GEOJSON structure with many variables that are not needed. However, this is something that has to be adjusted by Wapice and given the limited time this was not possible from their side. Therefore, the data that I need to send (Number of people present in a frame) is send under the name "accuracy".

V. Discussion

This project has shown me a number of valuable topics that are of major importance in the field of Embedded Systems. At the beginning three goals were set for me to dive into a bit deeper in order to explore the possibilities in these areas and get familiar with how to use these in a project. The three main topics were about the use of a sensor and its data acquisition, computer vision/image processing, and finally also communication protocols. In the project these three topics have been used in order for me to get more practical experience with all the theories I learned about in the past years. But also fairly new topics that I felt I missed in my subjects have been addressed, such as for example MQTT. Besides the practical experience I also got a first impression on what it's like to work with the support of an actual research group of the university, which I have been looking forward to for a long time.

The first topic, data acquisition from a sensor, kind of surprised me in the fact that it is more difficult than you would think it is to get data from a sensor. Before, I have always done the theoretical work on for example signal processing from sensors but never the actual data acquisition. The need to test the setup often and knowing the inter workings of a sensor exactly have surprised me. Although I feel like it should be possible for this to be easier I did enjoy getting more in-depth.

The second topic, computer vision/image recognition, was good to work on in practice for once. Even though I did not manage to get a concluding technique working for each situation, I did get the opportunity to test out several different techniques such as YOLO, Canny Edge, and Contour. This made me familiar with the logic behind each approach and different ways to implement it. The fascinating thing that this project helped me realize is the tremendous amount of approaches with which one can work on image recognition problems and the pace in which the space is improving.

```
IoT Hub device sending periodic messages
Sending message: {"type":"feature","id":"110AF1844EA0","properties":{"client":"Embassy","accuracy":74,"battery":"89%","color":"black","dateTime":"2022-02-25T18:25:14+01:00"},"geometry":{"type":"Point","coordinates":[78.011578, 34.357068]}}
Message successfully sent
```

Fig. 25. Message successfully sent to IoT Hub

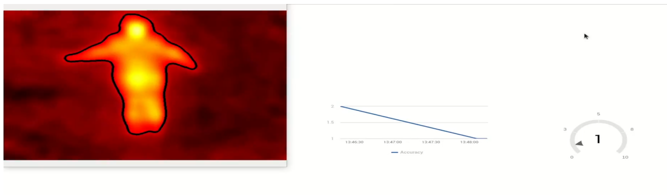

Fig. 26. Dashboard showing the detection of a person

The third topic, the MQTT protocol that is used as the standard approach in IoT communication was also completely new to me. While I did follow some courses in networking I never got the opportunity to actually get hands on experience with it apart from the very basics. For this project the MQTT connection with the Microsoft Azure IoT Hub gave me the possibility to learn about its implementation and also handed the time to get to understand the protocol better.

On the challenge of person detection using a thermophile sensor, a decent solution was created, however, with multiple limitations. Most of these limitations are with regards to the image processing in order to detect people in a frame. Not surprising is that this part is also the most creative part with most uncertainty. The main limitations in this area are with respect to the detection of people in the case multiple of them are present. As already pointed out in the results section it becomes troublesome for the program to detect multiple people once they are intermingled with each other. In order to solve this one could think of relatively simple solutions such as for two people for example identifying a gap within the contour to be indicating two people standing next to each other with hands on each other's shoulder. However, a gap could also be between a body and an arm of the same person. All in all, there is no final solution to this problem, and also not yet found in literature. Many approaches lack in one way or another and with an image without memory (like e.g. NNs could have) it is basically near impossible as far as I have found. With a video, there are more options, think about trace tracking and prediction of that trace for example. For this however, also a PIR sensor would have to be used together with the thermophile one. In other words, in order to make this approach work, more sensors are desired to be integrated in the system.

With PIR a person's movement could perhaps be followed and predicted on the direction someone would move. This way a person's heat contour size can be identified and followed around, knowing that when it suddenly changes there is a chance of interference when that person is for example walking past someone else or behind an obstacle. This brings us to the other challenge and current limitation, which is with regards to obstacles that are in the way. This might be an easier problem on its own since with obstacles in the way there is not another moving heat source meddling with a person. In this case one could for example think of grouping all detected contours within a certain horizontal (x-axis) range together and counting it as one person (when only feet and head are visible, one person would still be counted instead of two). This, however, based on my own tests, did not give sufficient results the moment more people were in the frame. Again, resulting in a problem of which no solution has been found just yet.

VI. Conclusion

Eventually the system is privacy-aware since no approach is used that could possibly identify a person's identity. Normally a thermophile sensor might be able to identify a person, however with a low enough resolution such as this for this sensor (32x24 pixels), it only shows the mere presence of the entity in a frame. As can be seen in the images in the report no privacy rights are violated this way. With this privacy aware approach other challenges do appear (multiple people, objects in sight) which normally would be less troublesome to solve, would privacy not be of any issue. Possible improvements could contain more sensors to also estimate movement within a frame, making it easier to actually track people and perhaps predict their behavior.

The system developed is robust in the ways it can be deployed, both on the ceiling or a wall and whichever angle, all is possible. The deployment does not affect the occupancy detection, although from the ceiling it would be easier since no obstacles can get in the way. On the other hand, on the ceiling the range of the sensor would be shorter as the sensor cannot scan all the way to the back of the room that way. This is a first step towards the deployment of highly flexible, context-aware systems through being device-free, low-power and privacy aware. When people keep enough distance and without obstacles this solution works the best, however, should one need more precise context-awareness systems that go beyond occupancy detection, then more research is expected for just the thermophile sensor or more sensors are to be used.

All in all, this project has given me ample opportunities to experience both in-depth knowledge and the more practical sides that are associated with Embedded Systems. I have been given the time to do a deep-dive on topics such as data acquisition, image processing and the MQTT protocol and have experienced working with a research group first hand. For this, I want to thank both VP and Sujay for the support they have shown and the opportunity they presented me with.

References

[1] K. van Berlo, ”Privacy-aware, Low-power, Occupancy detection of Multiple People using a Thermophile Sensor”, GitHub, 2022. [Online]. Available: https://github.com/KilianBerlo/LOCI. [Accessed: 25- Feb-2022].

[2] E. Commission, ”2018 reform of eu data protection rules”, 2019. [Online]. Available: https://ec.europa.eu/commission/priorities/justice-and-fundamental-rights/data-protection/2018-reform-eu-data-protection-rules_en. [Accessed: 08- Feb- 2022].

[3] F. Zafari, A. Gkelias and K. Leung, ”A Survey of Indoor Localization Systems and Technologies”, IEEE Communications Surveys & Tutorials, vol. 21, no. 3, pp. 2568-2599, 2019. Available: 10.1109/comst.2019.2911558.

[4] C. Álvarez-Aparicio, Á. Guerrero-Higueras, F. Rodríguez-Lera, J. Clavero, F. Rico and V. Matellán, ”People Detection and Tracking Using LIDAR Sensors”, Robotics, vol. 8, no. 3, p. 75, 2019. Available: 10.3390/robotics803007.

[5] S. Narayana, R. Prasad, V. Rao, T. Prabhakar, S. Kowshik and M. Iyer, ”PIR sensors”, Proceedings of the 14th International Conference on Information Processing in Sensor Networks, 2015. Available: 10.1145/2737095.2742561.

[6] Melexis, ”MLX90640 32x24 IR array Datasheet”. no. 12, 2019. [Online]. Available: https://www.melexis.com/-/media/files/documents/data sheets/mlx90640-datasheet-melexis.pdf. [Accessed: 08- Feb- 2022].

[7] ”Sobel Derivatives — OpenCV 2.4.13.7 documentation”, Docs.opencv.org, 2022. [Online]. Available: https://docs.opencv.org/2.4/doc/tutorials/imgproc/imgtrans/sobel_derivatives/sobel_derivatives.html. [Accessed: 08- Feb- 2022].

[8] ”OpenCV: Morphological Transformations”, Docs.opencv.org, 2022. [Online]. Available: https://docs.opencv.org/4.x/d9/d61/tutorial_py_morphological_ops.html. [Accessed: 08- Feb- 2022].

[9] ”Person & Presence Detection”, Heimann Sensor, 2022. [Online]. Available: https://www.heimannsensor.com/person-detection. [Accessed: 08-Feb- 2022].

[10] M. Ivašić-Kos, M. Krišto and M. Pobar, ”Human Detection in Thermal Imaging Using YOLO”, Proceedings of the 2019 5th International Conference on Computer and Technology Applications, 2019. Available: 10.1145/3323933.3324076.

[11] F. Suard, A. Rakotomamonjy, A. Bensrhair and A. Broggi, ”Pedestrian Detection using Infrared images and Histograms of Oriented Gradients”, 2006 IEEE Intelligent Vehicles Symposium. Available: 10.1109/ivs.2006.1689629.

[12] Y. Chen, S. Jhong, G. Li and P. Chen, ”Thermal-Based Pedestrian Detection Using Faster R-CNN and Region Decomposition Branch”, 2019 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS), 2019. Available: 10.1109/ispacs48206.2019.8986298.

[13] J. Song, L. Liu, W. Huang, Y. Li, X. Chen and Z. Zhang, ”Target detection via HSV color model and edge gradient information in infrared and visible image sequences under complicated background”, Optical and Quantum Electronics, vol. 50, no. 4, 2018. Available: 10.1007/s11082-018-1442-z.

[14] S. Narayana et al., ”LOCI: Privacy-aware, Device-free, Low-power Localization of Multiple Persons using IR Sensors”, 2020 19th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN), pp. 121-132, 2020. Available: 10.1109/ipsn48710.2020.00018.

[15] N. Hassan, K. Ming and C. Wah, ”A Comparative Study on HSV-based and Deep Learning-based Object Detection Algorithms for Pedestrian Traffic Light Signal Recognition”, 2020 3rd International Conference on Intelligent Autonomous Systems (ICoIAS), 2020. Available: 10.1109/icoias49312.2020.9081854.

[16] B. Suto et al., ”HSV color space based buoy detection module for autonomous underwater vehicles”, 2015 16th IEEE International Symposium on Computational Intelligence and Informatics (CINTI), 2015. Available: 10.1109/cinti.2015.7382944.

[17] M. Monir Rabby, B. Chowdhury and J. Kim, ”A Modified Canny Edge Detection Algorithm for Fruit Detection & Classification”, 2018 10th International Conference on Electrical and Computer Engineering (ICECE), 2018. Available: 10.1109/icece.2018.8636811.

[18] ”MQTT - The Standard for IoT Messaging”, Mqtt.org, 2022. [Online]. Available: https://mqtt.org/. [Accessed: 15- Feb- 2022].

[19] Melexis, “Melexis/MLX90640-Library: MLX90640 library functions,” GitHub. [Online]. Available: https://github.com/melexis/mlx90640-library. [Accessed: 08- Feb- 2022].