Internet of Things for Room Occupancy Monitoring

Nuno Miguel Monteiro Mano Passada Ferreira

Thesis to obtain the Master of Science Degree in

Electronic and Computer Engineering

Supervisor: Prof. Luís Miguel Veiga Vaz Caldas de Oliveira

Examination Committee

Chairperson: Prof. João Fernando Cardoso da Silva Sequeira
Supervisor: Prof. Luís Miguel Veiga Vaz Caldas de Oliveira
Member of the Committee: Prof. António Manuel Raminhos Cordeiro Grilo

December 2019

Declaration

I declare that this document is an original work of my own authorship and that it fulfills all the requirements of the Code of Conduct and Good Practices of the Universidade de Lisboa.

Acknowledgments

First and foremost I would like to thank my dissertation supervisor, professor Luís Caldas de Oliveira of Instituto Superior Técnico. The professors' feedback has been immensely valuable and his knowledge about many different subjects allowed to complement the work I was developing with some valuable insight, only able to be provided due to the professors' experience. The quick feedback to all the changes in work also helped improved my workflow.

I would also like to thank Carlos Diogo Henriques , for being helpful with all the hardware development plus providing me with the needed tools to develop the needed prototypes. He also suggested new points of view to my work, which helped me turn my work into something more versatile.

Another person that has been valuable in my developments was Dário Silva, helping me with suggestions related to hardware development and providing ideas to solve problems whenever they surfaced.

Everything would not be possible during my 5 years of studies at Instituto Superior Técnico without my colleagues, which I can now call friends, since the constant exchange of ideas and companionship during exam seasons proved to be priceless. I would also to thank those friends whose friendship has been a long lasting issue, because when you have the same friends for more than 10 years, that's when you know you have got friends and support for life.

Finally, I would like to thank my family. My sister, Daniela, for walking this engineering path before me, thus teaching me how to deal with the associated workload and helping me with my needs ever since I entered pre school. For my mother, Alexandra, whose nurturing nature helped soothe me during times where the going got rough. I know I could and can always count on her for knowing what to say and welcome me with her open arms. For my dad, Francisco, whose strict discipline taught me that everything in life can be accomplished if you work hard for it and focus on your goals. For my grandmother, Dulce, for being the sweetest grandmother a grandson could ask for and praying for me every day.

Abstract

Determining the number and location of users inside a building is important information for both efficiency and security reasons. Many approaches can be used for this purpose including movement detection, identification of human bodies or changes in environmental conditions. In this work we used some of these approaches to try to find technologies that provide a good enough estimation of the occupancy of a room, meeting the criteria of being low-cost, non-intrusive and preserving the user's privacy. Therefore, we used vision-based technologies only for controlling the experiments and focused on other types of sensors for the proposed solutions. We started by reviewing of the technologies that can be used to count the number of persons in a room, from which we selected four: Time of Flight (ToF), Thermal Camera, Millimetre-wave Radar (mmWave) and Air Quality, this last one measured by the concentration of Carbon Dioxide and Volatile Organic Compounds.

As a preliminary experiment, we started by testing the ToF sensor and the mmWave Radar to count passing vehicles at a gate of a car park, for which specific hardware and software was developed. The next step was to address the problem of counting the number of persons inside a room. Specific hardware and software were developed for each selected technology that was tested and tuned in different experimental conditions. Finally, the four technologies were tested simultaneously in a real meeting room to compare their performances.

The best results were obtained with the Thermal Camera and the ToF based solutions. We also found a positive correlation between the decrease of Air Quality and the occupancy of a room, in the situation where no doors or windows were opened. The Thermal Sensor showed a greater accuracy than the ToF Sensor. However, the latter excels in cost-effectiveness, being particularly useful in scenarios where a low cost or low complexity solution is needed.

Keywords

Air Quality; People Counting; Occupancy Detection; Time of Flight; Thermal Camera; TI mmWave; IoT

Resumo

Determinar o número e localização de utilizadores dentro de um edifício é informação importante tanto por razões de eficiência como de segurança. Muitos métodos podem ser utilizados para este propósito, incluindo detecção de movimento, identificação de corpos humanos ou mudanças nas condições ambientais. Neste trabalho utilizámos algumas dessas abordagens de modo a encontrar tecnologias que fornecem uma estimação de qualidade suficiente da ocupação de uma sala, que cumpram os critérios de ser uma solução de baixo preço, não intrusiva e que preserva a privacidade do utilizador. Logo, utilizámos tecnologias baseadas na visão apenas como controlo das experiências, focando-nos noutros tipos de sensores para as soluções apresentadas. Começamos por analisar algumas das tecnologias que podem ser utilizadas para contar o número de pessoas numa sala, das quais foram seleccionadas quatro: Time of Flight(ToF), Câmara Térmica, radar milimétrico (mmWave) e Qualidade do Ar, sendo que este último mede a concentração de Dióxido de Carbono e Compostos Orgânicos Voláteis no ar.

Como experiência preliminar, começamos por testar o sensor ToF e o radar mmWave para contar veículos que passam na cancela de um parque para carros, desenvolvendo software e hardware para este propósito. O próximo passo foi abordar o problema de contar pessoas dentro de uma sala. Hardware e Software específico foi desenvolvido para cada tecnologia escolhida que foi testada e afinada em diferentes condições experimentais. Finalmente, as quatro tecnologias foram testadas simultaneamente numa sala de reuniões real para comparar as suas performances.

Os melhores resultados foram obtidos com as soluções baseadas em Câmara Térmica e ToF. Também foi possível encontrar uma correlação positiva entre o decréscimo da Qualidade do Ar e a ocupação de uma sala, no caso em que nenhumas portas ou janelas foram abertas. A Câmara Térmica revela uma maior exactidão que o sensor ToF. Contudo, este último é excelente em cenários onde uma solução de baixo custo ou baixa complexidade é necessária.

Palavras Chave

Qualidade do Ar; Contagem de Pessoas; Detecção de Ocupação; Time of Flight; Câmara Térmica; TI

mmWave; IoT;

Contents

List of Figures

List of Tables

List of Algorithms

Acronyms

IR-UWB Impulse-Radio Ultra-Wideband

IR Infrared

ToF Time of Flight

RGB-D Red Green Blue Depth

FMCW Frequency-modulated continuous-wave radar

ANN Artificial Neural Network

DNN Deep Neural Network

RSS Really Simple Syndication

CWM Continuous Wave Modulation

PM Pulse Modulated

CMOS Complementary metal-oxide-semiconductor

APNG Animated Portable Network Graphics

FOV Field of View

FFT Fast Fourier Transform

TI Texas Instruments

FoV Field of View

RGB Red Green Blue

SNR Signal to Noise Ratio

DSI Departamento Serviços de Informática

API	Application Programming Interface
ADC	Analog to Digital Converter
VOC	Volatile Organic Compound
IoT	Internet of Things

1

Motivation and Problem Definition

Contents

1.1 Motivation

There is currently a growing need to make buildings as intelligent as possible, either for energy efficiency reasons or to make them safer for their users. This requires the knowledge on how the building is used. That is, how many persons are inside the building and where they are. Measuring room occupancy is, thus, an important step to make buildings more intelligent.

1.2 Problem

There was some work previously done about air quality monitoring for occupancy detection, but it lacked validation. So the natural evolution of this work is to seek technologies that offer validation to this hypothesis. To validate this proposal, all it is needed is to have the air quality inside a room, during a limited period, and to know how many people are inside the room. This can be done manually, but such is a daunting task. Counting people in an automatic manner is an active topic in many different areas of research such as biology, medicine, quality control, industrial machine vision processes, because there are such a wide array of applications in which people counting is a necessity. Either for security [1] or efficiency [2], there are many applications that need a reliable and automated way to count people. One of the examples is the fact that with the increased concern about the air quality in enclosed spaces such as metro stations, classrooms, workplace locations, there has been a rising demand in studies that measure the air quality [3–6] (regarding quantities of pollutants) in various enclosed spaces, such as public transportation. But, in order to be able to properly determine and possibly predict the evolution of these values, we need to have reliable data, to be able to create a model for prediction.
There are many different methods for counting people that are being used such as mechanical methods like turnstiles. The major disadvantage of many of the methods is that they limit the flow of people in public areas. However, with the lower cost of technology, we can now use ideas such as Impulse-Radio Ultra-Wideband (IR-UWB) Sensors [7], infrared sensors (IR) [8] , thermal people counters [9], stereo [10], time of flight (ToF) [11], Wi-Fi [12, 13] or just simple Red Green Blue (RGB) Cameras [14].Video recording is highly discouraged nowadays due to security and privacy concerns, therefore we are left with mainly IR, ToF and thermal counters to answer the need of a good estimate of the number of people. But there has been one new development lately available which is the Frequency Modulated Continuous Wave (FMCW) radar technology [15, 16], implemented in the TI mmWave line of products. This technology seems to be a good alternative given that its sensors are able to track multiple people at the same time, due to Doppler, or can be hidden (thus reducing privacy concerns) behind a thin wooden wall or glass without much or any performance degradation (as shown in TI training articles [17], making it easier to implement in public spaces).

1.3 Organization of the Document

This thesis is is organized as follows: first we introduce the State of the Art in Chapter 2. This chapter delves into the current technologies available for people counting . The next chapter, chapter 3, we present ways of testing some of these technologies in experiments that will allow us to measure their performance. The results of those experiments are presented in chapter 4 to select the technologies that will be used. Each of the chosen technologies will have a dedicated chapter: the ToF in Chapter 5, thermal camera in Chapter 6 and TI mmWave in chapter 7. At last, all the sensors will be placed together in the same setting in Chapter 8.

2

State of the Art Review

Contents

In order to properly define a solution for the problem presented, we needed to research what kind of technologies could be used in order to count people.

2.1 Air Quality Monitoring

The air quality monitoring device that we will use in this work has the ability to measure the concentration of Volatile Organic Compounds (VOC) and Carbon Dioxide (CO2). This device was developed in a previous project to detect room occupancy. In this work we will try to see if VOCs and CO2 can also be used to count the number of persons in a room. VOCs are organic chemicals that have high vapor pressure at ordinary room temperature. The high vapor pressure is the result of a low boiling point, which causes large numbers of molecules to evaporate or sublimate from the liquid or solid form of the compound and enter the surrounding air. VOCs have multiple different sources, since they can origin from both human-made and naturally occurring chemical compounds. The second monitoring point is CO2. CO2 is widely known to be the most significant long-lived greenhouse gas in Earth's atmosphere. In this environment we will be observing, the fact that it is produced by people when they breathe, meaning there will probably be a link between higher concentrations of CO2 in a room and how occupied this room is. There are some studies that have proved that it is possible to correlate occupancy with both of these particle (VOCs and CO2) measures. [18, 19]

2.2 Vision Based People Counting

2.2.1 RGB Sensor

Figure 2.1: Example of an RGB Camera

While vision based people counters are not ideal nowadays due to privacy concerns, they can benefit from the low-cost cameras and well tested image processing algorithms of today. Most of the implementations rely on background subtraction with a type of segmentation with k-means [20]. K-means clustering aims to partition n observations into a k number of clusters in which each observation belongs to the cluster with the nearest mean, serving as a prototype of the cluster. To adapt this algorithm to people counting, people are seen as concentrated shapes that can be extracted using k-means clustering, for example.

Background segmentation can be performed [20] on human detection systems by computing the pixel-by-pixel difference between the current frame and the background image, followed by a threshold. Shape detection is a good idea for crowded places, since an overhead position of the camera just needs to detect heads for a more accurate guess. More advanced techniques can be used, depending on the level of detail needed (for example, if there is the need to track the whole body or if the goal is to just count how many people pass through a specific area).

The camera positioning is really important for people counting since it has a large impact in the result quality. For counting purposes, the zenithal positioning (which consists of a camera placed overhead) is the most adequate, since it reduces occlusion between objects, while also offering some advantages such as having a relatively constant size, eliminating the need for calibration, Another major advantage is that it addresses privacy concerns, since this positioning reduces the capture of peoples' faces. These types of methods manage to achieve accuracy of over 90% [14, 20] .

2.2.2 Depth Cameras

Figure 2.2: Kinect combined RGB + Depth images

With the advent of the popularity of depth cameras for video games, such as the Microsoft Kinect device, depth capable cameras have been getting more accessible and are thus a viable choice for identifying people. This kind of technology is usually employed in conjunction with a regular RGB camera [14], because the added information of depth allows to overcome some of the shortcomings of just using a RGB camera. Some of these shortcomings are, as mentioned before, issues caused by changes in lighting, shadows and compound objects.

In [14] a method is proposed that is able to guarantee a high immunity to foreground detection errors and to overcome the problems mentioned before that mostly show up when there are a lot of people in the camera frame. In [14], they also mentioned that using shape recognition in embedded systems is computationally expensive, making it an unviable choice. Tracking is also recognized as a expensive feature. In this study, it is mentioned that using the depth information allows a more accurate detection of the foreground when compared with a standard RGB camera. The false positive rate is zero when depth information is used with a RGB camera, which improves the precision to 0.990 in combination. In every scenario presented with this test, the algorithm when fed depth images in comparison with RGB images, performed better with depth. Placing the RGB camera indoor caused problems referred to be presence of reflections , while placing the RGB camera outdoor reveals that shadows become the major problem. The use of a depth sensor removes these problems but the splitting and/or merging of the blobs cause errors. In terms of performance, using a PC, with images with 640x320 resolution, results can be done with 189FPS and an ARM chip (single core, 0.8GB of ram) managed to do 17FPS, which is almost real time and there are more powerful chips available now.

2.2.3 Stereo Cameras

Using a stereo camera setup [10] that hangs in a zenithal position allows a better segmentation of background and people. This setup also allows for height measurement using trigonometry while allowing for a decent count of people. Unfortunately, the usage of dual cameras and its need for calibration means this setup isn't as sturdy and resistant to change than other proposed methods. Also using two cameras might be more expensive than a single camera, while the added level of complexity could be obtain better results by just adding a Depth sensor to a simple RGB camera.

2.3 WiFi People Counters

WiFi people counters have the advantage of covering more area than regular sensors. Most sensors only work in fixed gateways or check- points, due to their limited area of effect (for example a ToF sensor only works if the person crosses this sensor and a thermal based approach is only able to count the persons in its frame). This limitation is shown to cause low accuracy [12].

On [12] it was possible to achieve a performance of 93% and exhibited robust performance resisting environmental changes with negligible power usage. This paper goes really in depth in the Really Simple Syndication (RSS) studies on humans. This method uses the WiFi connection of smartphones as a counting tool.

Another study [13] used a deep neural network (DNN) with 3 layers and a single WiFi transmitter and receiver to estimate up to 9 people in a room with 88% of accuracy, using signal correlation. This is not ideal for environments that are highly crowded, because correlating both signals will be difficult.

2.4 Time of Flight

Figure 2.3: TeraRanger Evo ToF sensor

According to [11], ToF technology it is a non-contact measuring technique, that has been applied in a wide range of industrial appliances ranging from automatic assembly to quality assurance. These devices can provide depth information. They consist on a modulated light source such as a laser, a Complementary metal–oxide–semiconductor (CMOS) imager made out of an array of pixels and an optical focusing system. Two types of ToF depth sensors exist: the first class being represented by Pulse Modulation(PM) sensors. In this type of sensors the distance is computed directly from the time of flight using a high resolution timer that measures delay between signal emitted and reception. Depth measures are then obtained by:

$$d = \frac{TOF \cdot c}{2} \tag{2.1}$$

Where TOF is the time of flight and c is the speed of light. The second type of sensors are named Continuous Wave Modulation (CWM) sensors. The distance is calculated by the phase of the modulation envelope of the transmitted infrared light as received on a pixel. Using $s(t) = (\sin 2\pi * f_m * t)$ as model for transmitted light, where fm is the modulation frequency. The amount of light r(t) reflected by target is given by

$$r(t) = R \sin{(2\pi f_m t - \theta)} = R \sin{(2\pi f_m (t - \frac{2d}{c}))} \tag{2.2}$$

where θ is the phase shifting when the light returns to a sensor pixel, R is the amplitude of the reflected light and d is the distance between the sensor and the target. Therefore, d can be computed from θ as follows:

$$d = \frac{c\theta}{4\pi f_m} \tag{2.3}$$

PM sensors require a high resolution timer and a large bandwidth signal source to achieve high resolution measures, which means they are usually more expensive than CWM ones and better for long range applications. CWM devices are prone to aliasing problems, which can be diminished using multi frequency scanning.

CWM new counters can get really high accuracy and they work as a kind of thermal camera (with the array of 64x64) to detect people.

ToF sensors have the advantage of being able to improve 3D Stereo vision people tracking [11] because they improve the cases where we have untextured scenes due to homogeneous objects or poor illumination. ToF sensors add some depth based on geometrical constraints and invariants.

An approach based on ToF sensors requires no computation for 3-D scene re-construction and turns out to be independent of the degree of texture and the lighting conditions of the scene.

2.5 Thermal Imaging

Thermal infrared(IR) sensors [9] respond to emitted radiation, more so than reflected radiation. All objects emit heat by either conduction, convection and radiation. Radiation is the most important part for these kind of sensors, because objects continuously radiate heat with certain wavelengths, depending on the temperature of the radiating object and its spectral emissivity . As the object temperature increases, the radiation increases. The radiation emitted includes the infrared emission (which include wavelengths from 0.7 um to 100 um). These emissions are then detected by the thermal imager and made visible as an image in the form of a map of apparent temperatures.

Thermal imaging converts thermal radiation into a digital signal which is in turn converted into a visible image. To reduce cost we will use low resolution 8x8 or 16x16, but via interpolation we can add higher resolution images for improved human perception. This interpolation does not add extra information to the image. Lower resolution images are also easier to be processed and can be done in an embedded system.

Results obtained with standalone thermal Imaging are approximately accurate to 5% according to [9]. While combining both methods (image and using a ANN for the IR imaging) got this percentage down to 3%. The sensor should be applied at entrance to count people in a narrow patch and then add those people as they move inside the room, in order to be able to count large groups of people. The big advantage of thermal imaging is the fact it can function well in low lighting situations, unlike purely vision based systems.

2.6 TI mmWave

2.6.1 Theory

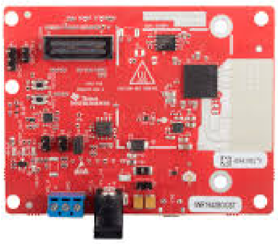


Figure 2.4: TI IWR1642 FMCW Radar

First of all to understand the usage of the TI mmWave Sensors, we need to understand the underlying technology. The sensing technology used is named Frequency Modulated Continuous Waves(FMCW) Radar. These types of radars measure the range, velocity, and angle of arrival of objects in front of it.

A FMCW radar transmits a signal called a ”chirp”. A chirp is a sinusoid whose frequency increases linearly with time. [21]

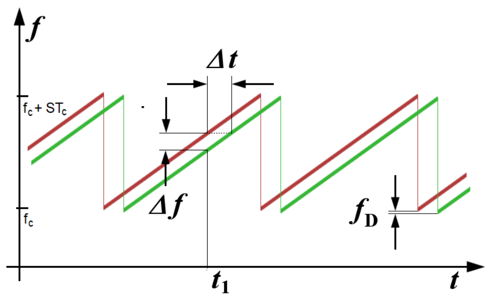


Figure 2.5: Example plot of time against frequency

One of the most convenient ways to represent these chirps are in a frequency vs time plot. A chirp is characterized by start frequency(Fc), Bandwidth(B) and duration(Tc). The slope (S) of the chirp defines the rate at which the chirp ramps up. The unit is Hz/us (aka frequency per second)

Below there are some of the most important equations. An object at a distance d produces an IF frequency of :

$$f_{IF} = S2d/c \tag{2.4}$$

The ADC sampling rate Fs, limits the max range (d_{max}) to :

$$d_{max} = Fs \cdot c/S \tag{2.5}$$

The Chirp Bandwidth and the IF Bandwidth are related. A sinusoid in time domain produces a peak in the frequency domain. In general the Frequency domain is complex (i.e. each value is a phasor with amplitude and phase) which means we have the following: $A \cdot e^{(j\theta)}$

Both the transmitted and received signal, get input into a mixer producing an IF signal with a constant frequency $\frac{S2 \cdot d}{c}$.

A small motion in the object will change the phase of the IF signal but not frequency.

The phase difference measured across two consecutive chirps can also be used to estimate the velocity of the object by:

$$v = \frac{\lambda \cdot \Delta\Phi}{4\pi \cdot Tc} \tag{2.6}$$

Where $\Delta\Phi$ is the phase shift between the two chirps and where Tc corresponds to the time that separates the two chirps.

Multiple objects that are equidistant from the radar, each travelling with different velocities can be distinguished using a Doppler FFT.

In what regards more in depth info of the TI mmWave device, ADC data corresponding to chirps are stored as rows of matrix. Range-FFT on each row resolves the objects in range. After this, Doppler FFT resolves the velocity of each object (i.e. objects at same distance but difference velocity can be distinguished). If they have same velocity but are different, you need more than one antenna to measure the angles of arrival.

Better knowledge of the technology will be needed if it is intended to try and change the clustering that TI already offers to its users, but since the IWR1642 device does most of the processing in the board before giving access to that data, it should not be needed.

2.6.2 Use Cases

Currently Texas Instruments has several demos available using this technology, being the most relevant for the intended application the people counting demo [22] which allows for clustering and point cloud formation in a closed room with great success. It does this by checking what is static (and consists of the background) and what is moving and then, with a correct parameter setting, allows for people counting. Using this and modifying the supplied code to count people after they cross an imaginary line might be a good hypothesis for a private people counter, if we manage to find a positioning of the device that allows for good performance even in crowded situations (e.g. if we want to use the device in the entrance of an auditorium).

	RGB	Depth	RGB + Depth	Stereo	WiFi Counter	ToF	Thermal	TI mmWave
Low-Cost	5	4	3	2	5	3	4	1
Performance	3	4	5	3	3	N/A(3)	4	5(theoretical)
Range	2	2	2	2	3	4	2	5
Complexity	3	3	4	5	3	2	3	4
Robustness	2	3	4	2	2	3	3	5
Availability	5	3	3	1	5	3	2	3
Total	20	19	21	15	21	18	18	**23**

Table 2.1: Comparison of different methods

2.7 Conclusions

In this chapter we overviewed several of the available technologies for people counting purposes. We started with the air quality monitoring , mostly for occupancy detection, since it had already been used in a previous project. Then we moved onto RGB and depth cameras, since they are some of most prevalent technologies for the problem we are trying to solve. Some ideas that do not have such a big representation in products, such as stereo cameras and WiFi Counters were also considered. Finally, we ended on three of the most promising technologies with the ToF, thermal camera and the TI mmWave. According to what we have learned about each of them, Table 2.1 reflects our initial thoughts on how good each of the technologies is, for our intended purpose. This table also helped us pick the technologies we wanted to test. In chapter 3, we will then set up the testing methodology to test the technologies chosen, which were the depth camera, ToF sensor, thermal camera and the TI mmWave.

3

Definition and proposal of methodologies

Contents

3.1 Research Question

The people count problem surfaced since there was a need to correlate various readings, provided by different types of sensors, from closed rooms (e.g. class rooms, conference rooms) with the number of people inside them. This is important because the intent is to be able to model the air quality in relation to the number of people, therefore a good estimate of the number of people inside the room is needed. Like previously mentioned, there are many commercial products available nowadays that already give a count of people passing in a certain area, which are mostly used for advertising or just flow counting purposes in shopping centres or enclosed spaces. Those solutions usually present some form of limitation and with this work we are trying to find a suitable solution to those. One limitation, for example, is that many of these system are vision based, which means we need to have a camera pointing to people in order to correctly count the people flow. But with a product like the TI mmWave, theoretically, it is possible to put this device behind a plywood wall and still have good counting results. **This work aims at answering the following research question: how good is the estimate of a low cost, non intrusive solution for occupancy detection, that can work together with indirect occupancy measures such as the air quality?**

3.2 Experiments

To answer the proposed research question, different tests will need to be done, mostly regarding the density of people in the range of our sensors. It is expected for the sensors to perform great in sparsely populated areas, but with the main application goal being auditoriums we need for the sensors to perform just as well in very densely populated areas. Other possible outcome is to expand our counter to not just people, but vehicles as well (if a good counting algorithm is developed, it should work for both, since an algorithm that's robust for people counting in crowded environments, should have a good overall performance).

Since we need to validate how good the estimate of each method is, all the sensors will need to be tested in the same environment, to have comparable results. The use of conference rooms in Tecnicos' Alameda campus was the obvious choice since they have a constant people flow, plus an ease of access to service anything about the system, should such a need surface.

As for the car counting implementation, for the full systems, once again the choice of the Alameda campus from Tecnico seems to be the best option. Having the information about the actual value of cars from Tecnicos' Informatics department, plus the general proximity to the system to ensure the experiments were working correctly. In this case, the sensors will be counting the cars that exit the campus.

3.3 Technologies

From the preliminary research , it seems that the technologies that are more promising are the TI mmWave Radar alongside the RGB + Depth Camera or a ToF sensor for a preliminary count. The intent will be to try and get an accurate reading for number of people in a room to be able to correlate air quality readings (such as temperature, humidity, CO2...) with the number of people inside a room and, in the future, maybe automate the air quality control systems (such air conditioning systems). Starting with the systems to be placed outdoors, we will focus mainly on two different approaches: the ToF based one, which we will use with the help of a Raspberry Pi 3B+, since the sensor just needs an I2C connection to interface with the Raspberry. The sensor uses the 5V line from the Raspberry Pi to be powered. The Raspberry itself uses a standard microUSB power supply that is capable of offering 5V at 2.5A. Using the TI mmWave, on the other hand, is a bit more demanding to the equipment. Since there is a need to run Matlab, and in real time with pointclouds being feed into the system, the computer used with this radar should be at least a quad core system, with 8GB of Ram. To power the mmWave, a power supply of 5V and 3A with 5.5x2.1mm barrel Jack is needed.

For counting people indoors, the mmWave setup will be similar, the only difference being the Matlab Demo that is running on the computer.The ToF demo is being run on an Arduino Uno, interfacing with the VL53L1X via I2c. The counting will be done in Arduino which will send the data via Serial to Raspberry Pi Zero W, which in turn will be connected to the internet posting the findings on a website. The Pi Zero W can be powered via a simple microUSB 5V 1A power supply and the Arduino Uno can be powered of a similar power supply. For the thermal system, we will be using a Raspberry Pi 3B, which will be interfacing with the MLX90640 sensor via I2C and a C library. The visible camera will be plugged into the Raspberry Pi via flat ribbon cable.

3.4 Conclusion

In this chapter we set out the question that will define this paper, which is the possibility of having a low cost, non intrusive solution for occupancy detection. We then outlined the experiments that will be conducted in a final phase, alongside the system requirements, for the full developed systems. In the next chapter, we try to choose which of the technologies we presented in chapter 2 are the ones to use in our use cases.

4

Preliminary Results

Contents

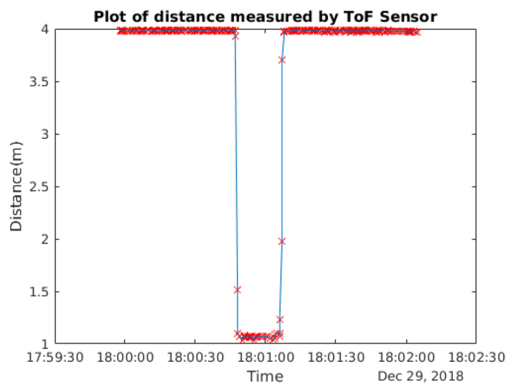
The first room used for our experiments was the GA1 amphitheatre, one of the main amphitheatres in Instituto Superior Técnico, with a seating capacity of 130 people. This means that, if we have a 90% accuracy with any of our counting sensors we should get a count of at least 117 correctly detected people. With the use of the technology mentioned before it should be possible to obtain this goal even for some crowded areas, especially if the settings on the mmWave radar are correctly chosen. Depth camera results would also be interesting to obtain since on some paper references they managed to obtain really high accuracy using just a depth image.

4.1 ToF Sensor + Camera for GroundTruth

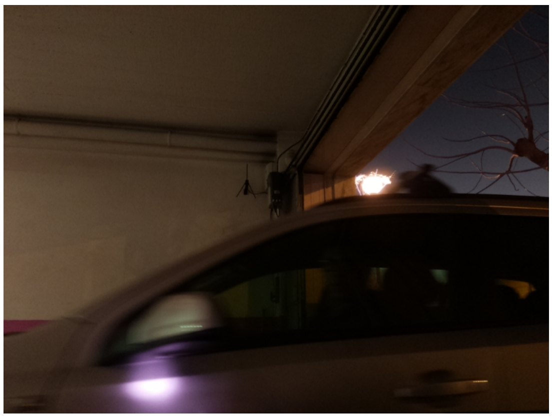
One of the first setups to be tested will be to use a TeraBee TR EVO 60M to determine the passage of big objects, such as cars through gates in some locations in the Tecnico Alameda Campus. We decided to implement this system with a simple ToF sensor because the units being measured are vehicles, which are easier to be measured by this type of system (which gives us a single distance reading). The rest of the testing setup will be to implement a simple shutter system that will trigger a picture sent by FTP when the sensor detects a car (this allows to check if our hypothesis is correct).

4.1.1 Preliminary Data

The version of the car detector and counter used here determines if the thing being measured is a car if the distance being measured is less than half the background distance and we have 3 of those measurements (since a person or a bad reading could trigger a car count). The picture is also stored locally, but will be stored remotely in the full version.



(a) Plot of the distances(red crosses representing the separate data points)



(b) Car picture taken upon crossing

Figure 4.1: TR-EVO Preliminary Data

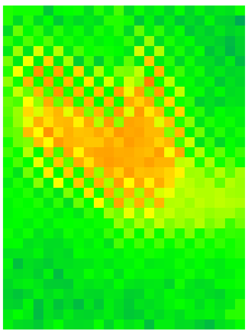
While observing Figure 4.1(b), we can notice a small circle of light in the lower left corner which corresponds to the IR light being emitted by the TR-EVO Device (which is giving us the distance reading). For further improvements, we need to force a refresh rate (currently it does at least 2Hz, but sometimes can go up to 3 readings per second). Also, depending on the location of setup some adjustment to the parameters might be needed. Ideally the sensor and camera position will also be overhead.

4.2 Thermal Camera

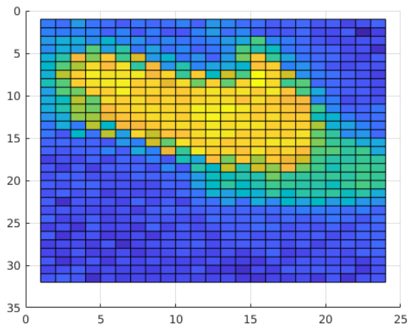
Thermal Cameras are being used everyday more and more, due to their availability to the general public and due to them becoming more accessible. Temperature is one of the ways it is possible to count people and using one of the many available sensors plus a Raspberry Pi for some processing it should be possible to get some decent results.

4.2.1 Preliminary Data

The main connections used in the preliminary test setup were a Raspberry Pi Zero W with a Melexis MLX90640 thermal camera in a breakout board interfacing via I2C. The drivers to use the thermal camera still need some improvement, but it was possible to obtain some results.



(a) Video frame of Thermal Camera)



(b) Plot of the temperatures recorded in Matlab **UI!**

Figure 4.2: Thermal image representation

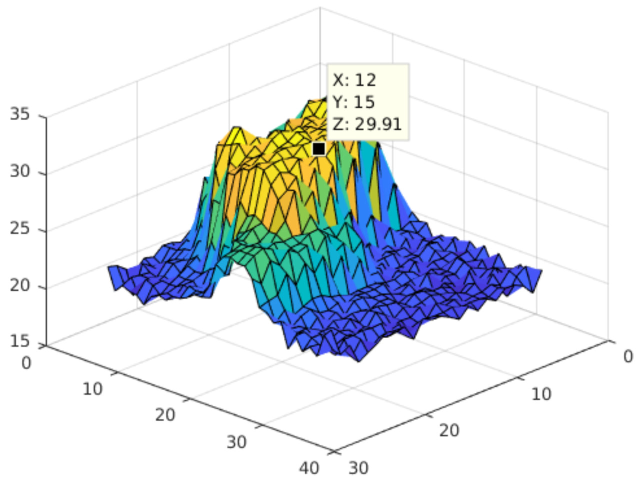
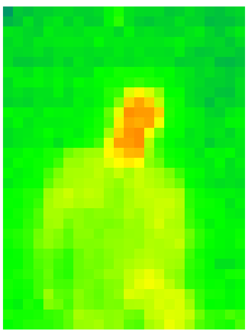
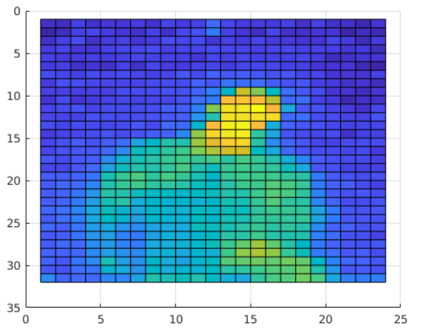


Figure 4.3: Surface plot of temperatures

These are the results that were obtained when an hand was positioned about 20-30cm to the sensor. The video frame rate was of 16 frames per second, but there was some checkered patterns when the motion was fast that will need addressing in the future. But since the main usage will be heads, the next data set was obtained about 60cm from the sensor and features a human head.



(a) Video frame of Thermal Camera)



(b) Plot of the temperatures recorded in Matlab **UI!**

Figure 4.4: Thermal image representation

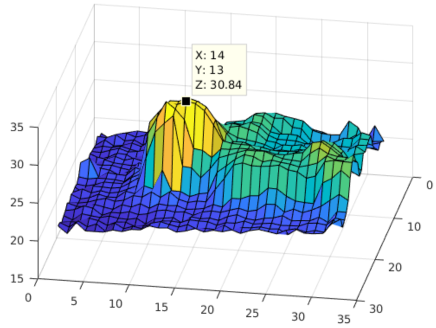


Figure 4.5: Surface plot of temperatures

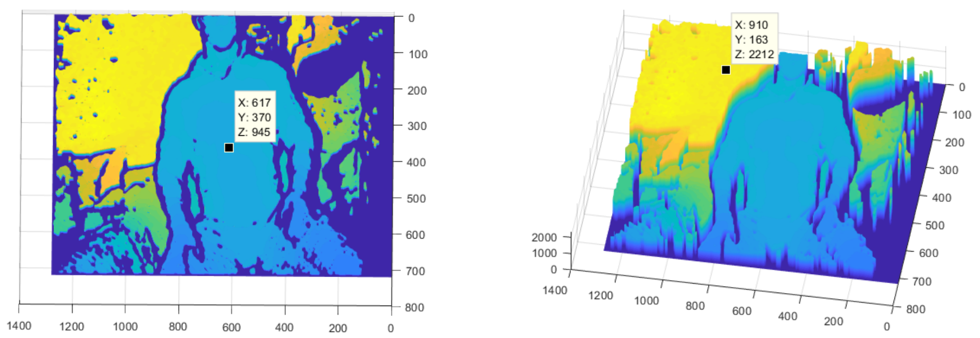
4.3 Intel Realsense (RGB + Depth)

The Intel Realsense camera combines a RGB sensor plus a Depth image sensor [23], in what is basically a more advanced version of the most popular and well known Kinect from Microsoft. This camera has better features in terms of resolution (that may or not matter, since if we have limited processing power, downsampling might be a necessity for a real-time system), where we can count with active stereo depth image of 1280x720. The main difference between this system and the mentioned Kinect is how the Depth Image is obtained. Like previously mentioned, the Intel system has a pair of Stereo Cameras alongside a texture projector that results in a depth image. The Kinect has a depth sensor that possesses a monochrome CMOS sensor and infrared projector that help create the 3D image. The depth image distance are also measured by transmitting IR waves and measuring its ”time of flight” after it reflects off the objects. The method Intel developed should be more robust, thus having a better depth image to develop the needed people counting algorithm.

The preliminary data will be obtained through two different Realsense models, the D415 and the D435. Both models function through USB3 to provide RGB-D data with a maximum depth image resolution of 1280x720. But the main differences between the two are the FOV(field of view) and shutter type. While the Realsense D415 has an approximately FOV of 70 degrees, the D435 has a wider FOV of around 90 degrees, which means we will have to choose the better camera for the pretended use case. Another thing that might be important is that while the D415 is an all in one module, the D435 is based on a Realsense module with an attached but separate RGB camera, which might make difficult to mix the RGB and Depth images, if needed. The bigger FOV also implies that the D435 has a minimum distance for detection than the D415 camera.

4.3.1 Preliminary Data

To start, the first images were obtained with the Realsense D415 at approximately 90-100cm of distance from the camera and processed via the Rosbag capabilities of Matlab. The results can be seen below:

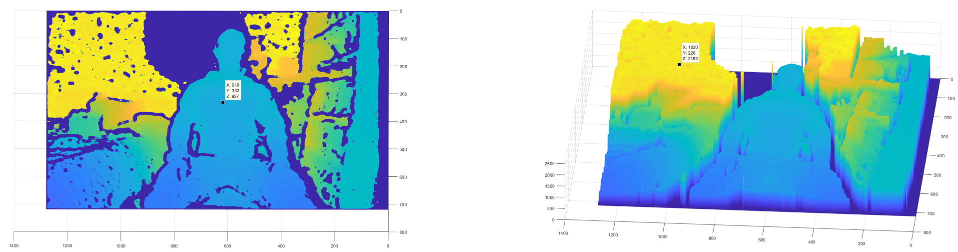


(a) Image frame in just the XY plane

(b) Same image but possible to analyze depth

Figure 4.6: Data obtained via Realsense D415

To try and maintain some consistency with the data, the same video was captured but now with the D435 camera:



(a) Image frame in just the XY plane

(b) Same image but possible to analyze depth

Figure 4.7: Data obtained via Realsense D435

As it is possible to observe from both pictures, the D435 provides a image with more content in it (due to its wider FOV) but loses in detail and accuracy. The wider FOV might be useful to cover a wider area while positioning these cameras overhead, which should reduce the need of cameras (Intel mentions that pure raw accuracy might not be needed in case of just people detecting systems).

Both these pictures were obtained using the High Accuracy preset of the Intel Realsense Viewer, but the higher degree of customization of these cameras is also a plus in their usage (alongside the resolution, since we get a depth image of 1280x720).

4.4 TI mmWave

One of the other suitable candidates for implementing an algorithm is the TI mmWave which seemed to have the most promising technology, specially due to the fact that we can put this sensor behind a thin wall, maintaining some of the performance. There is also a lack of products developed using this technology, which means there is more interest in trying to develop a product that uses it. The fact that we have signal processing in the chip that we are going to be using means that most of the DSP processing is already done. Thankfully, TI provides some demos that illustrate the capabilities of this product, which will be the base of our work, most specifically the algorithm will be developed on top of the people counting demo, which already clusters the points that represent people. The main goal of the new algorithm will be to check if the points cross a specific line that indicates that they are entering or exiting the room.

4.4.1 Preliminary Data

The data was obtained using the people counting demo from TI, but for future use, an adaptation of the traffic monitoring demo might be useful (using only a single track to see points that move in and out of frame).

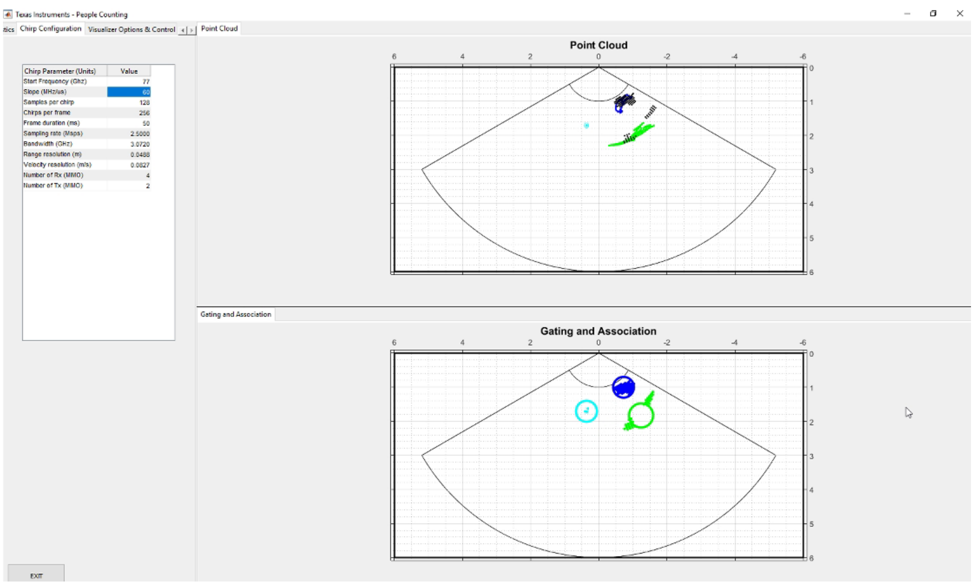


Figure 4.8: TI mmWave People Counting Demo

This dataset was picked because it showcases how important the correct chirp configuration is. In the obtained dataset, there was only one person in the room, yet we detect 3 different clusters, mostly due to reflections and improper SNR settings. Some of the possible ideas to reduce the ”ghosts” issue are to increase the number of points that form a cluster, or to increase the SNR cutoff (it might be too low, causing these reflections to be considered people).

4.5 MVP

After establishing which one of the technologies has the better results with the implemented algorithm, we need to define what is the Minimum Viable Product (MVP), which is the smallest thing we can build that gives the customer value. This means we have to establish a platform for the user to be able to interact with the counting product developed. This means we need to see the points that give value to what was developed, according to the technology chosen (e.g. for the TI mmWave, the fact it doesn't have to be in open space to work). The estimated minimum MVP, will be a counting product that works with a website in which the user can interact and see the occupation along time, locations, or maybe even proceed to do some action when we exceed a certain amount of users in the desired location.

To conclude, the absolute minimum MVP would be a working people counter, if it works using a technology that isn't already widely used, and a people counter plus user interface if the results aren't as satisfactory.

4.6 Conclusion

In this chapter we gathered data using some of the sensors from chapter 2, such as the ToF sensor, the Thermal Camera, Intel Realsense (which is a RGB-D camera) and the TI mmWave. Of all these technologies we will end up not using the RGB-D camera due to the processing power needed (depth images with 1280x720p are resource heavy), alongside the fact that Depth images with such resolution can also be used to find details or identify the people in frame. One could argue that the mmWave is also resource heavy (which it is, due to the need of being used with a computer), but the novelty and promise of the technology made us want to evaluate the sensors performance. And for the ToF and thermal camera, we felt that these are the technologies that made more sense in an IoT perspective, due to the relative low complexity and low processing power needed to implement.

In the next chapter we will begin the development of the ToF solutions, both for car counting and people counting purposes.

5

Time of Flight

Contents

After some preliminary tests in a garage, we installed a ToF-based system in one of the main entrances of Instituto Superior Técnico Campus in Alameda. The data was collected from 1st of July to the 12th with some of the days not having full reading measurements. Later, another option showed up, which allowed us to count people using a single ToF sensor, due to a new release from ST-Electronics, the VL53l1x.

5.1 Car Counting

5.1.1 Setup Architecture

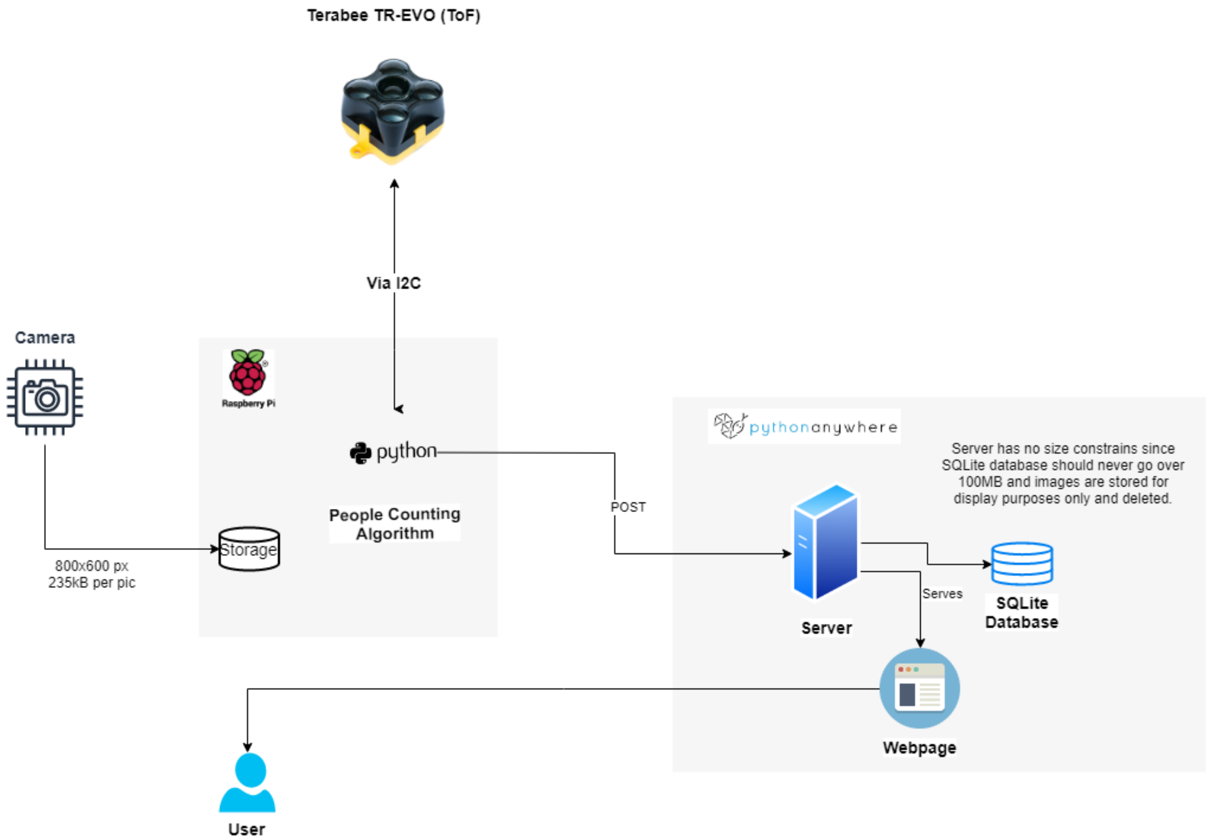
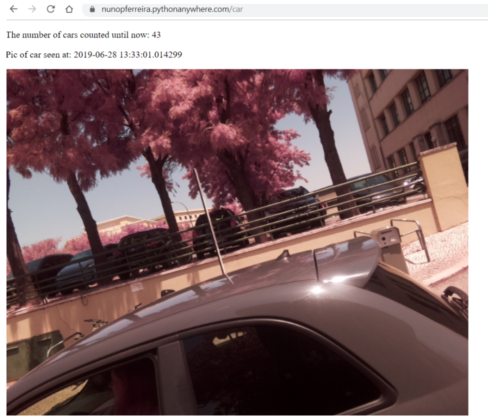


Figure 5.1: TR-EVO System Architecture

A picture of the setup can be shown in Figures 5.1 and 5.2(a). The ToF sensor had to be installed outside the security booth to avoid interference from the glass during readings. Even with this positioning, sometimes the readings without a car passing by would not correspond to the real value, due to sunlight in the sensor. These incorrect readings do not matter for the counting system, since due to them having a really high value, ignored by the algorithm.

(a) TR Evo Installation



(b) Webpage

Figure 5.2: TR-EVO System Components

Now back to the architecture, the system is composed by a TeraBee TR-EVO ToF sensor which is connected via I2C to a Raspberry Pi that is running a Python script for people counting. This Raspberry also has a camera that is activated when the Python script considers there is a car crossing, to serve as a GroundTruth. This Python script also interacts with a cloud service hosted in PythonAnywhere, where we display a webpage (as seen in Figure 5.2(b) that contains the latest car crossing with a timestamp).

5.1.2 Algorithm

Algorithm 1 ToF Sensor Algorithm

```
procedure PROCESS DISTANCE READINGS
    distance ← readi2c(trevo)
    if distance >= 0.5 then
        if distance < backgroundavg/2 then
            car = car + 1
            if (car > 2)&&(pic == 1) then
                take_upload_picture()
                pic ← 0
                total_cars ← total_cars + 1
    else if distance < backgroundavg/1.3 then
        car ← 0
        pic ← 1
    if car == 0 then
        reads ← reads + 1
        total ← total + distance
        backgroundavg ← total/reads
```

For the Teraranger TR-EVO, a python library was created for easier interfacing with the sensor. The features of it are to read from the device and to change the I2C address of it. In terms of algorithm, it is a pretty simple algorithm with a certain number of flags that allow the Raspberry Pi to know whether it should count the reading to the background distance, or if the distance reading corresponds to the sensor being blocked. Regarding the algorithm, all the variables start with the value 0, and the portion of code shown corresponds to the loop that runs inside the program. The algorithm starts with the reading of the current value of the sensor. After this, we need to figure out if the sensor is reading an actual value (this means a value of 0.5m or more, since the minimum distance for this sensor is the 0.5m range). If that is the case, we then proceed to two different cases: if the distance read is less than half of the current background value (line 4), we increase the number of counts that the sensor has been blocked (we can't consider a pass with a single sensor block, because this would trigger a lot of false positives). In case this increase drives the variable to a count number of 3 and the flag to take a picture is active (line 6) then we register the count and upload the picture to the webserver, while making the picture flag zero, and increase the count of the total number of people counted. Line 10 of the algorithm serves to see if the person has crossed the sensor, making the value measured closer to its default background value. If that is the case, the count variable is set to 0 and the picture flag is back to 1 (meaning it can now take another pic when crossed).

Line 13 is the last statement, that means if the counter has been set (or its current value is 0), we update the current *backgroundavg* variable, via total and reads variable.

This algorithm is backed up by a website that serves as display for the last picture taken (with the timestamp of it) alongside the current count of people.

Unfortunately with the current sensor, since it gives us a single distance reading, it is impossible to discern if the people being counted is leaving or entering the room, therefore this system either needs at least two sensors (to know which one is triggered first) or needs a different application. The different application suggested is to use this sensor to count car entrances in one of the side doors of the university, since the sensor does well with big objects (that should trigger less false positives) and there is a road specifically for entering and another one for leaving, and the results shown in the next section are for cars leaving the University.

5.1.3 Results

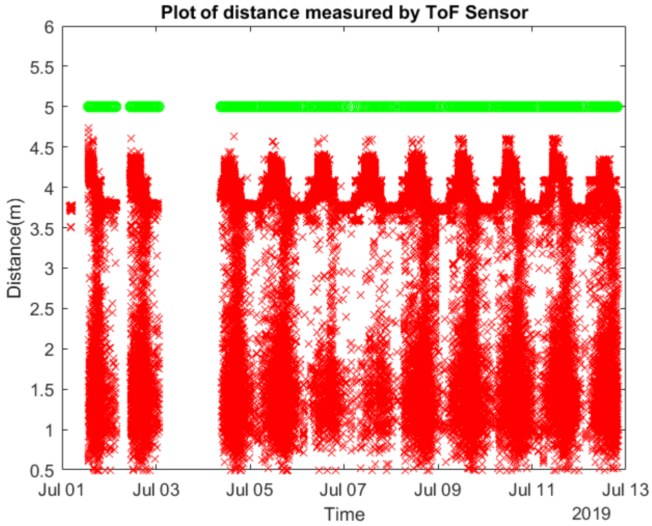


Figure 5.3: Counting of cars leaving Técnico

The values obtained for all the days can be seen in the Figure 5.3 above, where the green dots represent the cars detected by the system and the red markers the distance readings by the ToF sensor. Since this is an hard way to evaluate performance, we decided to pick one of the days and cross the data obtained via the ToF sensor with the card readings of the exit gate in order to have a performance metric of this system.

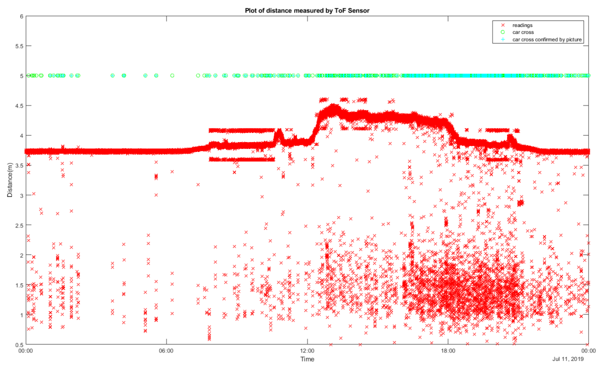


Figure 5.4: 11th July Data

In order to single out one of the days measured, in order to have more specific data, 11th of July was the day picked seen on Figure 5.4. Noted on the graphic is also the photographs that we have confirmed of vehicles, because sometimes the system denoted a car crossing, but the camera did not connect or take the picture.

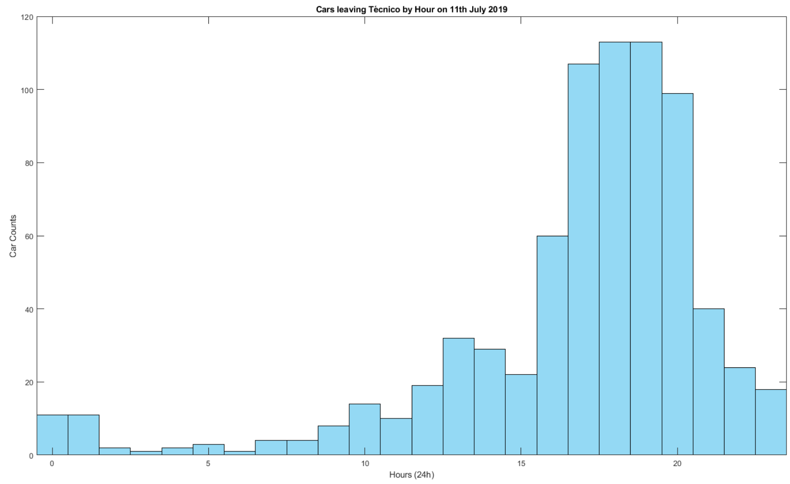


Figure 5.5: Histogram of 11th July Data

The histogram on Figure 5.5 was also plotted to show which hours of the day had more traffic and clearly they show that later on the day has the most activity regarding exiting the university, for a total of 747 cars during the day of 11th of July.

All this data needs to be compared with the system that controls the entrance of the campus. With the help of DSI (Tecnico's information technology services), the number registered by the system was 854 for the full day of the 11th of July. The data for the ToF, had to be time synchronized with the system from DSI.

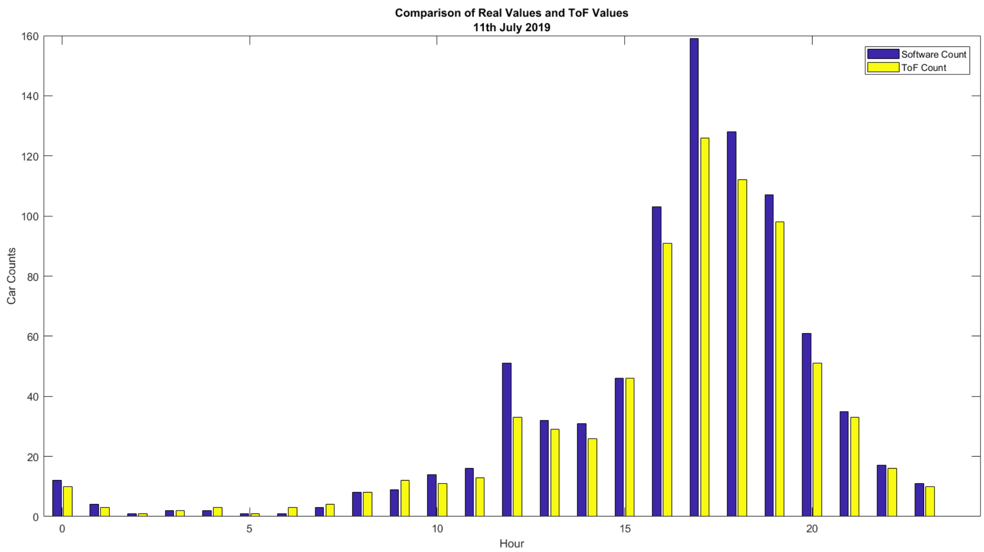


Figure 5.6: Software data

As shown in Figure 5.6, the ToF follows mostly the access control system's data, but we also got some photographs that showed cars that were registered but not registered in the official system. The accuracy obtained by the ToF solution was 86.69%, which is an acceptable performance, but it would have been expected a better outcome. As it was expected most of the time what happens is that the ToF counts less occurrences than the Software, but there are some anomalies in low count cases, where the ToF counted more than the Software. This has raised some questions, so we turned into the validation pictures to assess whether this was a mistake or an actual good count.

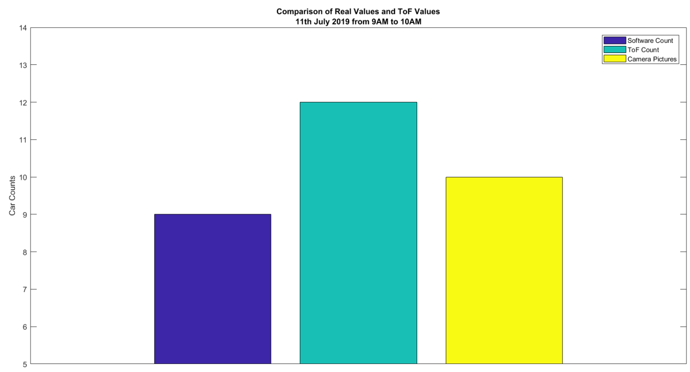


Figure 5.7: Comparison between values obtained via Software, Camera and ToF Sensor

Using the 9AM to 10AM bin, we can see that we have an extra count from the camera when comparing with the Software count, and we in fact have all the 10 images of the crossings. But a close observation of the timestamps, shows that the software has a timestamp with '11-Jul-2019 10:00:00' and the Camera with '11-Jul-2019 09:59:31', which means that the difference is just a matter of second, which could be due to different clocks in both systems, making this a non issue.

In conclusion, this system proved to have a satisfactory performance when there is the need to know the approximate flux of cars (with ideally two time of flight sensors in a single Raspberry Pi, in order to coordinate both entrances and exists, in case there are two separate paths for each). A good example of an install, would be the use of this sensor in the entrance of the garage of a private building, in order to have some estimate of how much the people that live in it use the garage, and when. It would be a simple install and affordable to do so, since this setup can easily be done with a Raspberry Pi Zero W and a ToF sensor such as the VL53l1x, since the Terabee Evo used in here had amazing range and performance but the extra range is not needed in this situation. With its low cost, it is something every condominium administrator could add for extra information about the use of its premises.

5.1.4 Bill of Materials

In order to evaluate the system, a bill of materials of the devices used is necessary. More so when we are trying to evaluate how good a low-cost solution is. Therefore, two bill of materials are proposed for each system, the first one being the components used during testing and the second one being replacements that could be used when applying the system in a real world environment (where for the car counting system, a camera is not necessary for the validation).

Part Name	Quantity	Type	Price	Link
DFRobot Raspberry Pi 3 Model B+ Starter Kit	1	SBC Kit	90.68 €	https://tinyurl.com/y2yetdl5
TR-EVO-60M-EVAL	1	ToF Sensor	141.10 €	https://tinyurl.com/yyarnc66
Raspberry Pi Camera Board v2 - 8 Megapixels	1	Camera	35.05 €	https://tinyurl.com/yyvgapt5
Total			266.83 €	

Table 5.1: Bill of Materials for the Materials Used in testing

Part Name	Quantity	Type	Price	Link
RPI Zero W Camera Pack	1	SBC	57.86 €	https://tinyurl.com/y3953kcy
VL53L1X-SATEL	1	ToF Sensor	17.32 €	https://tinyurl.com/y3qp64sp
VEL05US050-UK-UB	1	Wall Mount Ac Adapter	5.85 €	https://tinyurl.com/yxqwxos7
Kingston microSDHC C10 32GB	1	MicroSD Card	5.00 €	https://tinyurl.com/yyscahd4
Total			86.03€	

Table 5.2: Bill of Materials for the bare product

The first table 5.1, represents the system used, where some of the parts were not acquired, such as the TR-EVO which was already available in the parts bin. That's why we used it, since the performance it gives us, 60M of range, was not needed in this case. But overall, 260€ for a prototype is not a bad value. But this value can be easily improved. On table 5.2, we now choose a less powerful version of the Raspberry Pi, but, that for this kind of processing, works perfectly. We also save on using the VL53L1X ToF sensors, and we buy them in a package that contains two (a version with a single sensor in a breakout board is not available). The price could be even lower if we did not want a starter kit with camera, but the camera helps during the configuration setting of the ToF, making adjusting the performance much easier, for around 20€ more.

5.2 ToF People Counter

5.2.1 Setup Architecture

A picture of the setup can be shown in Figures 5.8 and 5.9. Starting with Figure 5.8, it can be broken down into two hardware systems and a cloud server. The interaction with the VL53l1x sensor is made via I2C with an Arduino. This Arduino counts the entrances and exits with an algorithm and sends the output of the count via Serial Port to a Raspberry Pi Zero W, that publishes these values to a website, alongside storing the values in an SQLite Database. This Database is also stored server side for redundancy and

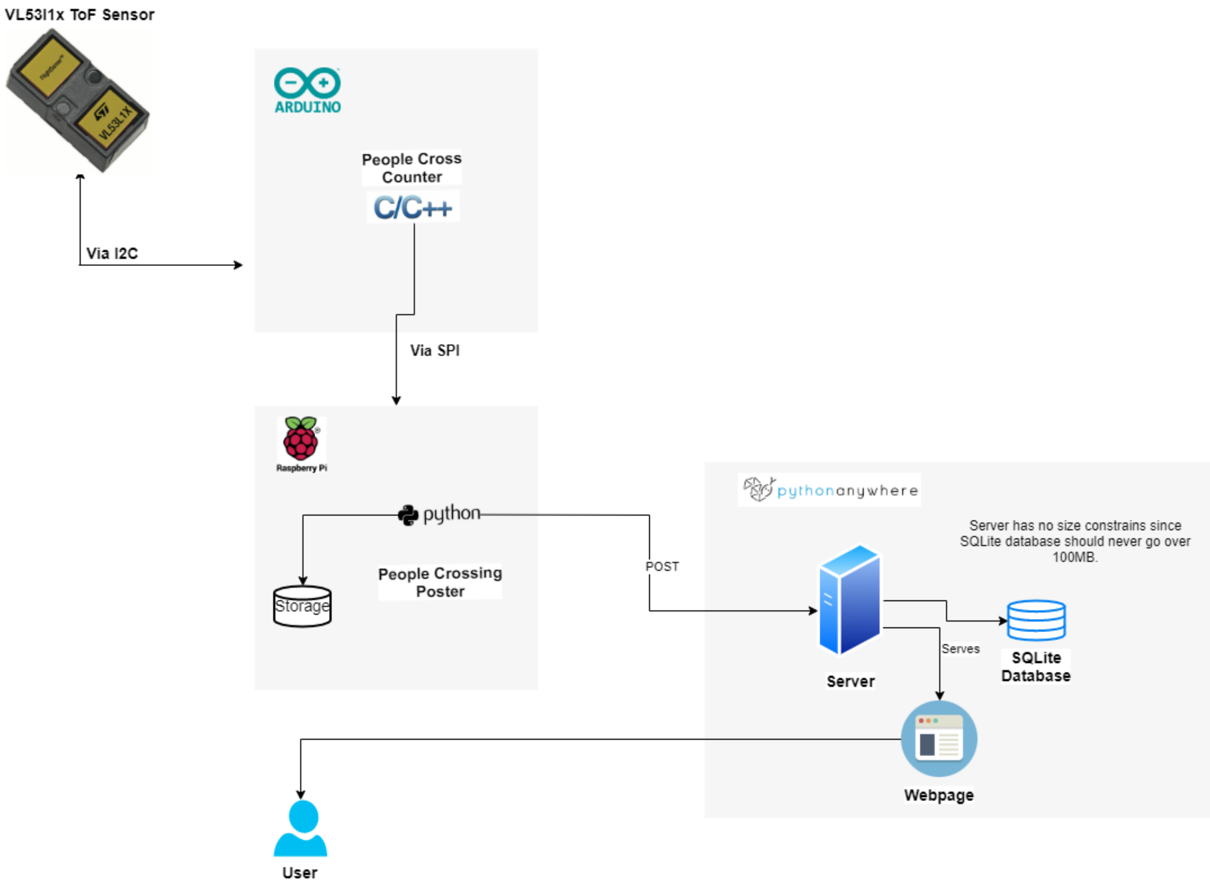


Figure 5.8: TR-EVO System Architecture

to make possible the plotting of the results. The website displays the current count of people inside the room and the timestamp of latest crossing. On Figure 5.9 we can observe the installed system. The positioning of it in the middle of the door is crucial, since the cone of view of the ToF sensor is very sensitive (bad positioning could make that people passing in the opposite corner of the door would not be detected).

For interfacing with the VL53l1x, the brand (STMicroelectronics) provides a low level API and there is also an Arduino Library. Unfortunately, to properly count people, we need the capability of changing the Region of Interest (ROI) on the fly, which was not available by default in the Arduino Library. So, with some modifications, we used the low level API provided to get the desired functionality in the Arduino Library. Other than that, it is a pretty simple algorithm, inspired by the demo that STMicroelectronics have on their YouTube channel that is made to work on their Nucleo Boards.

The counting algorithm relies on a list of states that have to occur in a certain order to detect if a person has crossed the specified area and in which direction this area has been crossed. These states are stored in a list and compared to two default lists of states that represent how the area is crossed in two different directions. When no-one is seen in either of the two zones, the list of states is reset. If

Figure 5.9: Installed System

we consider that a person detected in the front zone equals 2, and a person detected in the back zone equals 1, the algorithm adds the value of the two states and stores the result as soon as it changes. Eventually, if the consecutive states in the list are 0, 1, 3, 2, 0 or 0, 2, 3, 1, 0 this means a person has been detected in one direction or the other, as described in the figures below.

The algorithm validates a crossing event only when a person has fully crossed the two zones. It does not validate the event when the person remains for a long time under the FoV or when the person decides to return from the place he came from.

Algorithm 2 is the one being used in the VL53l1X sensor. This is one iteration of the algorithm which reads the distance for one of the zones, since the algorithm consists in reading both zones determined by the ROI being set, in an alternate way. The first step is to set the ROI of the current zone being analysed(either the left or the right zone). With this ROI, now we get the distance reading from the current zone. This distance is then evaluated to see if it has crossed the desired threshold (the distance reading that we consider to be a person). If that is the case, the status flag Someone Detected is set to 1. After, in case this flag is active, we need to check whether this was a hit on the left or the right zone. If the zone that was detected was the left zone, and the current zone status is that someone was detected in that zone, the Status for all the Zones is incremented to 1. This is a control variable that can have values ranging from 0 to 3. When 0, it means that no one is being detected in all the zones. When 1, it means that there is someone detected on the left zone. When 2, it means detection on the right zone. For 3, it means there is detection in both zones at the same time. Going back to line 2, if the previous status of the right zone was of detection of a person, then we add 2 to the overall status counter. The distinction between Right and Left is needed due to the algorithm running the same for both zones, but it runs a certain part if the zone being analysed is a certain one. That's why the same line of thought is done for the line 2 and below. If someone was detected, this algorithm also puts the Event Occurred flag to 1, which then triggers a line of events. In line 2, we resize the path filling size variable, which only happens if its size is smaller than 4, since only when it reaches 4, the person passage is analysed. The

Algorithm 2 ToF People Counting Algorithm

```
1: procedure PEOPLE COUNTING TOF
2:     Set_ROI_Center(zone)
3:     distance ← vl53l1x_read_distance()
4:     if distance < Threshold_Distance then
5:         Someone_Detected = 1
6:     if Someone_Detected == 1 then
7:         detected = check_zone_detection()
8:         if detected == LEFT then
9:             if Current_zone_Status == Someone_Detected then
10:                All_zone_Status += 1
11:            if Right_zone_prev_Status == Someone_Detected then
12:                All_zone_Status += 2
13:            Left_zone_prev_Status == Someone_Detected
14:        else if detected == RIGHT then
15:            if Current_zone_Status == Someone_Detected then
16:                All_zone_Status += 2
17:            if Left_zone_prev_Status == Someone_Detected then
18:                All_zone_Status += 1
19:            Right_zone_prev_Status == Someone_Detected
20:        Event_Ocurred = 1
21:    if Event_Ocurred == 1 then
22:        Resize_Path_Filling()
23:        if Right_zone_prev_Status == empty && Left_zone_prev_Status == empty then
24:            if Path_Track_Filling_Size == 4 then
25:                if Check_Entry() then
26:                    PeopleCount = PeopleCount + 1
27:                else if Check_exit then
28:                    PeopleCount = PeopleCount - 1
29:            else
30:                Update_Path_Track(All_zone_Status)
31:    Zone = Zone + 1
32:    Zone = Zone%2
```

following line, shows that in case both right and left zones previous statuses are considered empty, the algorithm checks whether there was an entrance or leaving of the room, depending on the Path Track format. The path track is the history of the sensor activation. For example, a path track of 0 1 3 2 counts as an entrance and increase in people counting, since the history of the sensor was an activation of the left zone, activation in both zones, and then finally activation only in the right zone. But, as mentioned in line 2, the path track is only updated if we do not reach the path track filling size of 4. The algorithm ends with switching zones to do the same, all over again for the next zone.

Although preliminary results with sparse usage (just one or two people crossing the sensor) revealed to work properly, performance of the sensor and algorithm in a real world situation is not yet displayed, but can be seen in chapter 8.

5.2.2 Bill of Materials

Part Name	Quantity	Type	Price	Link
Adafruit Raspberry Pi Zero W Budget Pack	1	SBC Kit	34.32 €	https://tinyurl.com/y62szzmn
Arduino Uno	1	Microcontroller	19.80 €	https://tinyurl.com/y5cxkwoo
VL53L1X-SATEL	1	ToF Sensor	17.32 €	https://tinyurl.com/y3qp64sp
USB Cable Type B	1	USB Cable	2.60	https://tinyurl.com/yxu3l2aj
Total			74.04€	

Table 5.3: Bill of Materials for the Materials Used in testing

In table 5.3, we can see the full price of the testing prototype for the ToF people counter. For around 70€ it is possible to have a small system with the VL53l1X sensor, and since the Raspberry Pi is mostly for web interfacing, the Pi Zero W works flawlessly. Since this is already a pretty cost effective option, the prototype could be sold as is, no more cutting costs would be needed.

5.3 Conclusion

In this chapter we developed two different products based on TOF technology. The first one allowed us to count cars with an accuracy of 86.69%, with an alternative prototype suggested for as low as 86€. This, for a low cost, non directional count of vehicles, would be interesting to apply perhaps inside parking garages as an alternative to the usual parking barriers. Although, if needed, the VL53l1X could be used to have a directional counting. The second system consisted on counting the people going in and out of a room using a simple prototype of around 74€. The results of its performance will be revealed in a future chapter 8. In the next chapter, we will present our approach to count people in a conference room using a thermal camera.

6

Thermal Camera

Contents

After the initial data acquisition and usage of the thermal camera was done, the next logical step is to create an algorithm for its use in closed spaces. Two different approaches were considered: a static one to be used in conference rooms to count the current number of people sitting at a table; the second one is a people flow counter that is placed at the entrance of a room to count how many enter and leave the room. This allows to have a number of people to compare with air quality monitoring, in order to determine the occupancy of a room.

6.1 Static Setup

The first setup iteration records 1 second at 8fps, which leaves us with 8 frames. The algorithm is applied to each of those frames and then the mode is used to determine the actual number of people. The evaluation happens every minute.

The first setup was done with data processing via Matlab, with a simple people counter in frame. It was possible to reproduce this behaviour in an all in one system that runs in a Raspberry Pi. Adaptive thresholding according to temperature was implemented.

On the first iteration of the Raspberry Pi system we retrieved an image sequence from the Melexis MLX90640, processed each frame with some kind of thresholding (sufficient to only leave the hotspots of the heads) and use the nearest neighbour algorithm to find connected blob components. If the threshold was applied successfully, the number of people in the frame is given by the number of connected components in frame.

6.2 Setup Architecture

6.2.1 Static People Counter

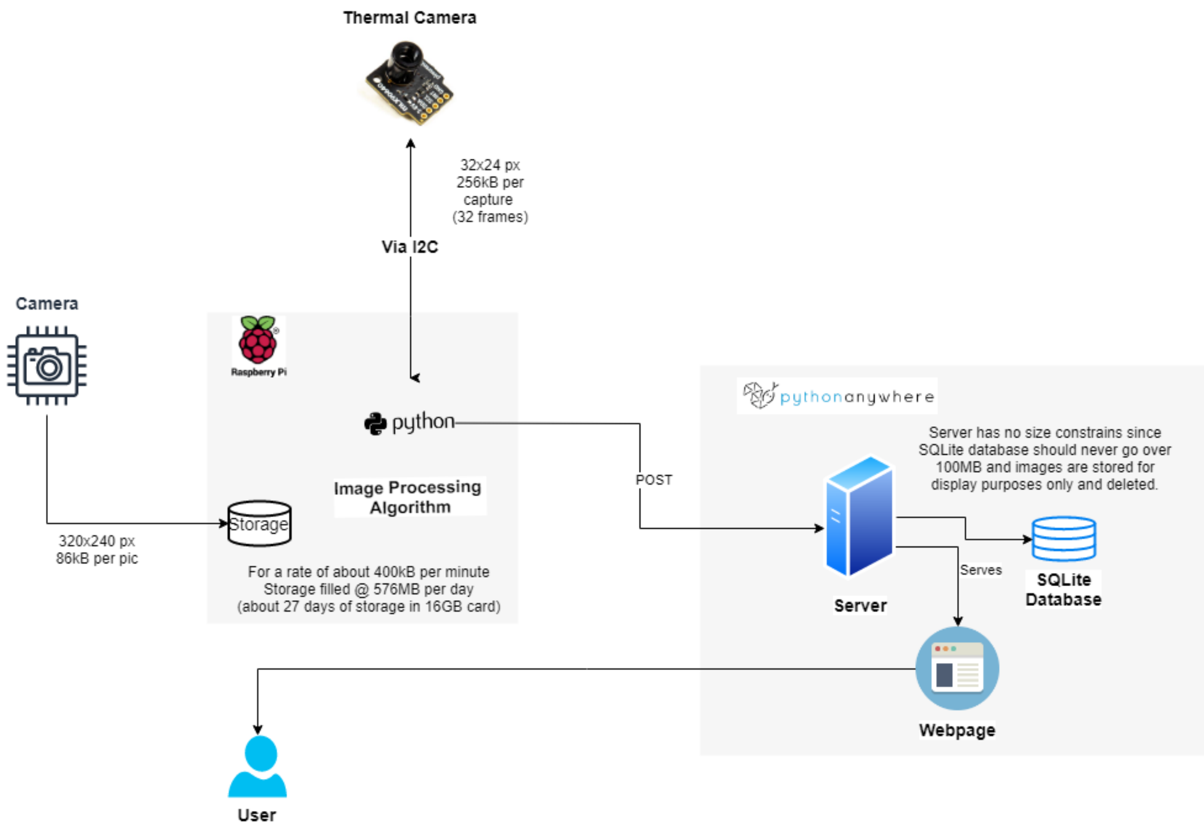
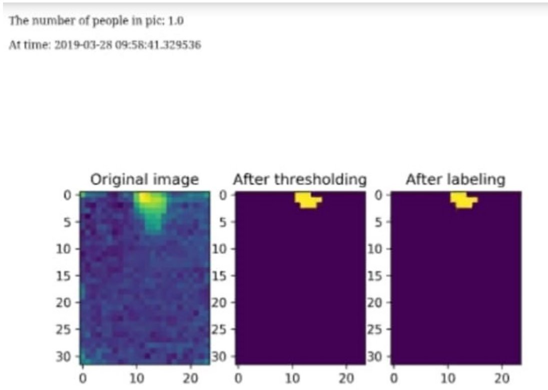


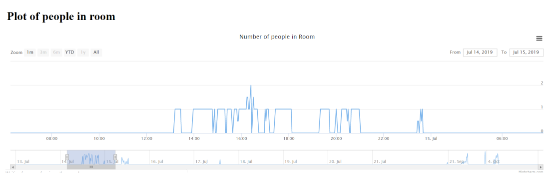
Figure 6.1: Static System Architecture

As seen in Figure 6.1, the thermal camera is connected to the Raspberry Pi via I2c, with a C++ library with some modifications made in order to save the values of temperature to files, enabling the Python script to be able to process them. The output of the library by itself is only an APNG file, which is basically an image sequence. Then the Python script processes the temperature values as seen in Algorithm 3 and posts the count result to a cloud server that updates the webpage for the user to see. This webpage, as seen in Figure 6.2 has a picture of the current thermal image, alongside a graph of the last days of people counted.

6.2.2 People Flow Setup

Matlab setup tracks centroids of people moving through and tries to assign them via Kalman filter, to know if the trajectory passes through the enter or leave room line. The base thought in this people flow setup is pretty simple, as we use the people counting per frame analysis as done in the Static Setup,



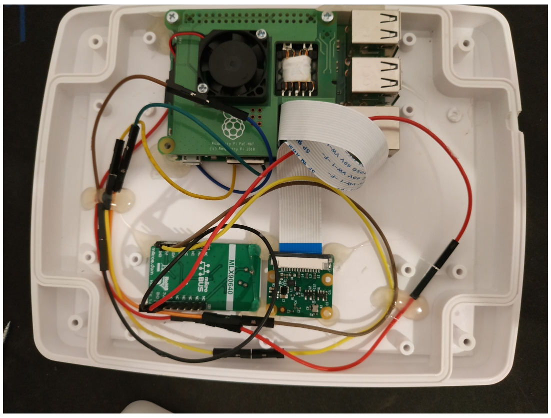


(a) Timestamp Portion of Website

(b) Graph Portion of Website

Figure 6.2: Website capabilities

but now we compute the trajectory of centroids between frames, using a Kalman filter to assign certain trajectories to its corresponding track. Unfortunately for a final setup, this architecture needs an extra computer compared with 6.1, therefore we did not have more than prototype results.



(a) Interior of the Thermal Setup

(b) External Look

Figure 6.3: Thermal Sensor Hardware Overview

In Figure 6.3(a), it is possible to see the interior of the thermal setup. On top we have the Raspberry Pi, with the two cameras on the bottom: the thermal on the left and the visible on the right. All this makes for a really clean setup, as it is possible to notice on Figure 6.3(b).

6.3 Algorithm

Algorithm 3 Thermal Camera Algorithm

1: **procedure** PROCESS IMAGE
2:     *Size* $\leftarrow$ 3
3:     *numpic* $\leftarrow$ 8
4:     *temperatures* $\leftarrow$ read(file)
5:     **for** $i = 0; i <$ *numpic* **do**
6:         *timage[i]* $\leftarrow$ temperatures[i]
7:     $threshold \leftarrow$ median(*timage*)
8:     **for** $i = 0; i <$ *numpic* **do**
9:         **for** $x = 0; i <$ *columns* **do**
10:             **for** $y = 0; i <$ *lines* **do**
11:                 **if** timage[i][x][y] $<$ *threshold* **then**
12:                     $timage[i][x][y] \leftarrow 0$
13:         result[i] $\leftarrow$ connectedcomponent(timage[i])
14:         **if** result[i].size $<$ *Size* **then**
15:             $(remove(result))$
16:         numppl[i] $\leftarrow$ size(results[i])
17:     pplcount $\leftarrow$ mode(numppl)

The algorithm for processing the thermal image is as follows: from a 32 frames per second video, we pick out the first 8 (therefore the numpic variable in the algorithm), in order to reduce the timeframe of analysis (and to reduce change in the pictures being analyzed). Then, we need to find what is considered background, in order to apply a threshold to the picture. This is made by median analysis of all pictures together (line 7 of the algorithm). Then, we apply the obtained threshold to each individual frame in order to have a preliminary analysis. This should leave us with only the objects of interest, which are the ones with the highest temperature. But sometimes there are some sensor faults, therefore there are small spots with high values that have high temperature. So after this it is run a connected components analysis in order to find objects in the picture. But not all objects in the picture are people, since there are some objects in conference rooms, for example, that also emit heat, such as laptops or projectors. Fortunately, these objects have a smaller footprint than people, so for the connected components we remove those that have a pixel size below a threshold (for now 3 pixels is the threshold). The number of connected components left give us the number of people in the room. After this, the mode of values of people in the 8 frames is classified as the number of people inside that room in that timeframe. This number is sent via POST to a server in order to display the information on a website, which displays the current image of the room, plus a graph of the count of people (last 3000 points, due to limitation of the highcharts library used). The server also updates the sqlite database used for storage with the new value.

Algorithm 4 Thermal Video Kalman Algorithm

```
procedure PROCESS VIDEO
    timage = readingsequence()
    threshold ← median(thermalpictures)
    track ← createtrack()
    for i = 0; i < numframes do
        for x = 0; i < columns do
            for y = 0; i < lines do
                if timage[i][x][y] < threshold+5 then
                    timage[i][x][y] ← 0
                if timage[i][x][y] > (threshold + 5) then
                    logic[i][x][y] ← 1
                else logic[i][x][y] ← 0
        result[i] ← connectedcomponent(timage[i],3)
        track_assigned_centroids = kalmanfilter(result[i].centroids)
    if size(track_assigned_centroids)¡ 8 then
        Delete_track()
    for i = 0; i < size(track_assigned_centroids) do
        direction = check_trajectory(track_assigned_centroids[i])
        if direction == 0 then
            'Did not cross line'
        else if direction == 1 then
            'Entered room'
            count = count + 1
        else if direction == -1 then
            'Left room'
            count = count - 1
```

This second option has the problem of not being able to work in real time, since it relies on having a 30 second data gathering, which gives us 480 frames to base the work on. Also, it needs a dedicated computer since it runs on Matlab. The main steps of the algorithm have the same thresholding technique used on the smaller image sequence of Algorithm 3 , where the median for the images is obtained and serves as threshold to zero the non important parts (of lower temperature) of all the frames. After this, it is run a connected component analysis on each frame(which is just to run a connected component analysis in a binary image with a neighbourhood of 8). This gives also a labeled binary image that we can use with the regionprops function of Matlab, that allows to get the centroids of each labeled group. In order to match each centroid from image to image, a Kalman Filter is used to produce a projected trajectory for each of the centroids in the frame. Then, using a cost criterion, in the next frame, a centroid is assigned to the track we had in the previous frame with the initial centroids. So the idea is to predict new location of tracks, then, for the detections in the frame, assign them a track, which is given by calculating the cost of the detection with every track available. The track with the lowest cost, will be the correct one for the detection. If any detection is non assigned, we update the assigned tracks and update those that are unassigned by deleting lost tracks and creating new ones if we believe these new detections don't belong to any old track.

After doing this procedure, with the locations of the centroids in the several different frames, we expect to have the trajectory of different people that crossed our thermal camera in the video. So the last thing to do is to check if the trajectories crossed the line that we consider the entrance and when they did, if they crossed it going outside or inside the room. The sum of all the values of this tracking variable gives us the number of people inside the room for the specified 30 second timeframe.

6.4 Results

Since the Kalman people tracking algorithm using Matlab wasn't feasible in real time, due to the post processing needed, we only have the prototype data counting which can be seen below.

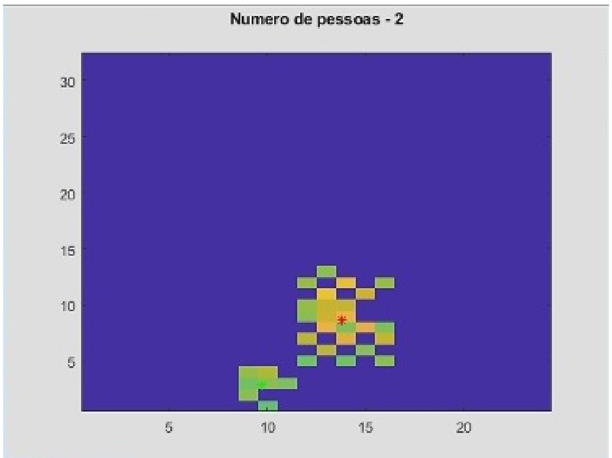


Figure 6.4: Plotting of video people tracking with Kalman centroids

This is a frame of the Video Kalman People Counting system. Notice the centroids and the people counting on top of the frame. This post video acquisitions analysis was processed in Matlab with a Kalman filtering forming the trajectories of the blobs.

In Figure 6.5 it is possible to notice the different points that the people centroid crossed and the point where we decide if that person is leaving or entering the room. Despite the good results from the dataset (with two people leaving an entering, we managed to count it correctly), the need of a secondary program and computer with Matlab showed that this was not the way to pursue. So it was time to test and improve the static people setup, that could be used in conference rooms, for example.

The static people setup showed more promise in preliminary testing, even more so with the fact that it can be done all in a single Raspberry Pi, making this a very interesting proposal for an all in one system. Unfortunately, as we will see in the results, densely populated rooms can prove to be a challenge for a thresholding method as simple as the median one we used.

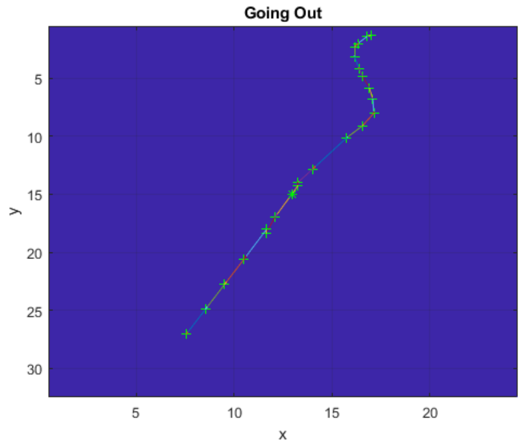


Figure 6.5: Plotting of video people tracking with Kalman centroids

The first result presented here is the dataset of a simple room in Técnico, which is most of the times sparsely populated, with an usual maximum of three people inside it at the same time. The actual values of people inside the room were obtained with a still frame of a camera connected to the raspberry pi and manually inserted in a data structure via observation of those pictures.

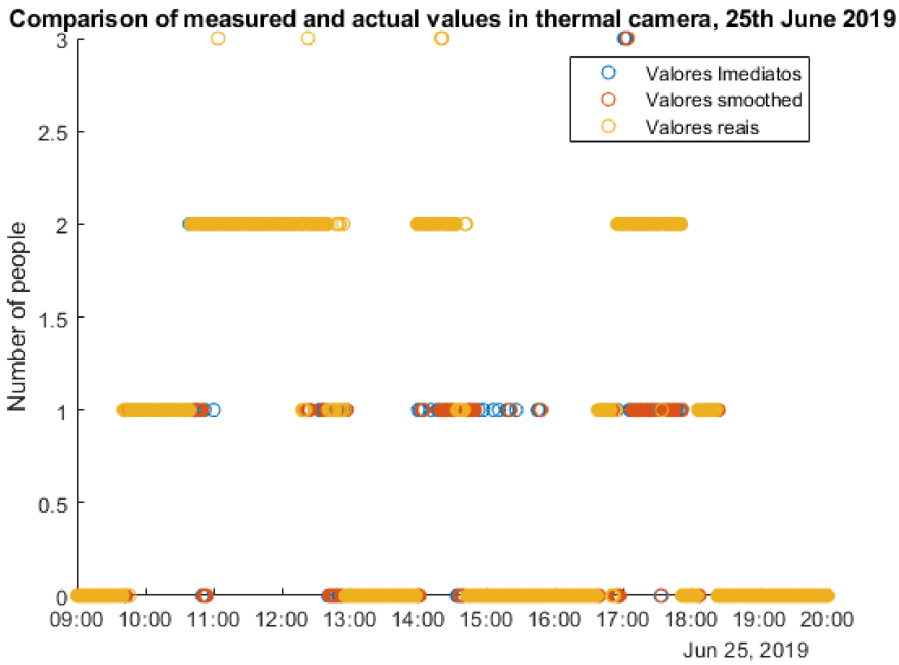
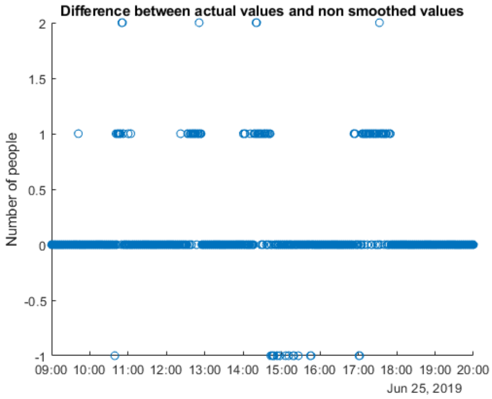
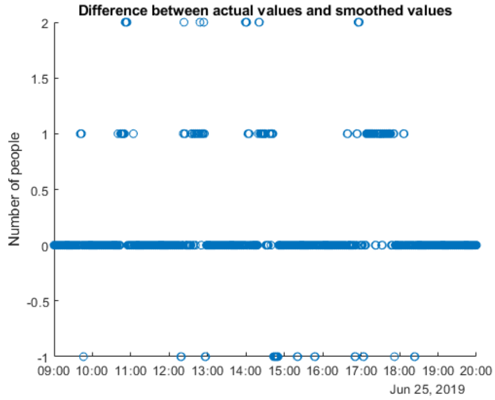


Figure 6.6: Comparison of people counting methods with actual values

The methods being compared are the instant people count (which is the direct output of the algorithm) and a smoothed version which only changes the people counted after that value was seen a certain amount of times (in order to reduce false positives, but this also filters out rapid changes in people inside the room even if they are correct).



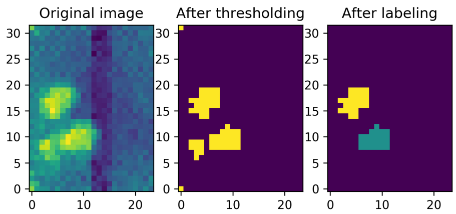
(a) Delta between real values and non smoothed values)



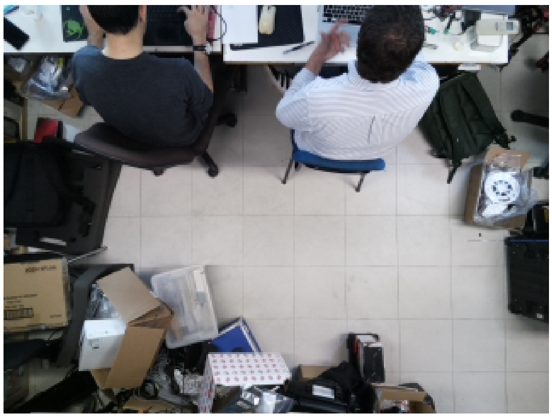
(b) Delta between real values and smoothed values

Figure 6.7: Delta between real values and output

In Figure 6.8, it can be observed that the smoothed values show a clear delay in following the actual people in the room, as it is expected. Therefore it is a developer choice if he prefers a more stable and less reactive system, to smooth out the people count, or if he prefers a more reactive system. The choice will depend on the desired implementation.



(a) Thresholded thermal image



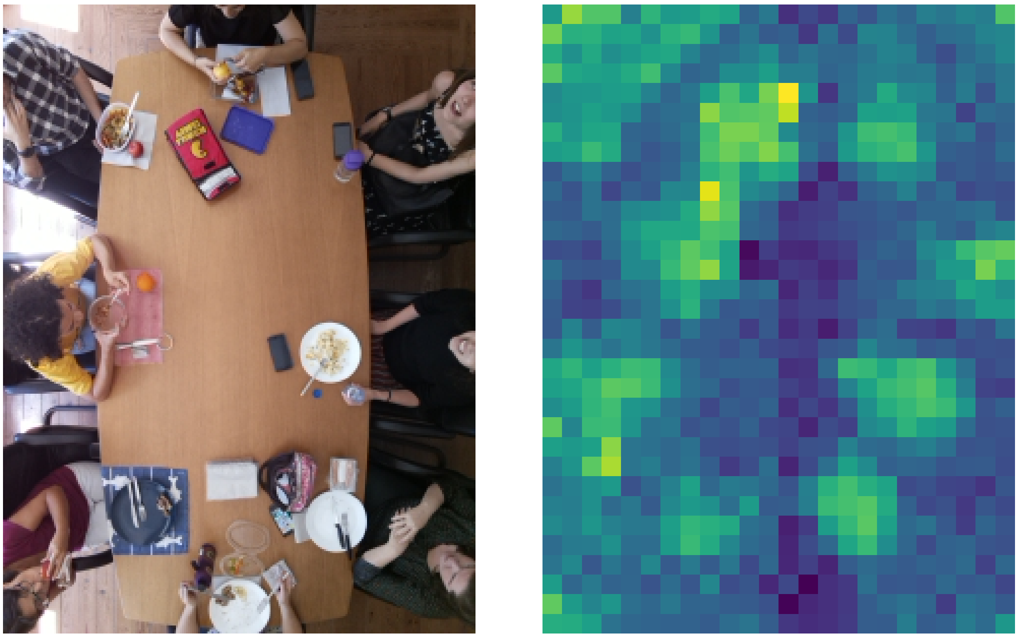
(b) Real Picture

Figure 6.8: Comparison between real and threshold

The median thresholding, as seen in picture above, works really well when the picture isn't fully populated. The algorithm worked mainly as it was expected, having only some mistakes when a hot laptop was mistaken as a person, due to it's enormous thermal footprint. Even then the non smoothed system managed 83.1% accuracy and the smoothed system 80.18% accuracy.

But recalculating the threshold based on median every frame has some issues, that can be seen in the next dataset that was placed inside a house that got room temperatures up to 30 degrees Celsius, plus a maximum occupation of up to 8 people.

When all the room has an high temperature, the temperature resolution of the thermal camera we are using makes for a really difficult contrast between background and people. The median in a setting like this is also really high, which is not ideal for the thresholding. Therefore we needed an alternative for the thresholding, and as such the second method we tried for the background was to threshold based on a temperature sensor placed on the room. But first, a comparison between a fully populated thermal room where one of the images was thresholded using just the median value and the other one has a better thresholding using a median summed with a constant (to increase differentiation between people).



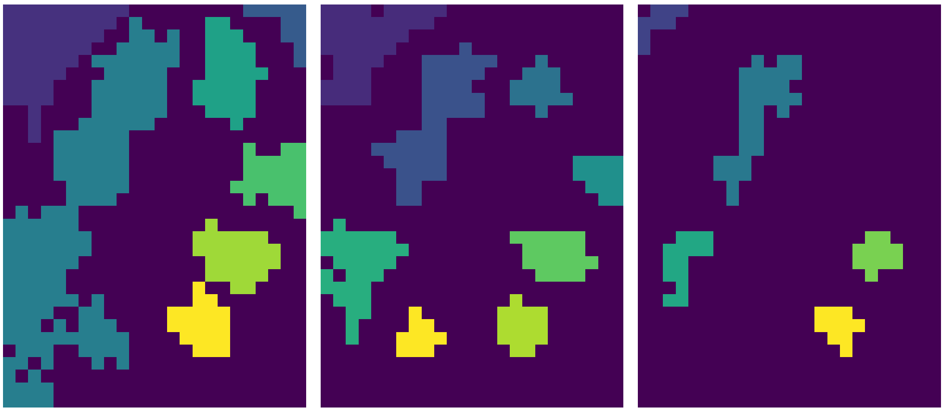
(a) Real Picture

(b) Thermal Image

Figure 6.9: RGB and Thermal Images of the room

In Figure 6.9, although the thermal image is slightly twisted and on the top left of the thermal image there is a body of heat (which is due to the lamp where the setup was installed), we can clearly see that it is possible to tell apart each one of the persons sitting at the table, with a proper threshold setup.

In Figure 6.10, even with a near perfect thresholding as seen on the left, the different temperatures in each of the places makes it that you cannot differentiate the two people in the top left (that are considered just one blob) without losing the person in the bottom (represented in yellow). The thresholding on the right is obtained by removing all the values that are below the median value of the 8 pictures captured + 1, while on the left you see the impact of using the threshold with just the median value. If you add



(a) Thresholded Image using just median value

(b) Thresholded Image using median value plus constant(+1)

(c) Thresholded Image using median value plus constant(+2)

Figure 6.10: Differences in thresholding

2 to the median value, a lot of the detail of the image is lost, such as three of the people in the frame, while not solving the problem of two people being considered one. This shows the importance of a good threshold value. Even then, with the dataset we obtained in this new location, we could not go over 77% accuracy (or 44.81% in the dataset part with pictures). A new method with an adaptive thresholding with room temperature was necessary (alongside the median) for better performance.

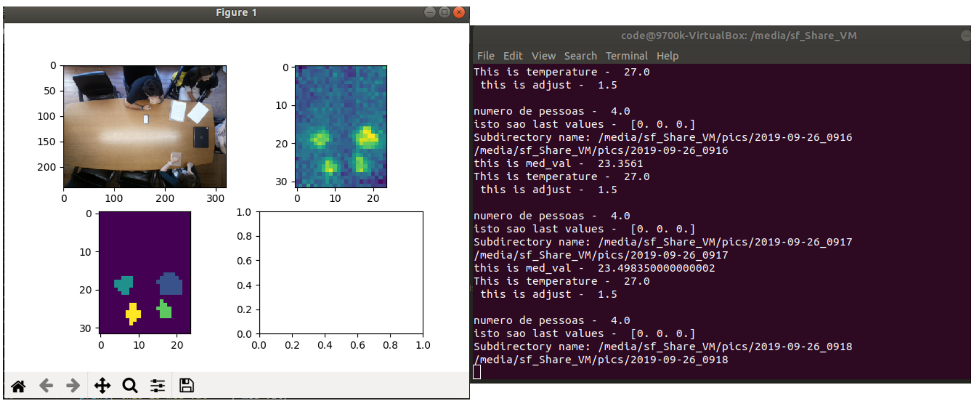


Figure 6.11: Temperature based thresholding

Using a temperature based thresholding, like on Figure 6.11 has improved the results. It is important to be able to dynamically adjust the threshold because the temperature present in the room influences how big is the delta between ambient and person. Using this new technique, results of around 82% accuracy were obtained. In chapter 8, we will delve further in depth to the setup and performance of this variant.

6.5 Bill of Materials

Part Name	Quantity	Type	Price	Link
DFRobot Raspberry Pi 3 Model B+ Starter Kit	1	SBC Kit	90.68 €	https://tinyurl.com/y2yetdl5
Raspberry Pi Camera Board v2 - 8 Megapixels	1	Camera	35.05 €	https://tinyurl.com/yyvgapt5
MLX90640 Breakout Board	1	Camera	89.87 €	https://tinyurl.com/yxl5qw3u
Total			215.04 €	

Table 6.1: Bill of Materials for the Materials Used in testing

In this case, we are talking about the total cost of the thermal counting system. Initially we used a Power over Ethernet (PoE) approach to the system, reducing the cable clutter to a single cable. But the location where we installed the system did not provide Ethernet via the cables in the room, so we decided to simply use a regular microUSB power supply with WiFi for data connection instead. This cost can be reduced for prototypes if the bare sensor is used (around 40€) or a cheaper breakout board is used (around 60€). This more expensive breakout board was chosen since it was one of the first in the market, plus the 110 degrees of FoV helped to cover a larger portion of the room. The total cost being around 215€, makes this an interesting proposal due to the accuracy obtained.

6.6 Conclusion

In this chapter we proposed different approaches to counting people with a thermal camera. The first one was to use the video recording capabilities of the MLX90640 sensor and try to record trajectories of moving people and notice when they crossed the room. The fact that this needed an extra computer, for Kalman processing in Matlab, quickly dismissed this method for application. The next logical step was to position the camera in an overhead position taking a sequence of frames and using median thresholding to count the number of people present at the table at that given time. This method proved to be relatively accurate with an accuracy of 82% in this iteration, value that is going to be improved in the final chapter, when temperature given by an external sensor is used to improve the threshold. The following chapter will demonstrate the setup used both for car and people counting using the technology that should produce the best results: the TI mmWave.

7

TI mmWave

Contents

The TI mmWave was the technology that showed the most promise in our preliminary tests, confirming similar results presented in papers such as [24, 25] . So it was decided to test it for not only people counting in indoors environments but also for vehicle counting, in order to have a comparison system for our ToF to be evaluated against.

7.1 Setup Architecture

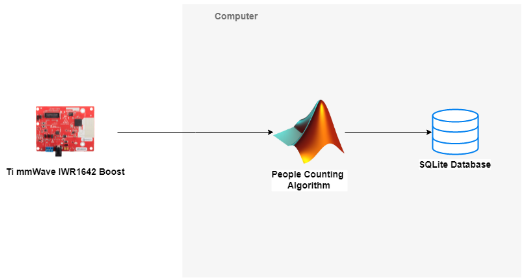


Figure 7.1: Basic setup architecture

This is the basic Setup Architecture for using a TI mmWave Radar, more specifically for People Counting purposes, but it can be expanded for car counting, if the mmWave is flashed with the correct binary. This is because depending on the binary that the mmWave is flashed with, there are multiple demos that can work in Matlab. But for the people counting case, we have a TI mmWave IWR1642 interfacing with a Windows computer via Matlab with serial interface (both input and output; input to receive some serial commands and output to provide the data to Matlab). The people tracker in a room was already made, but it was modified to be GUI-less for better performance. Choice of a proper configuration file is imperative for good performance, since we have to pick the correct parameters in order to avoid ghost reflections and other type of gating problems. Then, the counter is being logged into a SQL database for further analysis. There was a try of making it display in a webpage, but Matlab's POST is slow for this kind of device that has a high data refresh rate.

Choosing the right parameters for the radar setup is really important since we want to avoid multiple problems that can arise with an improper setup. The parameters that are available for tweaking are the chirp configuration (which needs to be setup for the range that is used) and the tracker parameter configuration, which is the configuration of the people counter demo being used as a basis. Some of the most common problems encountered are the fact that in a closed environment, there are a lot of reflections, which can be seen as ghosts, which are fake points that only clutter the area of interest. These happen mostly when people move near walls.

Another problem that can be faced is the assignment of multiple people to the same track, which means that the centroid being tracked is actually multiple people being considered as one. Since the implementation is going to be made in closed environments, people tend to follow each other closely. Therefore, in order to have the best accuracy, assuring that people moving close are correctly tracked is a must.

7.2 Car counting

Using the TI mmWave as a counter for passing cars in both lanes is possible by using the prepared sample code by Texas Instruments, which will let us compare the performance of this sensor with the Time of Flight one.

7.2.1 Physical Setup

Figure 7.2: Setup for counting both lanes

The TI mmWave was mounted outside (as seen in Figure 7.2 in such a position that allowed the radar to record both lanes of traffic that occur in the entrance of Instituto Superior Técnico - Both entering and exiting the university. And while it was recorded in the data the number of cars entering, only the exiting number of cars will be used for comparison purposes with the ToF Sensor.

Proper setup of the data recording demo is essential to assure that the cars are in their desired lanes, as well for the playback software to play the data back correctly. Incorrect setup (or in cases where the centroid doesn't have the needed persistence) can make the software miss the passage of a car.

7.2.2 Result Analysis

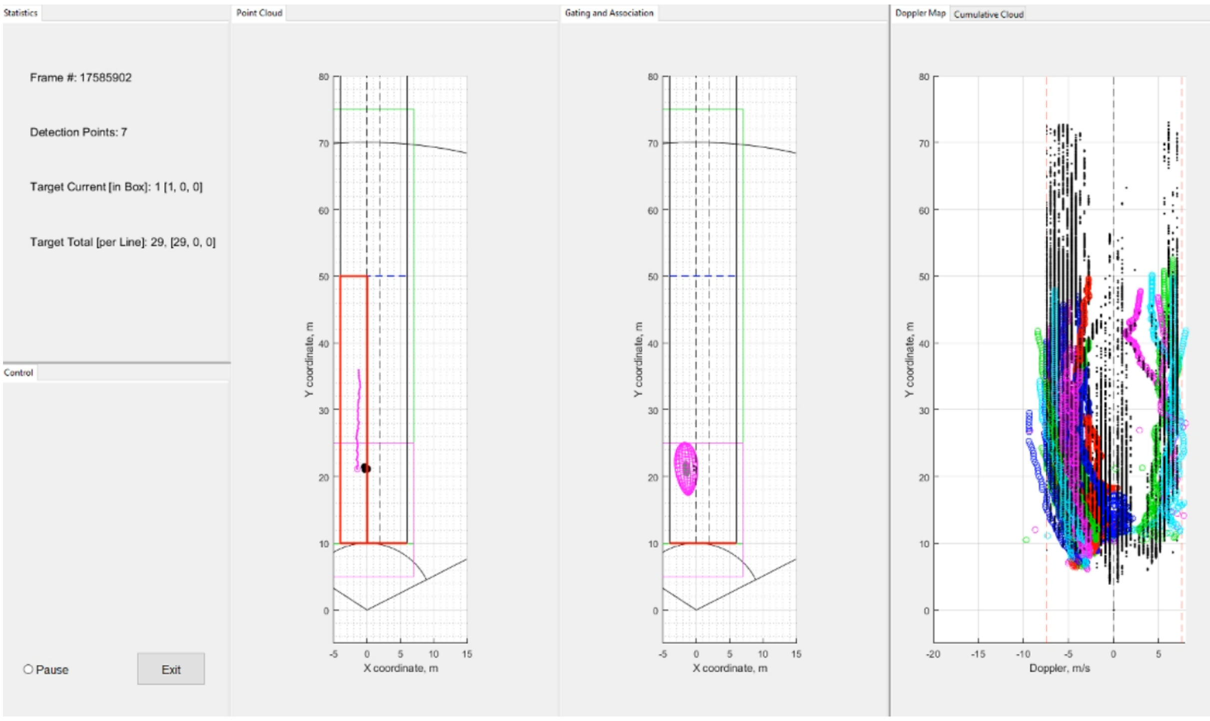


Figure 7.3: Playback Software for counting

In order for data to be comparable to the work developed in chapter 5, the data chosen for analysis was for the 11th of July. Unfortunately, due to TI's design choice, the timestamp present in the data for the software seen in Figure 7.3 is only a synchronization timestamp and not the actual data of the system, so the exact time when the car has exited is not possible to get. It is however possible to gather the data from a certain timeframe and analyse it in terms of total cars. When replaying back data, the software counted 487 cars in this timeframe and with hand-made annotations (noticing when the centroids would form, but would not count as crossing the exit), the count is 648.

The ToF sensor in the same day (11th of July) counted 746 vehicles, which corresponds to an accuracy of 87.35%. This is a better result than the one obtained by the mmWave prototype. The results aren't directly comparable, because while the ToF sensor is doing the detection when the vehicle is stopped, and with a low sample rate (the vehicle is only stopped a few seconds while the driver passes his card), the mmWave prototype has a job that is several levels of complexity above the task being done by the ToF. In the detection scene of the mmWave prototype, both the sensor and the DSP acquire the targets while moving, and track them continuously in a vision field that has several dozens of square meters.

The results of the mmWave in a first analysis are well below the expectations with an error of 206 counts(648 detections in 854 vehicles that left the zone), although having in consideration that we have a portion of the park (112 parking spots) that we are not counting, since the vehicles leave this parking zone to the exit, not crossing the main road being monitored. This park also has typically a rotation rate that is superior to 1. In an hypothetical scenery that the park was only occupied and unoccupied the 112 parking spots, the mmWave detection rate would be of 87.3%, which is a similar value to the ToF sensor that does not suffer from this uncertainty.

The prototype that we built allowed us to obtain in a reliable and regular way data throughout a long period of time, which was only limited because the installation caused some discomfort to the security guards, due to the computer running the system being a potential thing for people to steal. Also the necessity of using a powerful computer for the processing, constant electrical power and the system complexity (which took longer than initially foreseen to be stable), have made the use of the mmWave for car counting in other scenarios not worth it.

Having this thoughts in mind, the results are probably due to setup characteristics and chirp selection, since the TI mmWave (and FMCW radars in general) are very dependant of the environment where you setup them, besides having a correct chirp configuration for the current use. The main problem we could find was that sometimes cars could not be correctly identifying if they were too close, or sometimes there was a large pointcloud that was not classified as a car. Real world performance is always different to test setups, and probably the position of the radar, where it was lower than the cars it was detecting, did not contribute for the cases where two cars were closely following each other (the hypothesis is that most cars that were not counted happened during the most dense hours, in terms of car crossings). The main advantage with this system is that the radar can detect both incoming and outcoming lanes.

7.3 People Counting

For counting people using the TI mmWave there were a lot of promising ideas that could be used. Firstly, we thought of using the PeopleCountingDemo just as it comes out of the box, but we noticed this would not be enough, even when using the people in box counting option. This is because, in the software, people could stop being detected, if they did not move enough, since the radar only detects moving objects. This would make for an incorrect count. So the option chosen for tracking people in and out of the room was to check when the centroids being detected crossed a certain crossing zone, and the direction of crossing determined if they were entering or exiting the room.

The results will be shown in section 8, where we will fuse all sensors in a single environment, but with some preliminary analysis, the people counting algorithm fares pretty well to count the people that exit the room, but since the setup in the room makes that the centroid is only tracked very near the moment the person enters the room, sometimes the centroid is only generated after the person has already crossed the line that counts the crossings.

7.4 Bill of Materials

Part Name	Quantity	Type	Price	Link
PSC15R-050	1	AC Adapter	11.56 €	https://tinyurl.com/yyuvduzq
MEDION Akoya i7 9700, 8GB Ram	1	Computer	579.99 €	https://tinyurl.com/y24mblcd
IWR1642BOOST	1	FMCW Radar	307.85 €	https://tinyurl.com/y6rpxu3q
Total			899.40 €	

Table 7.1: Bill of Materials for the Materials Used in testing

The Bill of Materials is going to be similar for both setups using the mmWave, since the only thing needed is the computer to interface with the Radar and the Radar itself (plus a suitable power supply). Although the prebuilt computer indicated is not the same we used (parts were bought separately), the hardware components are similar and equivalent.

7.5 Conclusions

In this chapter we wanted to see if the TI mmWave could live up to its expected performance. Results in car counting have shown that it could register a performance of around 75.8% raw accuracy and when accounting for the uncertainty of the parking spots next to the exit (that would make the centroids of the cars appear too late), we would get 87.3% of accuracy, a value similar to the one obtained in chapter 5. Even then, this is not a good value if we compare the systems strictly in cost/performance ratios. And while it is true that the TI mmWave had a task that is several degrees of complexity harder than the TOF system (it had to track the cars), we expected it to at least have better results than the TOF system. For people counting we just established the general parameters of the final setup that we are going to evaluate in the following chapter. We are setting up for the last chapter, where air quality measurements and the people counting systems we have created will be compared to the reference values of people inside the testing environment, making us understand which one has the best performance according to our metrics.

8

People Counting and Air Quality

Contents

After all the testing done in a separate manner, it is time to see how all these solutions compare when set up for use in the same room. So once an appropriate location was chosen, we compared the ToF, thermal camera and TI mmWave performance. All these devices will be used as previously mentioned.

8.1 Physical Setup

The physical location of the setup is very important, mostly due to the nature of every sensor.

Figure 8.1: Environment where the setups will be installed

Starting with the ToF based solution, it needs to be placed in such a way to people have to pass in the zone created by its laser. Proper sensor position is important for good results. The door, as seen in Figure 8.2(a), ideally would be narrower to assure that people cross the ToF sensor.

This was the best location in the room to place the sensor, since it is close enough to the door that results should be good.

Now for the thermal sensor, the main issue that occurred in the first data acquisition, in chapter 6, was the presence of an overly big hot spot caused by the lamp where we installed the sensor. This lead to trying to install the camera directly underneath the lamp, but unfortunately this placement, even with the wide FoV of the thermal camera (110$\deg), couldnotcapturetheentiresittingtable.$

As seen in figure 8.2(b), we had to go back to the original solution, but this time the camera was installed further away from the lamp, reducing the hotspot to a minimum. The hotspot still exists when the sun hits a certain part of the hardwood floor as seen on figure 8.9

At last, there is the TI mmWave, where the positioning is also important. Ideally, the sensor would be positioned in an head on position with the door, to be able to create the centroids while the person is still outside the room, making for a better tracking. This is impossible in this case, since the objective of

(a) Location of ToF Sensor

(b) Location of Thermal Camera

Figure 8.2: Location of install of two sensors

(a) Location of TI mmWave

(b) Location of Air Quality Sensor

Figure 8.3: Location of install of other two sensors

these sensors is to prove they can be placed in location that do not disrupt the ambient, and a radar on a stand directly in front of the door (making sure that the person could see it), is certainly not the most aesthetic option. So it was placed in one of the shelves to the side, at an height good to detect people while standing, but not so much while sitting (which is not the desired goal anyway).

All the sensor data will be crossed with an air quality sensor, shown in figure 8.3(b) , to determine if there is actually a correlation that can be made from the air quality inside the room and how many people there are actually inside. This air quality sensor was obtained off the shelf, so everything works properly.

8.2 Results

The main importance of these results is to see if we can actually see a correlation between air quality monitoring (most namely CO_2 and VOCS) and the occupation of the room, calculated via our devices.

8.2.1 Air Quality

The best way to start analysing results is to determine what was the air quality like inside the room, and then try to cross it with the Ground Truth of people using the room. Then, if it is possible to validate that the air quality can be positively correlated to the number of people inside the room, use the air quality graph to compare how well our devices performed.

The first analysis is the plot of both VOC (Volatile Organic Compounds) and CO_2 (Carbon Dioxide) in ppm (parts per million) during a 24 hour period.

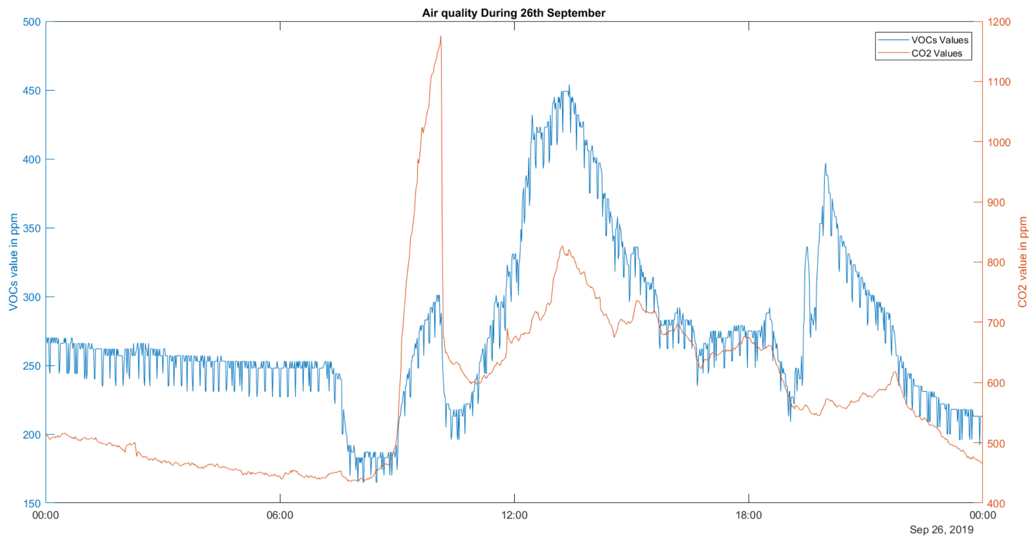


Figure 8.4: Air Quality during 26th September

In figure 8.4 we can notice that both CO_2 and VOC stay relatively constant until around 07:00 but while the VOCs drop, the CO_2 stay constant. This can be explained by opening the doors of the room but not going inside, since VOC can be dust being released after opening of the doors. But when they both start to rise is when the room starts being occupied (proved by the people occupation graph in figure 8.5). It is also possible to see that while the first meeting has a much higher CO_2 concentration, but the VOC did not increase much, at lunch hours, when people eat and release particles of food to the atmosphere, the VOC increase much more than during the first meeting. But people were using the room, as seen once again in figure 8.5. It could also be that during lunch the door was left open so the CO_2 concentration was not as high, or perhaps the smaller duration of this event did not allow the CO_2

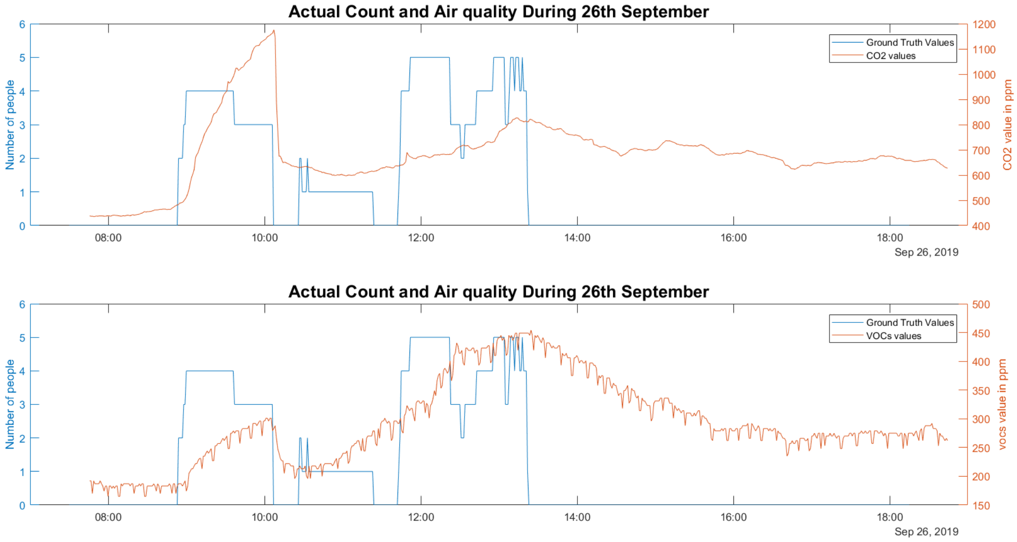


Figure 8.5: Ground Truth Comparison with Air Measurements

to accumulate as much. At last, notice the decrease in CO2 in an approximately constant manner during the rest of the day, but a peak in the VOCs could be explained by someone releasing dust particles in the room (around 20:00). The CO2 starts decreasing in a bigger peak around 22:00, which makes sense since the room being monitored is at ground level. Since CO2 is an heavy gas, it accumulates at the bottom, so while the room might not be occupied, if there are people inside the building, the CO2 level only starts rapidly decreasing after they all left.

This small comparison between air quality monitoring and Ground Truth of the room occupancy shows that it is possible to determine if a room is being occupied with some uncertainty added to the mix. For example, just looking at the CO2 graphs, one would probably conclude that the first meeting had higher occupancy than the one that occurred during lunch. But using out people count solutions, one could see that is not the case, which is also confirmed by the ground truth. So ideally, the air quality measurements support another counting system.

8.2.2 ToF

Using the ToF solution proposed in chapter 5 revealed to be an adequate solution. Some issues were expected, since this is a single ToF sensor measuring two zones, so if two people cross the zones with a small space between them, they can be incorrectly identified. A faster processor than the one on the Arduino Uno, could have also helped.

The results shown in Figure 8.6 show that there were more entrances were counted than exits, which is not correct (that would mean that in the end of the day, there would be people inside the room). This

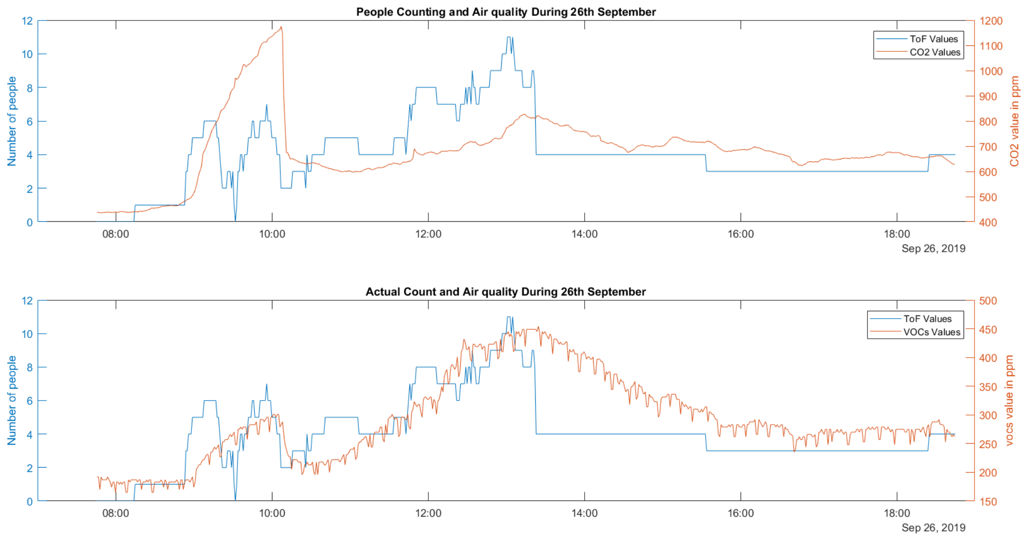


Figure 8.6: Air Quality results compared to ToF People Count

denotes that either the sensor has a better performance in one of the directions, or that people have different behaviours when leaving and entering the room. This last hypothesis seems a possibility, since that in most conference rooms, people tend to enter the room slowly and with space in between them, while they tend to leave in a more rushed manner, with less gaps in between persons.

The results are pretty satisfactory since it is possible to notice a relationship with the increasing values of people count (when it starts increasing around 9:00, both CO2 and VOCs values start increasing), having a drop off a little after 10:00, when the counting system determined that people left the room. A little after this, we started seeing another increase in VOC and CO2 around 12:00. The clearly noticeable part of people leaving the room, can be seen around 13:10, making the CO2 and VOCs value peak and slowly get lower over the next few hours. The person being recorded leaving at around 16:00, probably entered and left the room very fast, making this +1/-1 count happen, which is one of the problems of the ToF sensor. In a macro point of view, it is possible to see that the occupation is clearly shown by the air quality and that the clear increases in ToF both in entering and leaving the room predate changes in the air quality.

8.2.3 Thermal Sensor

Once again, this location was chosen to test the performance of the thermal camera in the same way as before. The new and improved version of the algorithm with adaptive thresholding based on temperature was used in order to increase performance, but while reviewing the images there was something odd we noticed. In Figure 8.7 we can see the various elements of the Thermal Sensor. On the left, we have

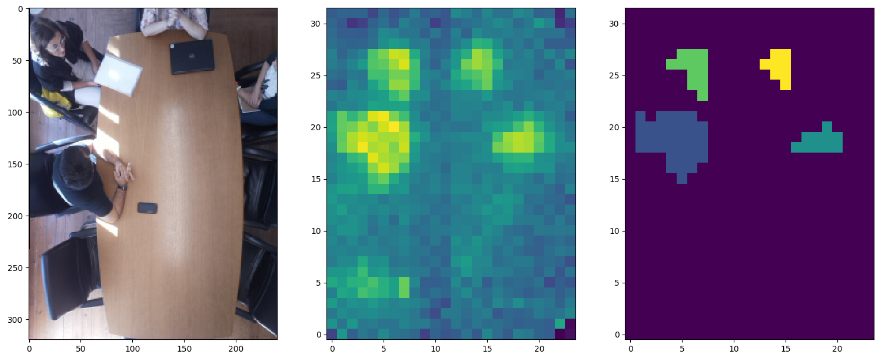


Figure 8.7: Visible Image and corresponding Thermal Image

the visible picture that is the basis for performance analysis of the thermal sensor. The middle picture, is the original thermal picture, without any type of thresholding or adjustment. And on the right it is possible to see the outcome of the algorithm, with four identifiable blobs corresponding to each one of the persons in the picture. But are the visible pictures and thermal pictures completely overlapped in a 1 to 1 relationship? Not quite, as we will see in figure 8.8.

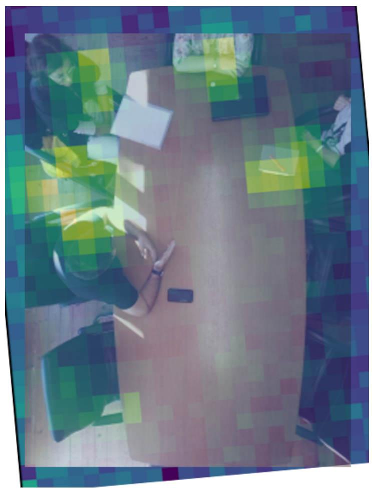


Figure 8.8: Overlap between visible images and thermal camera

In figure 8.8, we can see that due to the wider FoV of the thermal camera, it captures some zones that the visible camera for ground truth does not capture.

As seen in figure 8.9, the spot on the floor that receives the sun gets too hot, which leads to false detection on the algorithm. So we had to change the algorithm to remove counts in that general area.

As for this sensor, in figure 8.10(a) and figure 8.10(b) we have an accuracy of 86.07% which is pretty

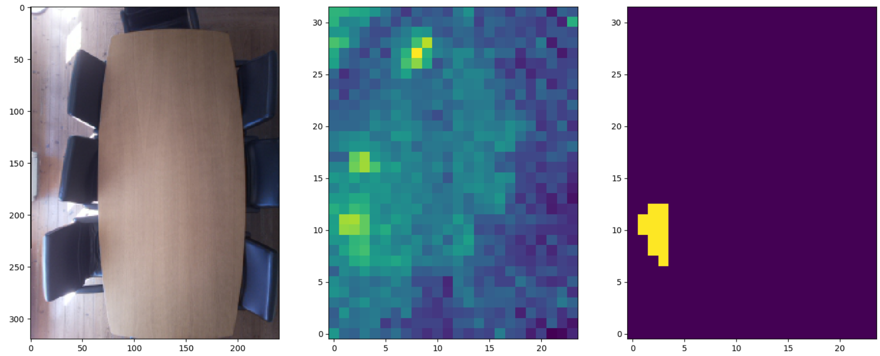
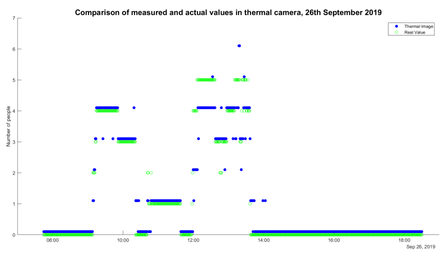
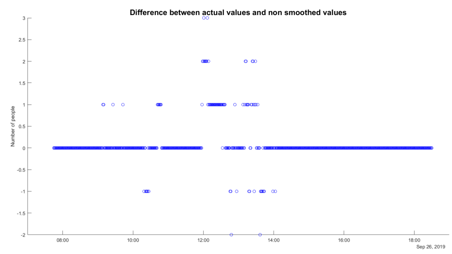


Figure 8.9: Sun hitting the floor



(a) Thermal Sensor Performance



(b) Thermal Sensor Delta with Real Count

Figure 8.10: Thermal Sensor Performance Overview

good since during the lunch hour the room is densely populated. Even with this level of accuracy, the thermal footprint of the image lets us know if the room is very populated or sparsely populated. After this we need to see if the results compare in a positive way with the air quality measurements. The data presented in figure 8.10 can be represented in a more friendly visual way, to represent the performance difference of the thermal sensor to the actual count, via the use of an heatmap.

If the heatmap represented in figure 8.11 does not present a clear view of the data, a more technical view of the data is presented in table 8.1 below.

Number of errors(+/-)	Percentage
0	86.07%
1	11.30%
2	2.32%
3	0.31%

Table 8.1: Performance of Thermal Camera

With the known thermal performance results, we can safely say that it can be used to asses the occupancy of a room. So it is clearly interesting to cross the value of the thermal results with the air

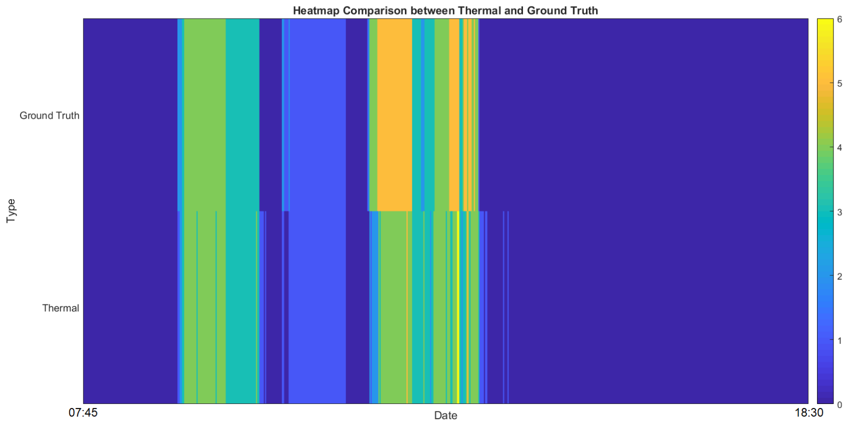


Figure 8.11: Heatmap Comparison

quality data, to see if there is some kind of correlation.

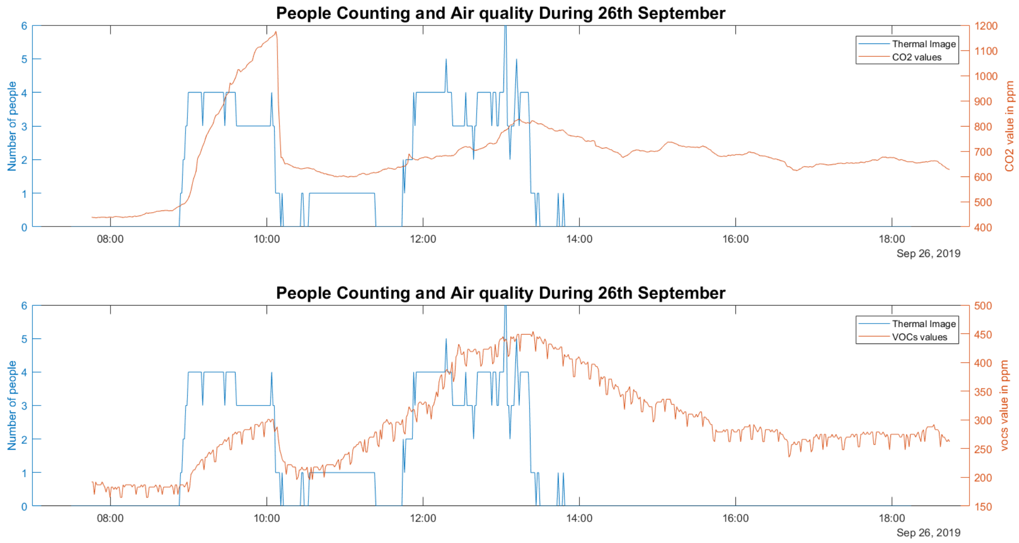


Figure 8.12: Thermal Sensor compared with air quality

Once again, we can notice a clear correlation between the air quality and the occupation obtained via thermal images. Both CO2 and VOC values have an increase when the room becomes occupied, and start decreasing after it becomes vacant. Once again this method proved to be a reliable indicator, matching the air quality.

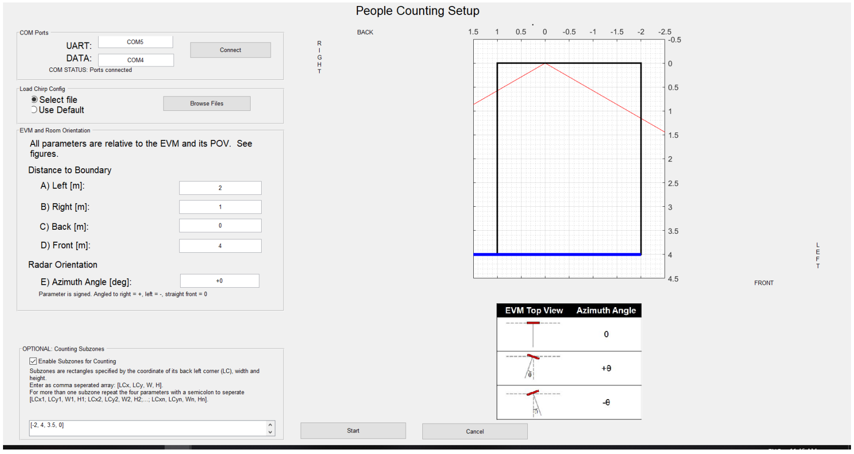


Figure 8.13: TI mmWave Software for People Counting

8.2.4 TI mmWave

In figure 8.13 we can see the configurable parameters of the software for people counting, most notably the location of the line to be crossed. This is one of the most important parameters for good performance, because it must be placed properly to detect people as they cross the door entrance.

Unfortunately, for the mmWave, the results weren't as good as we were expecting, but we clearly counted the exits from the room more successfully. This is due to the problem mentioned before, that the positioning of the mmWave sensor was not ideal.

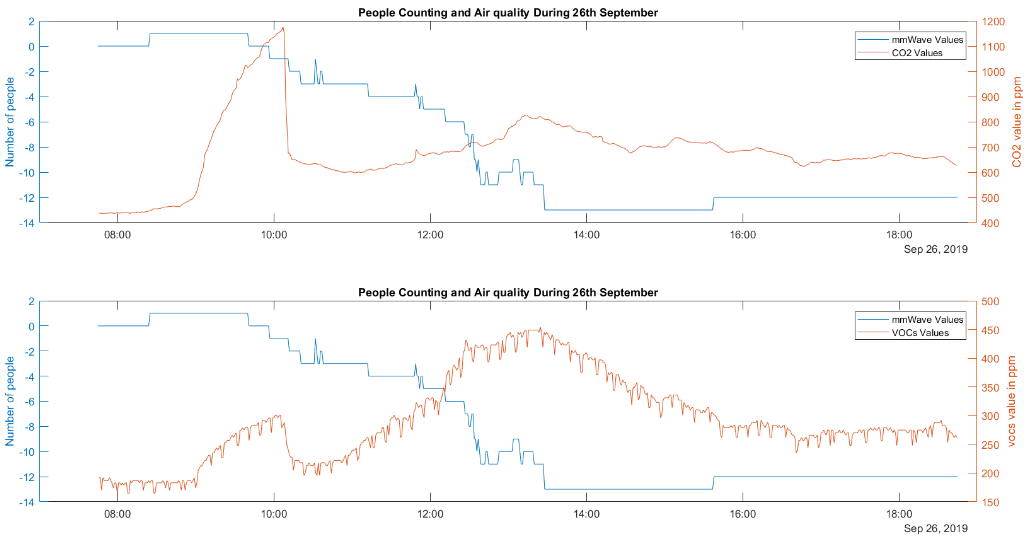


Figure 8.14: TI mmWave compared with air quality

In figure 8.14, we can clearly see that the people count keeps moving towards negative numbers, this is due to the fact that this sensor is detecting mostly the people that leave the room. It was expected not an ideal performance from the radar, since they work best in open air environments, mostly due to the amount of reflections that happen in closed environments. Even then it is possible to notice the decreases in CO2 and VOC with the decrease in the counts on the mmWave, noticeable around the 10:00 period and the 13:00 period.

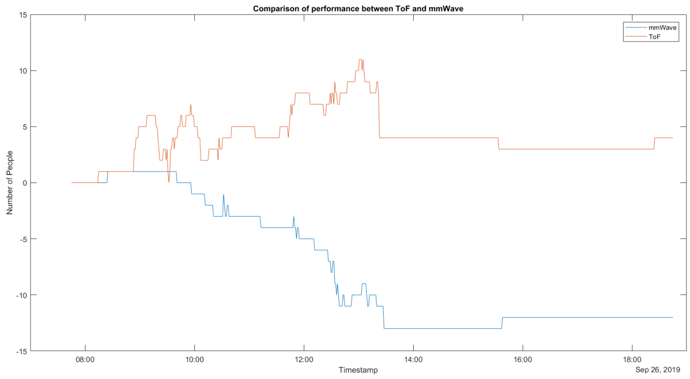


Figure 8.15: TI mmWave compared ToF Sensor

We figured that a comparison between counts in both ToF sensor and mmWave would be interesting since they both compared entrances and exists. We see that while people entering one at a time work okay, the ToF is an overall better choice with the capability of counting entrances in a more consistent manner. The leaving of the room can be seen in both cases, which means that the exits that were counted with the mmWave were also counted with the ToF.

8.3 Conclusion

In this chapter we overviewed the results of all the people counting solutions implemented by us, together with the air quality results. Noticeable good performers were most certainly the thermal camera, with an accuracy of 86.07% comparing to the people count and the TOF as a support technology for the air quality measurements. The TOF manages to correctly identify the bigger changes in people counts (as in when multiple people enter or leave the room) and in sparse cases (with just one people). The main issue happens when the error starts accumulating with several passages (three people enter but only two are counted). Even then, we can see that when the TOF detects an increase in people in the room, the air quality shows it as well. Another good thing was the air quality monitoring as an indicator for occupancy. In the first peak of room usage, since the room door was maintained closed, we could clearly see an increase in CO2 values, which means that we can use this system together with one of the other good performance sensors in order to correctly determine the occupation of multiple rooms. The results that did not satisfy us were the TI mmWave ones, because while exits, in a macro point of view, could line up with the TOF system, entrances were not correctly gauged, since our placement of the radar did not allow for centroids to form in due time (this would only be possible in a straight forward positioning of the radar). In the next chapter we will produce some closing remarks about the several experiences conducted and how can they be improved upon.

9

Conclusions and Future Work

Contents

It is now time to better analyse and conclude upon the results obtained in the previous chapter, alongside giving some thoughts and remarks on how to improve this work for future endeavours.

9.1 Conclusions

Having compared the results obtained with different technologies, seen in Chapter 8, it is time to update the comparison table 2.1 with the results that we obtained in our tests.

	ToF	Thermal	TI mmWave
Low-Cost	(3)5	(4)3	(1)1
Performance	(3)3	(4)4	(5)2
Range	(4)4	(2)2	(5)5
Low Complexity	(4)4	(3)3	(2)1
Robustness	(3)3	(3)3	(5)3
Availability	(3)4	(2)3	(3)2
Total	(18)**23**	(18)18	(23)14

Table 9.1: Comparison of different methods

In Table 9.1, we can see the results for the performance and value proposition of each technology. This table is smaller than Table 2.1, because we can only compare the tested technologies. The values are rated from 1(worst) to 5(best) and the values in between parenthesis are the values from the theoretical point of view. The worst performer in comparison to theory was the TI mmWave. We expect this to happen mostly due to the added complexity of this device, which is most probably better used when building from scratch and not just trying to adapt already available demo software. Both the Thermal Camera proposal and ToF had their advantages. For car counting purposes, the ToF presented a simple solution achieving over 86% accuracy, as seen in Chapter 5. For indoor counting, the ToF could be correlated positively with the air quality. We noted that it could correctly monitor people's entrances in sparse situations, despite having more difficulties when multiple people were leaving the room at the same time. Even then, it retained its ease of use and low cost. With most of these solutions, only a Raspberry Pi Zero W is required to post the data to the web (due to the low processing power needed) and a microcontroller capable of handling the ToF data. The Raspberry Pi Zero W can even be omitted if using a microcontroller with some IoT capabilities, such a Narrow Band IOT (nb-IOT) support or SigFox, for example. The Thermal Camera solution is the best choice if the exact people count is desired, since the use of adaptive thresholding via room temperature yielded some good results. Once again, we obtained over 86% of people accurately counted. This number can be further improved if more data sets are used to fine tune the thresholds to the room temperature where the sensor is being installed. Being a thermal sensor, it is limited by the environmental heat sources that are hard to avoid, such as sunlight reflection or high laptop temperatures, creating a large hotspot. Finally, we can address our research question. **A low cost occupancy detection solution can be designed, with an accuracy up to 86%.**

9.2 Future Work

While the TI mmWave did not achieve the results we expected, we can see that there is a lot of potential with this technology, but it requires fine tuning of the code and its parameters to get the best performance out of it. Furthermore, the best use case is clearly in cars (or bigger objects) in outdoor environments, due to less reflections from walls.

For the TOF system, we considered it achieved a good performance both in car counting and people counting. The car counting system could be improved by using a sensor that allows to count direction, much like the sensor we used on the people counting system. This would allow the install to monitor the entrances in the parking garage of an apartment, for example. For people counting, we think that perhaps using a faster microprocessor would allow for reduced delays, making the zone switching faster, causing less error in the readings.

The thermal camera system proved to be able to monitor occupancy without many flaws and with further tweaks in the algorithm plus bigger datasets, its accuracy can only get better. The next logical step would be to remove the Raspberry Pi and install the MLX90640 sensor on a microcontroller with some IoT connectivity, sending the thermal frame to a cloud server while processing it on the server. This would allow the device to be battery powered, making the install of this system in multiple conference rooms easier.

Bibliography

[1] D. Fehr, R. Sivalingam, V. Morellas, N. Papanikolopoulos, O. Lotfallah, and Y. Park, "Counting people in groups," *6th IEEE International Conference on Advanced Video and Signal Based Surveillance, AVSS 2009*, pp. 152–157, 2009. [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp={&}arnumber=5279591https://ieeexplore.ieee.org/abstract/document/5279591

[2] K. Akkaya, I. Guvenc, R. Aygun, N. Pala, and A. Kadri, "IoT-based occupancy monitoring techniques for energy-efficient smart buildings," *2015 IEEE Wireless Communications and Networking Conference Workshops, WCNCW 2015*, pp. 58–63, 2015. [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp={&}arnumber=7122529https://www.researchgate.net/publication/228541268{_}Real-time{_}vision-based{_}people{_}counting{_}system{_}for{_}the{_}security{_}door

[3] K. F. Salama, R. F. Alhajri, and A. A. Al-anazi, "Assessment of air quality in bus terminal stations in Eastern Province , Kingdom of Saudi Arabia," vol. 4, no. 5, pp. 1413–1418, 2017. [Online]. Available: https://www.researchgate.net/publication/316484327{_}Assessment{_}of{_}air{_}quality{_}in{_}bus{_}terminal{_}stations{_}in{_}Eastern{_}Province{_}Kingdom{_}of{_}Saudi{_}Arabia

[4] S. Meinardi, P. Nissenson, B. Barletta, D. Dabdub, F. Sherwood Rowland, and D. R. Blake, "Influence of the public transportation system on the air quality of a major urban center. A case study: Milan, Italy," *Atmospheric Environment*, vol. 42, no. 34, pp. 7915–7923, 2008. [Online]. Available: http://dx.doi.org/10.1016/j.atmosenv.2008.07.046

[5] I. Medicine, "a Survey of Indoor Air Quality Within Public Transport," pp. 802–805, 2005. [Online]. Available: http://www.airitilibrary.com/Publication/alDetailedMesh?docid=WFHYXW178935

[6] M. Weng and X. Jin, "Study on the air pollution in typical transportation microenvironment: Characteristics and health risks," *Journal of the Air and Waste Management Association*, vol. 65, no. 1, pp. 59–63, 2015. [Online]. Available: http://dx.doi.org/10.1080/10962247.2014.962648https://doi.org/10.1080/10962247.2014.962648

[7] J. W. Choi, S. H. Cho, Y. S. Kim, N. J. Kim, S. S. Kwon, and J. S. Shim, "A counting sensor for inbound and outbound people using IR-UWB radar sensors," *SAS 2016 - Sensors Applications Symposium, Proceedings*, pp. 528–532, 2016. [Online]. Available: https://ieeexplore.ieee.org/document/7479903

[8] K. Hashimoto, K. Morinaka, N. Yoshiike, C. Kawaguchi, and S. Matsueda, "People count system using multi-sensing application," *International Solid State Sensors and Actuators Conference*, vol. 2, pp. 1291–1294, 1997. [Online]. Available: https://ieeexplore.ieee.org/document/635472

[9] I. J. Amin, A. J. Taylor, F. Junejo, A. Al-Habaibeh, and R. M. Parkin, "Automated people-counting by using low-resolution infrared and visual cameras," *Measurement: Journal of the International Measurement Confederation*, vol. 41, no. 6, pp. 589–599, 2008. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0263224107001078

[10] K. Terada, D. Yoshida, S. Oe, and J. Yamaguchi, "A counting method of the number of passing people using a stereo camera," *IECON'99. Conference Proceedings. 25th Annual Conference of the IEEE Industrial Electronics Society (Cat. No.99CH37029)*, vol. 3, pp. 1318–1323. [Online]. Available: http://ieeexplore.ieee.org/document/819402/

[11] A. Bevilacqua, L. Di Stefano, and P. Azzari, "People tracking using a Time-of-Flight depth sensor," *Proceedings - IEEE International Conference on Video and Signal Based Surveillance 2006, AVSS 2006*, 2006. [Online]. Available: https://ieeexplore.ieee.org/document/4020748

[12] H. Li, E. C. Chan, X. Guo, J. Xiao, K. Wu, and L. M. Ni, "Wi-Counter: Smartphone-Based People Counter Using Crowdsourced Wi-Fi Signal Data," *IEEE Transactions on Human-Machine Systems*, vol. 45, no. 4, pp. 442–452, 2015. [Online]. Available: https://ieeexplore.ieee.org/document/7155631

[13] Y.-k. Cheng and R. Y. Chang, "Device-Free Indoor People Counting Using Wi-Fi Channel State Information for Internet of Things," 2017. [Online]. Available: https://ieeexplore.ieee.org/document/8254522

[14] L. Del Pizzo, P. Foggia, A. Greco, G. Percannella, and M. Vento, "Counting people by RGB or depth overhead cameras," *Pattern Recognition Letters*, vol. 81, pp. 41–50, 2016. [Online]. Available: http://dx.doi.org/10.1016/j.patrec.2016.05.033

[15] A. Wojtkiewicz and R. Rytel-Andrianik, "Optimal detection and estimation in FMCW radar," *14th International Conference on Microwaves, Radar and Wireless Communications, MIKON 2002*, vol. 3, pp. 778–781, 2002. [Online]. Available: https://ieeexplore.ieee.org/document/1017956

[16] L. U. A. S. Publications, *State-of-the-Art Review on Automotive Radars and Passive Radar Reflectors Arctic Challenge research project*, 2018. [Online]. Available: https://www.lapinamk.fi/loader.aspx?id=983f39aa-c97f-4f2d-bb39-abe006c6bad3

[17] "Ti mmwave people counting through plastic and drywall and wood," http://dev.ti.com/tirex/#/?link=Software%2FmmWave%20Sensors%2FIndustrial%20Toolbox%2FExperiments%2FPeople%20Counting%20through%20Plastic%20and%20Drywall%20and%20Wood, accessed: 2019-01-04.

[18] K. P. Lam, M. Höynck, B. Dong, B. Andrews, Y. S. Chiou, R. Zhang, D. Benitez, and J. Choi, "Occupancy detection through an extensive environmental sensor network in an open-plan office building," *IBPSA 2009 - International Building Performance Simulation Association 2009*, pp. 1452–1459, 2009. [Online]. Available: https://pdfs.semanticscholar.org/8ee8/e6650e2b22e8dbd3ec7ba75971d53bc48111.pdf

[19] T. Ekwevugbe, N. Brown, and D. Fan, "Proceedings of the 2012 6th IEEE International Conference on Digital Ecosystems and Technologies: Complex Environment Engineering, DEST 2012," *IEEE International Conference on Digital Ecosystems and Technologies*, 2012. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/6227924{#}

[20] S. S. D. Almeida, V. Hugo, and C. D. Melo, "An Evaluation of two People Counting Systems with Zenithal Camera." [Online]. Available: http://www.decom.ufop.br/menotti/rp102/TrabalhoFinal-papers/01-PeopleCounting.pdf

[21] C. Iovescu, R. A. Manager, and S. Rao, "The fundamentals of millimeter wave sensors," 2017. [Online]. Available: http://www.ti.com/lit/wp/spyy005/spyy005.pdf

[22] "Ti mmwave people counting demo," https://training.ti.com/people-counting-demonstration-using-ti-mmwave-senso

[23] L. Keselman, J. I. Woodfill, A. Grunnet-Jepsen, and A. Bhowmik, "Intel RealSense Stereoscopic Depth Cameras," 2017. [Online]. Available: http://arxiv.org/abs/1705.05548

[24] P. Zhao, C. X. Lu, J. Wang, C. Chen, W. Wang, N. Trigoni, and A. Markham, "mID: Tracking and Identifying People with Millimeter Wave Radar," pp. 33–40, 2019. [Online]. Available: http://www.cs.ox.ac.uk/files/10889/{%}5BDCOSS19{%}5DmID.pdfhttps://ieeexplore.ieee.org/document/8804831

[25] [U+FFFD] P. Glavonjić, I. M. Milosavljević, D. P. Krčum, V. R. Mihajlovic, and L. V. Saranovac, "Person detection counter based on mm-Wave radar technology," *IcETRAN*, pp. 4–7, 2017. [Online]. Available: https://www.etran.rs/common/pages/proceedings/IcETRAN2017/ELI/IcETRAN2017{_}paper{_}ELI3{_}3.pdf

A

Code of Project

Due to being composed of multiple code pieces, code of the project can be found in the following link: https://drive.tecnico.ulisboa.pt/download/1695927966738046