Universidade de Aveiro
2018

Departamento de Eletrónica,
Telecomunicações e Informática

Rui Gonçalo Barreto
Abrantes Bóia

Pedestrian tracking and re-identification on top-view multi-camera system

Seguimento e re-identificação de pessoas num sistema multi-camera com vista superior

Universidade de Aveiro
2018

Departamento de Eletrónica,
Telecomunicações e Informática

Rui Gonçalo Barreto Abrantes Bóia

Pedestrian tracking and re-identification on top-view multi-camera system

Seguimento e re-identificação de pessoas num sistema multi-camera com vista superior

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia de Computadores e Telemática, realizada sob a orientação científica do Professor Doutor Paulo Miguel de Jesus Dias, Professor auxiliar do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro, e com colaboração do Jorge Amador e Victor Abreu, da empresa WithUs.

o júri / the jury

presidente / president

Professor Doutor António José Ribeiro Neves
Professor Auxiliar, Universidade de Aveiro

vogais / examiners committee

Doutor João Manuel Leite da Silva
Investigador Sénior, Altran Portugal (Arguente)

Professor Doutor Paulo Miguel de Jesus Dias
Professor Auxiliar, Universidade de Aveiro (Orientador)

acknowledgements

To my thesis advisor, Professor Paulo Dias, for the support and motivation that he gave me during this year. I would also like to thank Jorge Amador and Victor Abreu from the company WithUs for the collaboration and interest that they showed during this work, that interest also kept me motivated. And finally, to my family and friends for the continuous support shown throughout my whole academic life.

palavras-chave

Visão por computador, Re-identificação, seguimento de peões.

resumo

Ter alguma noção dos padrões de movimento dentro de centros comerciais e áreas de venda a retalho é uma informação cada vez mais procurada nos dias de hoje, com vários sistemas já desenvolvidos para contagem de pessoas e alguns com a possibilidade de acompanhar o percurso dessas pessoas dentro do estabelecimento comercial.

Esta dissertação tem como objectivo desenvolver um sistema que permita contar o número de pessoas que passa numa dada àrea, juntamente com o percurso feito por cada pessoa, mas principalmente conseguir re-identificar pessoas entre câmeras, com o objectivo de permitir detetar e analisar padrões de movimento em centros comerciais ou aplicações semelhantes.

O trabalho foi dividido em quatro partes importantes: o estudo dos sistemas já no mercado e dos algoritmos disponíveis; definição do caminho a seguir para atingir os objectivos; captura de vídeos adequados de forma a simular situações o mais realistas possíveis; testar os algoritmos e validar os resultados obtidos.

Os resultados indicam que este sistema permite extrair informação interessante relativamente a padrões de movimento e re-identificação de pessoas, no entanto o sistema pode ter de ser adaptado para sistemas com um número muito mais elevado de passagens.

keywords Computer vision, Re-identification, pedestrian tracking.

abstract

Understanding movement patterns inside retail areas is an information that is very useful nowadays, with several systems already developed to count and track pedestrians in those areas.

The goal of this dissertation is to develop a system capable of counting, tracking, but mainly re-identifying pedestrians across multiple cameras with top-view, so that we can detect and analyze movement patterns in retail areas or similar applications.

The work was divided in four important stages: the study and understanding of various systems that are already developed and the available algorithms; choosing the path to take in order to complete the proposed goals; video capture of adequate footage in order to simulate real scenarios as best as possible; test the developed algorithms and validate the obtained results.

Results show that this system is capable of extracting interesting information regarding movement patterns and also capable to re-identify pedestrians across multiple cameras. However, this system might need to be adapted for real life scenarios with a bigger number of pedestrian passages.

I. Index

II. List of figures

III. List of tables

1 Introduction

1.1 Motivation and objectives

In various scenarios (such as in retail or shopping malls for example), it is important to understand the patterns of movement of the costumers [Lebied, 2017], namely the most followed paths, and the amount of time spent in different areas.

This information is valuable since it allows operations to understand and improve product placement based on the movement patterns of their costumers [Business Reporter, 2016].

Even though this information is very useful, most of the times it conflicts with privacy [Dwoskin & Timberg, 2017], and this is a very important issue that must be addressed in a retail scenario.

The main purpose of this dissertation is to study the viability of acquiring information related to patterns of movement using a top view of the pedestrians. To gather that information across multiple cameras it is necessary to re-identify pedestrians across those cameras which is the main challenge of this thesis.

With that said this thesis is divided in three main objectives: Counting and Tracking within a single area of coverage; Counting and Tracking in scenarios with two cameras with areas of coverage with some overlap; And Counting and Tracking when we have two cameras covering different areas, using re-identification to associate pedestrians between cameras.

Since our main focus is to emulate retail scenarios, we will be capturing scenarios where the cameras are positioned between 5 and 7 meters above the ground and low light variation since in retail scenarios lighting is controlled. Also, the fact that the view of the scene is top view minimizes the risk of having pedestrians in front of other pedestrians, a situation also known as occlusion, situation which we will not focus on this thesis.

This thesis was developed within a collaboration between the University of Aveiro and the company WithUs who is interested in evaluating the technical viability of such a monitoring system.

1.2 Structure

This dissertation is divided in six chapters: Introduction, Pedestrian Tracking Systems and Studies, Pedestrian Tracking, Pedestrian Re-Identification, Results and Conlcusions and Future Work.

In this chapter we go into a brief explanation of our motivation, we also specify our main objectives for this thesis as well as our case study.

Chapter 2 is where we focus on what is currently available regarding pedestrian tracking and re-identinfication, including systems currently in the market and studies into pedestrian re-identification. We also take a first look into our proposed approach to the problem at hand, and we briefly tackle the differences between the existing systems and our approach. To end chapter 2 we take a look at the important characteristics to take into account when choosing the right equipment to capture the necessary images.

After defining the scenarios based on the information presented in chapters 1 and 2, chapter 3 focuses on our selected approach to track and count pedestrians. We start by taking a look at existing algorithms for the different steps of this process, and then we present our selected approach. We describe every step of our tracking algorithm and we present some additional tools developed in order to better visualize information present in the images. At the end of the chapter we present some preliminary results regarding the tracking and counting within a single image, and with two images with some overlap in the areas covered.

Chapter 4 is the chapter where re-identification comes into play. As always, we first take a look at some possible approaches, challenges and limitations for our problem, before describing our selected approach. In this chapter we will continue our preliminary tests, and we will present the first results of our re-identification approach.

The results of our final tests are shown in chapter 5 where we describe the conditions of each test, the problems found and possible solutions to those problems.

Finally, in chapter 6, we present some final thoughts and future work.

2 Pedestrian tracking systems and studies

In this chapter we present a brief review of existing algorithms and systems used to perform pedestrian tracking.

We will also look at some studies that try to solve the problem with pedestrian re-identification, as well as their potential to our application.

2.1 Existing vision-based tracking systems

There are various vision-based systems already created for retail that track pedestrians, and provide information based on this tracking.

We will take a closer look at some of these systems and analyse their pros and cons regarding their viability to our proposed solution.

2.2 Depth camera

Depth cameras (**Figure 1**) are structured light systems (**Figure 2**).

These cameras are often used in retail tracking systems because depth information is used to filter the scene and the objects to be tracked based on their height allowing to easily filter noise, such as shopping carts or toddlers, and a target height can be defined to track objects above that threshold, eliminating everything that is below those heights.

The main problem with these systems is the working distance. Usually these cameras can only give reliable information for relatively short distances, between 1 and 3 meters, making them impractical for systems where the cameras are at a significant height in relation to the pedestrians.

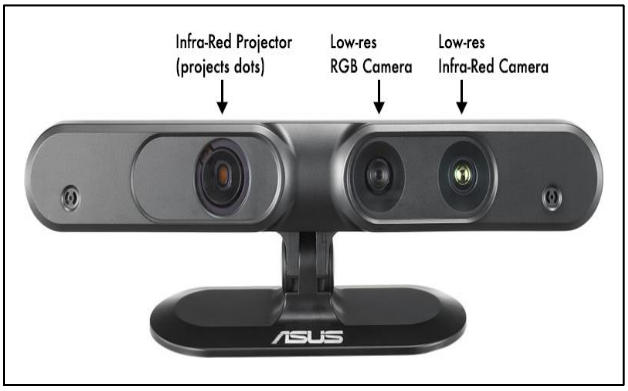


Figure 1 Depth Camera [McWilliams, 2013].

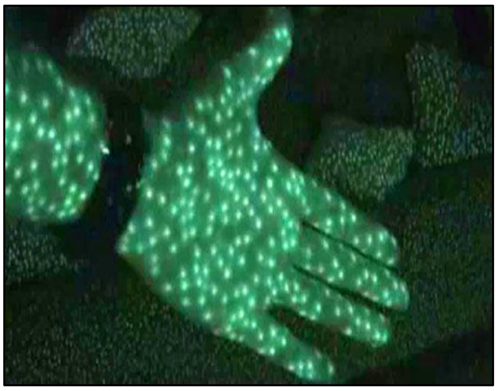


Figure 2 Example of known infra-red pattern projected [GSMArena.com, 2017].

2.3 Time of Flight camera

Time of Flight cameras (**Figure 3**) emit light at the scene at a given frequency, the light will travel, reflect on every object on the scene, and the reflection will be captured by the camera (**Figure 4**). Every pixel from the image will have information related to the time it took for the light to travel and hit that area of the scene.

Like the depth camera systems, the Time of Flight sensors provide depth information, but can give reliable information at a larger distance making it a better solution for systems that capture that information from a higher distance [Stahlschmidt et al., 2013].

However, Time of Flight cameras are an expensive option for a retail scenario where you must cover a big area, and they only rely on depth information which is not enough for pedestrian re-identification. Results may also be affected by other light sources, especially if there are more Time of Flight cameras nearby, but in our case that would not be a problem.

Figure 3 Time of Flight Camera [Multipix Imaging, 2017]

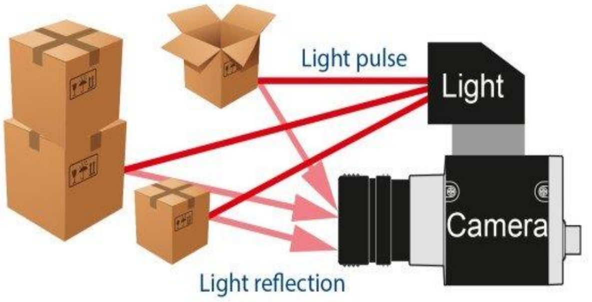


Figure 4 The light emitters around the camera will emit the light onto the scene, and the light will reflect on the objects present in the scene [Multipix Imaging, 2017]

2.4 Image based

Image based systems rely on image processing to extract information from a given image or live feed.

These systems rely, mostly, on the information that is present in the image, and thus the results obtained by these systems are easily influenced by the selected algorithm used to process the acquired image. This solution is interesting for various systems, since it can be adapted to the problem at hand, making these systems highly adaptable to different scenarios.

Most of the systems available in the market that use this approach only track pedestrians in a single image, and none of them deal with pedestrian re-identification between images.

In terms of pedestrian tracking the most common first step to tackle the problem without depth information is background removal (**Figure 5**). Based on the results of background removal there are multiple solutions to track the extracted areas across frames. The most used solution is to use tracking algorithms that only need to be given the areas extracted by the background removal (**Figure 6**). These trackers usually extract features from the various areas and try to correlate those features between frames.

Image based methods are more flexible and cheaper than the previous options, which may be used at 5-7 meters without a major loss of information.

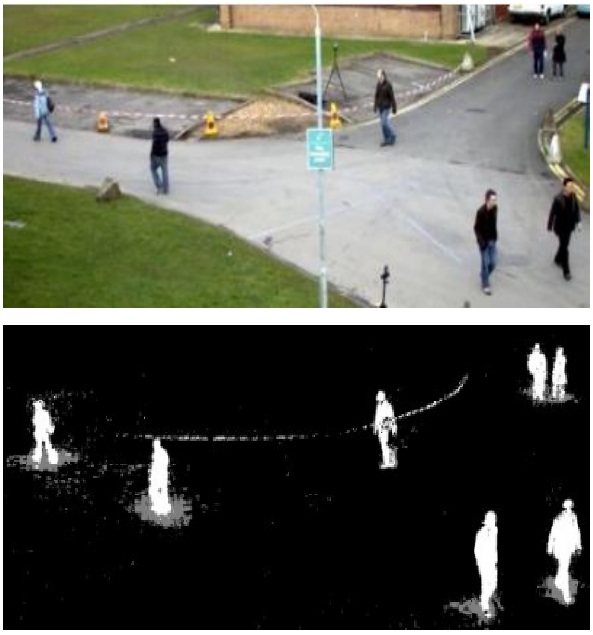


Figure 5 Example of background and foreground segmentation [Docs.opencv.org, 2015].

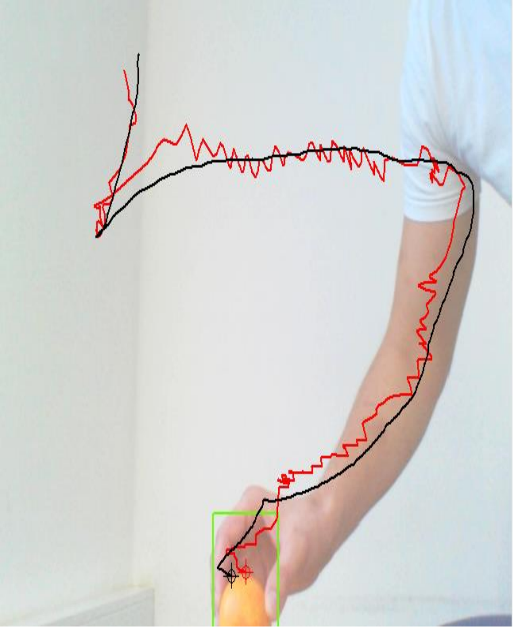


Figure 6 Example of a yellow ball being tracked between frames [Forums.ni.com, 2016].

2.5 Other systems that track pedestrians

There are other systems that track pedestrians without relying on a vision-based approach, usually by using signals and tracking the location of those signals.

2.5.1 Wireless

These systems use wireless receptors placed around the desired location.

The receptors capture all the wireless signals emitted within a given range and since multiple wireless receptors are placed on the location, it is possible to triangulate the location of the wireless devices in the area.

One drawback of this solution is that it does not provide reliable information in terms of all pedestrians present in that location: there is no guarantee that a given pedestrian uses an activated wireless. Also, the wireless signal can be affected by other conditions in the environment, for example other electronic systems, making the results even less reliable.

2.5.2 Radio Signal

These systems use radio signal receptors placed in a defined location.

These systems are more reliable in terms of pedestrian counting, since almost all pedestrians have cell phones. On the other hand, they also suffer from interference from other electronic systems that can affect the radio signal, making it very hard to find the location of the radio devices with a good degree of precision.

2.6 Existing Commercial Systems

Although no commercial systems solve completely the problem of pedestrian tracking in large areas, there are several systems available in the market that also provide some of the necessary functionalities in well-defined conditions. Here are some examples:

- Atlantic InStore Tracking [Tlantic.com, 2017] – uses both radio signals and wi-fi to extract information about movement and number of clients in a given store. As stated before radio signals and wi-fi approaches are prone to errors, since those signals might not be available for certain costumers.
- Amazon Go [Amazon.com, 2016] – this system uses both wireless signals and computer vision to track clients and get reliable information about what happens in the store. The system can identify the clients but require an authentication process when entering the store. The authentication is done by reading a QR code, generated by the Amazon Go app, from the cell phone and the system needs the cell phone wireless to track the client within the store [Youtube, 2016]. Since the client knows that the wireless is required in order to use the store, this approach might wield good results both in tracking and identifying the same client in different cameras.
- Xovis [Xovis.com, 2013] – uses stereo vision. The system can count and track within a single image, and it can also track pedestrians across multiple images if they have overlap. If the pedestrian leaves a given zone he will not be identified as the same pedestrian when he enters it again.
- ShopperTrak [ShopperTrak, 2010] – uses the same principle as Xovis (stereo-vision) counting and tracking pedestrians across a common area [Youtube, 2010].
- Mobotix MxAnalytics [Artsensor, 2013] – Uses an Image based approach and a fisheye camera sensor in order to cover a larger area. Similar to the previous systems, it is only concerned with tracking and counting pedestrians in the area covered by the camera [Youtube, 2013] and does not provide any re-identification capabilities.

2.7 Pedestrian Re-Identification

In terms of re-identification, most systems and studies are based on a side view of the pedestrians.

One of the studies [Kwangchol Jang et al., 2013] analyses the pedestrian colours, identifies the most common colours and uses that to try and recognize the same person in a different image. It also segments the person into two different parts, the upper body and the lower body, obtaining the dominant colours in each different area by calculating the HSV histogram of the pedestrian image. The mentioned study however, uses a side view of the pedestrians.

The system in [Nakatani et al., 2012] uses an overhead view to identify a person, but for it to work the camera must be close enough or capture an image with enough detail to be able to distinguish the hair pattern, that is supposed to be unique for each person.

2.8 Proposed solution

Based on the information presented in this chapter we decided to use an image-based approach since it allows for a higher degree of adaptability. The outcome is only limited by the information present in the image and the algorithms used to extract that information.

The two studies presented regarding pedestrian re-identification are not possible solutions for our problem, since the first deals with a side view of the pedestrian, which wields very different results and information from a top view (**Figure 7**), and the second needs the camera to be close enough to the pedestrian's head.

And that is where the top view becomes a challenge, since most of the common approaches used on a side view wield very different results when applied to the top view. However, this does not mean that we cannot use those studies as basis to find a fitting approach for the challenge with the top view, as we will see in future chapters.

There are three main distinct phases to reach the solution, pedestrian tracking and detection within a single camera, pedestrian tracking and detection with multiple cameras with overlapping areas, and pedestrian tracking, detection and re-identification between multiple cameras with no overlapping area.

During the pedestrian tracking phase within a single camera we will only focus on detecting the pedestrians and tracking their movement within the camera, as well as counting how many pedestrians moved within the camera's field of view.

After the pedestrian tracking phase is done we will attempt to pass the information obtained by one camera, to a different camera that has an overlap area with the first camera. This will allow for multiple cameras to be used to cover a wider area and at the same time track pedestrians across those cameras.

Finally, we will remove the overlap between cameras and we will attempt to re-identify on the second camera the pedestrians that were identified on the first camera. With this, there is no need for cameras to have overlap areas, and we can track pedestrians across multiple cameras spread around a given area.

The tool used for this study is going to be OpenCV, a multi-platform library that is used in the field of computer vision.

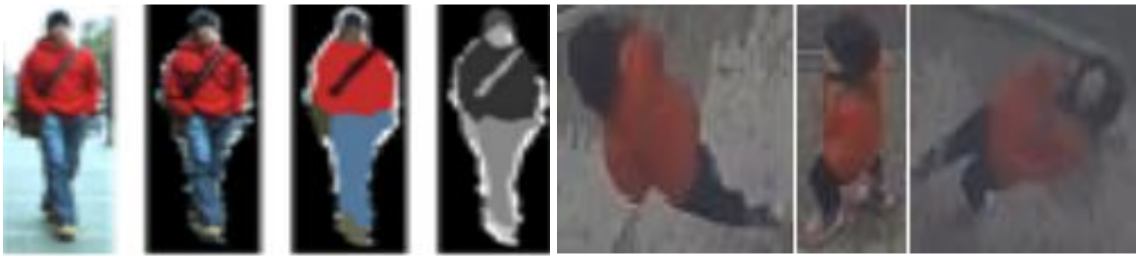


Figure 7 Side view [Kwangchol Jang et al., 2013] and top-view.

2.9 Cameras

There are characteristics present in cameras that we need to take into account when choosing the right camera for our work.

2.9.1 Camera sensor

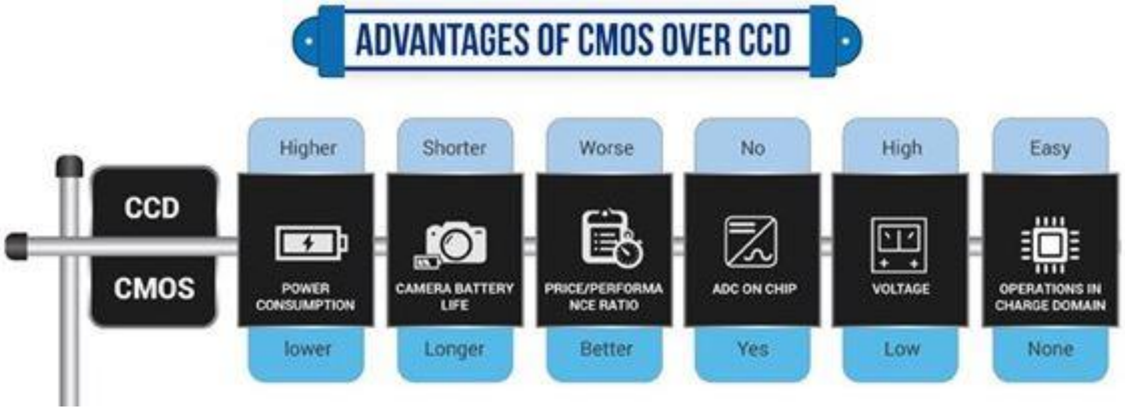
Camera sensor is the most important characteristic on a camera, since the resulting image is highly influenced by the sensor. Within the camera sensor there are other characteristics that are useful for different scenarios, for example, sensor size and resolution.

The most common camera sensors are the Charged-Coupled devices (CCD), and the Complementary Metal-Oxide Semiconductor (CMOS). Both have their advantages and disadvantages over the other.

The CCD sensor has separate sensors for the red, blue and green channels, making the images more accurate and sharper, which is very useful if we want to extract more information from the image.

As for the CMOS sensor, it has a bigger sensor size, which is a characteristic very important for scenarios where we want to have a bigger area of coverage without using a lot of cameras, and it also consumes less energy than the CCD sensor.

Based on this analysis, the most interesting sensor for our application is the CMOS sensor, since we will be working from a significant distance to the ground, making the CCD image quality not so relevant. The CMOS sensor will be able to cover more area with less cameras while also allowing it to acquire images for a longer period without the need to recharge the battery which is and interesting feature in a scenario where wiring is not possible, and the camera should run on batteries (**Figure 8**).



ADVANTAGES OF CMOS OVER CCD

	POWER CONSUMPTION	CAMERA BATTERY LIFE	PRICE/PERFORMANCE RATIO	ADC ON CHIP	VOLTAGE	OPERATIONS IN CHARGE DOMAIN
CCD	Higher	Shorter	Worse	No	High	Easy
CMOS	lower	Longer	Better	Yes	Low	None

Figure 8 Overview of the advantages of a CMOS sensor over a CCD one [Image-sensors-world.blogspot.com, 2016].

2.9.2 Sensor Dimensions

Sensor dimensions are the most important characteristics necessary to select an adequate camera for each scenario, since it influences the field of view that a camera can cover.

Field of view (FoV) is calculated based on focal length and sensor size. Sensor size, as the name implies, is the size of the sensor, and focal length is the distance between the sensor and the lens.

Since the angle of coverage is the same as the angle between the camera lens and the two sides of the sensor (visually explained in the next image) (**Figure 9**), the bigger the sensor the bigger the FoV gets, and the bigger the focal length, the smaller the FoV gets (**Figure 9**). So, for a big FoV we want a big sensor coupled with a small focal length.

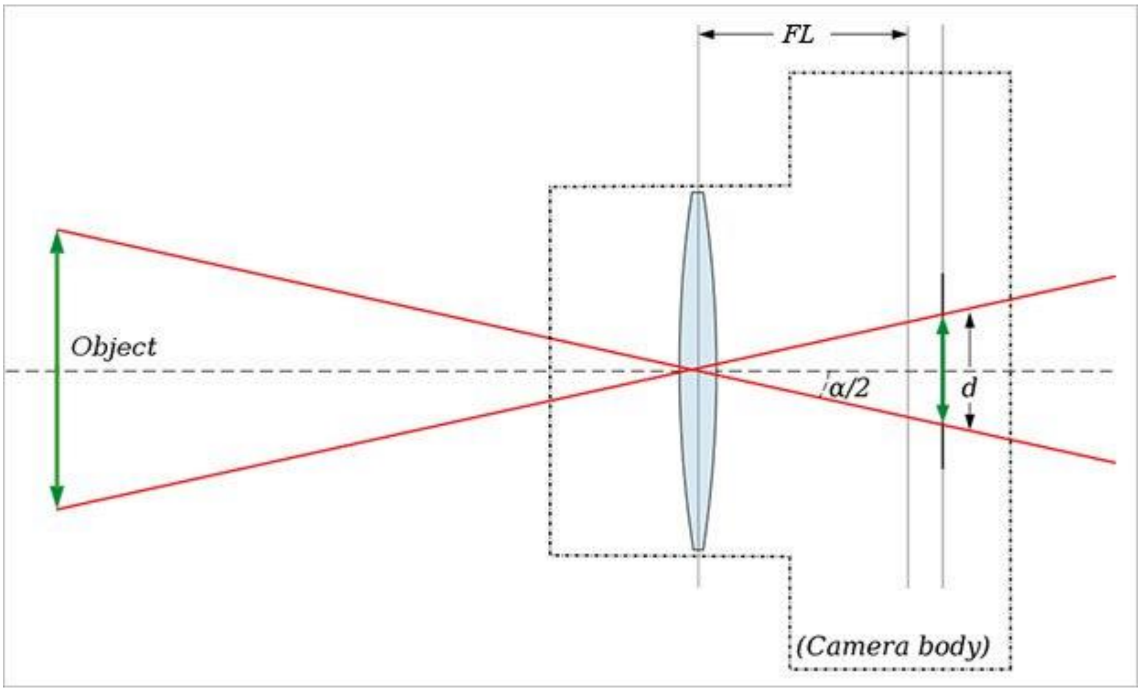


Figure 9 Angle of view formed from the combination of the focal length and the sensor size [PetaPixel, 2013].

2.9.3 Resolution

Resolution is the most known characteristic in a camera, since it represents the quantity of information present in the image. It is the quantity of pixels present in the image, and the more pixels in the image, the more detail we will get from the image.

2.9.4 Height and area to cover

Since we decided that our study case would deal with a situation where the cameras are positioned between 5 and 7 meters from the ground, and that we would want to cover as much area as possible, we need to have a camera with a good field of view and enough resolution so that we can extract more information from the acquired images.

Let us say that we want to cover an area of 25 meters horizontally and 15 meters vertically, and that our camera is positioned 7 meters above the ground. In that case, how many cameras would we need to cover that area, and how much area would each camera cover? If we assume that our cameras have a 1/2.3" sensor with 6,17mm width and 4,55mm height, and a focal length of 2,98mm, then the area covered by one camera would be [PetaPixel, 2013]:

Focal Angle (H) = 2*Arctan (6,17/5,96) = 91^{o}

Focal Angle (V) = 2*Arctan (4,55/5,96) = 75^{o}

Distance Covered (H) = 2*(Tan (45,5) *7) = 14m

Distance Covered (V) = 2*(Tan (37,5) *7) = 10,64m

Since one camera covers 14 meters horizontally and 10,64 meters vertically, we would need 4 of these cameras to cover the desired area (**Figure 10**).

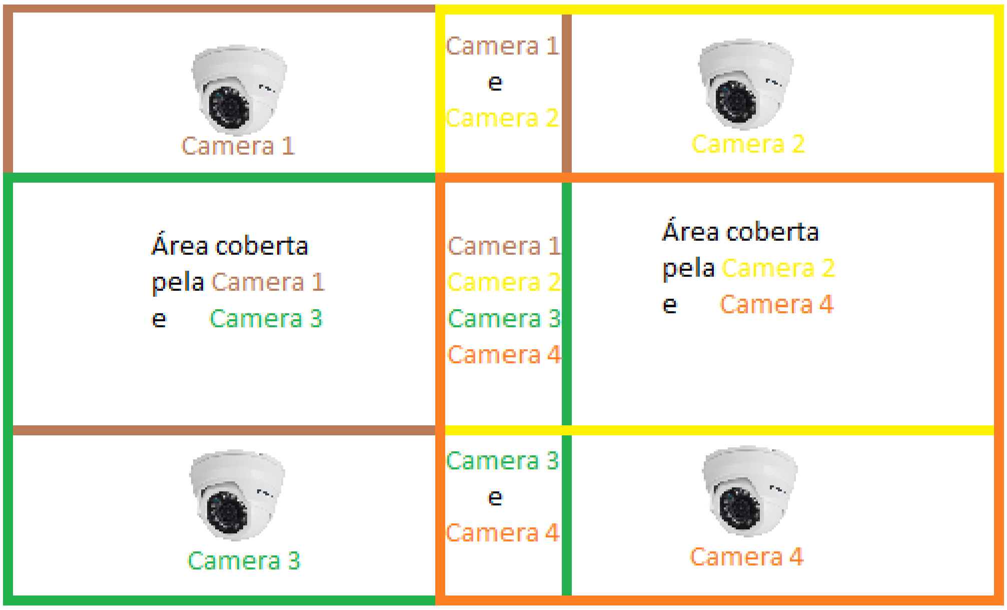


Figure 10 Area coverage of each camera, based on the calculations made.

2.9.5 Camera used

For our tests and having in consideration all the characteristics mentioned above, we decided to use two GoPro cameras for our scenarios, a GoPro Hero 3+ (**Figure 11**) and a GoPro Hero 4 (**Figure 12**).

We opted for these cameras since both have a very good field of view (**Figure 13**), based on the characteristics of the Hero 3+ [GoPro, 2017] we can say that at best we can cover an area of 25,2m x 15,12m, making it possible to cover very large areas with only the two of them, they are also very portable, and can capture with a good enough resolution for our study case.

Figure 11 GoPro Hero 3+ [Cameras, 2017].

Figure 12 GoPro Hero 4 [ebayimg, 2017].

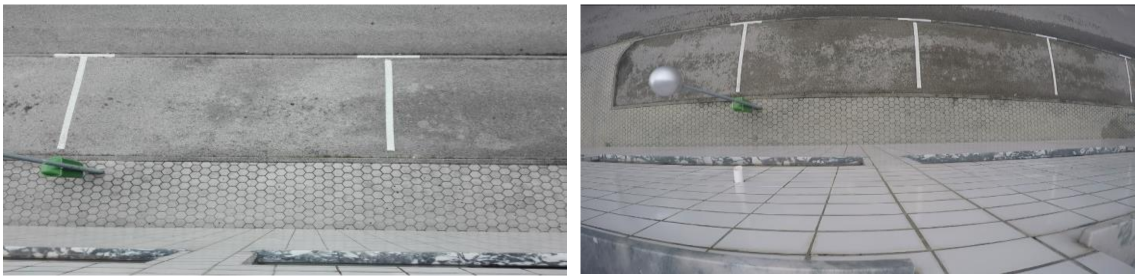


Figure 13 Area of coverage of a simple camera vs a GoPro (same area).

3 Pedestrian Tracking

This chapter describes the implemented tracking system developed.

We start by examining some of the algorithms that already exist for background and foreground segmentation as well as our chosen approach for extracting the relevant areas of the image.

After that, we look at possible approaches to track the extracted areas along the various frames of the video, and we go into detail describing our approach.

We will also look at ways to relay information between two different images, either by creating a single image with the two different feeds or finding the overlap area of both images and sharing information related to that area.

In the end we will finish this chapter with an explanation of two important visualization tools for our solution and the preliminary results obtained for counting and tracking within a single image and two images with an overlap area.

3.1 Background and Foreground Segmentation

Background and foreground segmentation is a pre-processing step that is used in a lot of computer vision-based systems. It allows for an extraction of moving objects in a given image, making it a good choice for tracking and counting.

Since we will be working with fixed camera positions and the pedestrians will be the only moving parts of the images this is the best approach for our solution.

OpenCV comes with three background subtraction algorithms [Docs.opencv.org, 2015], each with a different approach to separate the background from the foreground, but they all account for those factors.

3.1.1 MOG (Mixture of Gaussians)

MOG is an algorithm for background and foreground segmentation based on a mixture of Gaussian functions. It uses 3 to 5 Gaussian distributions to model each pixel considered as background. It also attributes a weight to the mixture dependent on the time that a given colour remains in the scene, because the more time a colour stays in the scene the more probable it is that it is a static object, and thus part of the background [Kaewtrakulpong & Bowden, 2001]. This algorithm already accounts for the presence of shadows, by assuming that if a pixel's colour stays the same but darker than that means that it is part of a shadow, making it more robust to shadow (**Figure 14**).

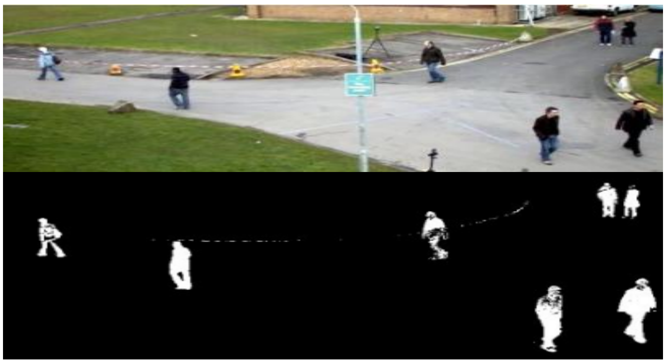


Figure 14 Example of MOG result [Docs.opencv.org, 2015].

3.1.2 MOG2 (Mixture of Gaussians 2)

MOG2 is a similar algorithm to the previous one, but with a slight improvement. While MOG uses a fixed number of Gaussian distributions to model the background pixels, MOG2 uses an adaptable number of Gaussian distributions for each pixel making it adaptable to changes in the scene and able to cope with light variations for example [Zivkovic, 2004] [Zivkovic & Van Der Heijden, 2006].

MOG2 also accounts for the presence of shadows, since it is also based on MOG, but it allows for the user to choose if he wants to detect shadows or not, representing the pixels that are part of the shadow with the grey colour (**Figure 15**), making it useful for situations where shadows might be important.

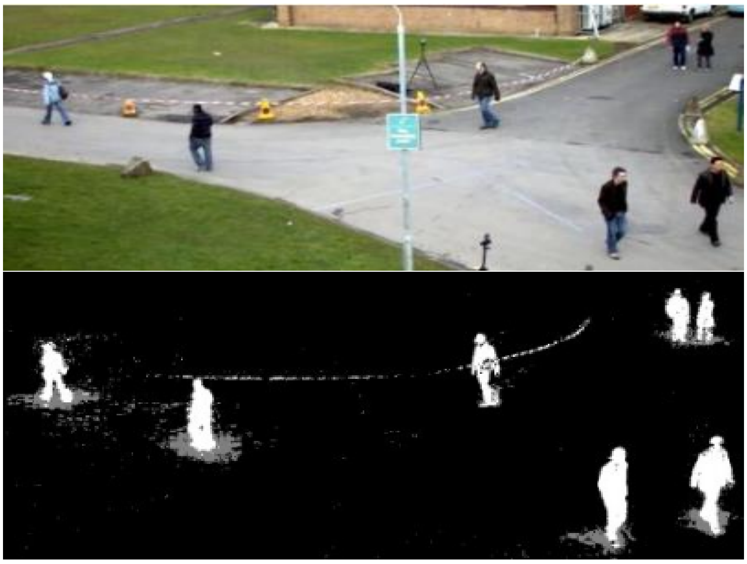


Figure 15 Example of MOG2 result [Docs.opencv.org, 2015].

3.1.3 GMG

GMG is an algorithm that uses a different method (**Figure 16**) from the previous ones, instead of using Gaussian mixtures to model background pixels, it uses a default number of frames to estimate the background and after that detects possible foreground objects by using Bayesian inference to estimate the probability of a given pixel being part of the foreground.

The information gathered from the newer frames has more weight in the decision-making process than the older one to deal with the light variation problem [Godbehere & Goldberg, 2014].

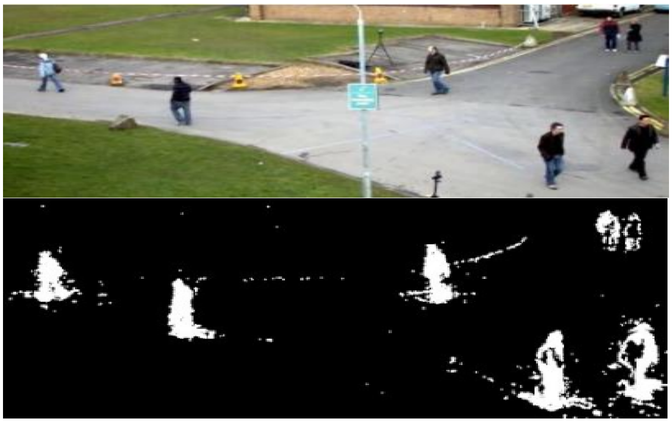


Figure 16 Example of GMG result [Docs.opencv.org, 2015].

3.2 Selected Algorithm

In the end the chosen algorithm for background and foreground segmentation was the MOG2 since it accounts for light variation, allows for shadow detection and does not need a default number of frames to pre-process the background, making it a good solution to process a set of frames or a video right away.

After extracting the foreground from the video, another algorithm from OpenCV is used to create a bounding box around each object. To create the bounding boxes representing the areas of interest within the frame, we find the contours of all the foreground objects and the left, right, top and bottom limits of each contour will define the bounding box of each object.

With the bounding boxes representing each object we can now use try and find those areas of interest in the next frames.

3.3 Object Matching

After detecting the areas of interest, it is necessary to associate objects between images: the tracking problem. There are multiple algorithms used to track objects after the areas of interest are defined and OpenCV provides the ones presented in this chapter.

3.3.1 Multi-Tracker

Multi-Tracker is an algorithm provided by OpenCV that allows for multiple object tracking between images. It only needs the detected areas of interest in a frame and a tracking algorithm, and it deals with the process of tracking them in the next frames.

The way it tracks those objects across multiple frames depends on the algorithm that the user chooses (**Figure 17**), it can be done by detecting key features within the areas of interest given and trying to match those features in the new frames, or by checking the neighbourhood of the areas of interest since it assumes that the object cannot move very far between pixels. When a given set of features is matched in a new frame it associates the previous area of interest that had those features with the new one.

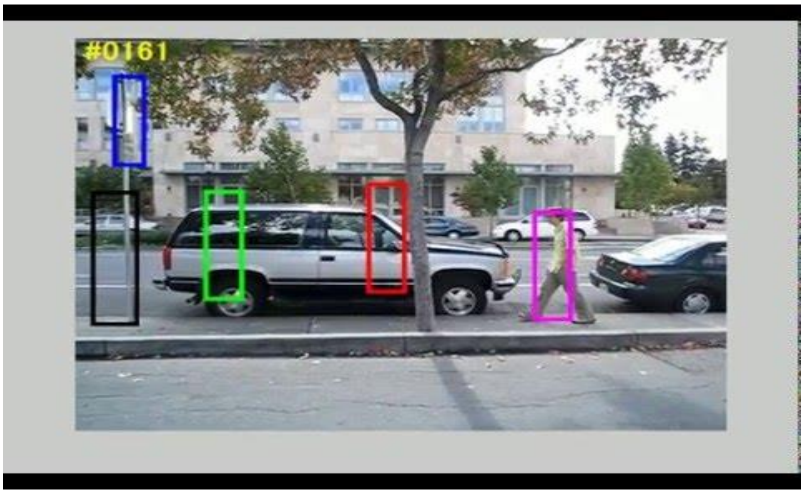


Figure 17 Representation of different results based on the algorithm used in the multitracker. Each rectangle with a different colour represents a different algorithm [Youtube, 2015].

3.3.2 Optical Flow

This method can also be used to track objects between frames. It detects apparent movement of a given object between frames using the notion that if a given point is moving between frames, then the neighbour points will also be moving in the same direction.

There are two main optical flow algorithms implemented in OpenCV.

The first one is the Lucas-Kanade [Lucas & Kanade, 1981]. method, where a set of points is given to the algorithm (most of the times this set of points are features extracted from an area of interest), and the algorithm forms a neighbourhood of 3 by 3 points and assumes that all those points will travel in the same direction between frames. It will calculate the location of those 9 points in the next frame and it will check if it finds a match in the next frame.

The second is known as dense optical flow and instead of using only a small set of points to track, it tracks all the points in the given frame, and forms a multi-channel image containing information related to the magnitude and direction of each pixel (**Figure 18**). With this information it calculates the probable location of the pixels in the next frame and checks if they match.

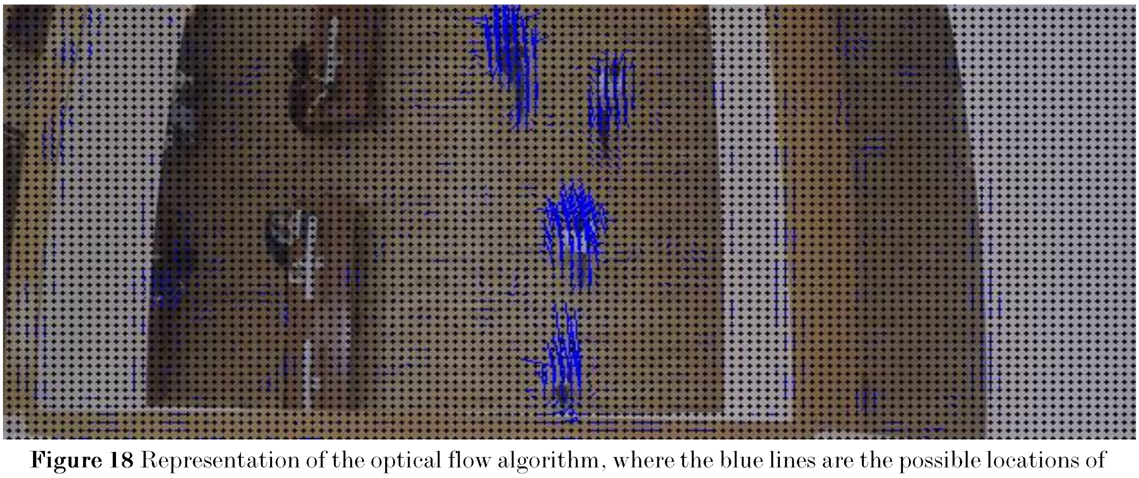


Figure 18 Representation of the optical flow algorithm, where the blue lines are the possible locations of movement on next frame.

3.3.3 Our approach

Since the areas of interest were already defined and the test images do not contain many pedestrians, none of the above methods were used to avoid extra complexity and a simple tracking method was instead implemented.

The implemented tracking method assumes that a person cannot have a large movement between frames. Since the areas of interest are calculated on every frame it is only necessary to save the areas of interest from the current frame to compare them with the next frame. The comparison is done by checking if a person bounding box in the current frame overlaps, by a threshold empirically defined, with a bounding box detected in the previous frame. If a bounding box overlaps with one from the previous frame we assume it is the same pedestrian, if it does not it will be considered as a new pedestrian.

This method is simple but wields good results since the number of pedestrians is low. It might need to be adapted in the future to work with a larger number of pedestrians or even groups of people.

3.4 Matching Pedestrians between overlapping cameras

One interesting set-up for pedestrian tracking involves a setting with several cameras with overlapping areas, allowing to cover wider areas of a given location. To relay the information of a given pedestrian to a neighbourhood camera without counting him as a new pedestrian it is necessary to find the overlapping area between cameras and correlate each pixel of the overlap area between images.

There are two possible solutions to this problem, stitching both images as one and doing the normal tracking method, or processing both images as separate images but relaying the information a pedestrian transitioning between images.

3.4.1 Stitching

OpenCV has a Stitcher class that given two images, stitches them together by matching similarities between them.

To be more precise, this algorithm splits the images into various sub-images and finds features on those sub-images. After finding the features the algorithm compares sub-images between the two images and tries to match them. Having matched the sub-images, the algorithm estimates the camera parameters to manipulate both images and create the final panorama image (**Figure 19**).

This method requires images with significant texture information, otherwise it will not find features to match between images and stitching will not be reliable. A possible fix to such situation is to manually correlate points between both images and feed those points to the algorithm making it possible for the algorithm to stitch both images.

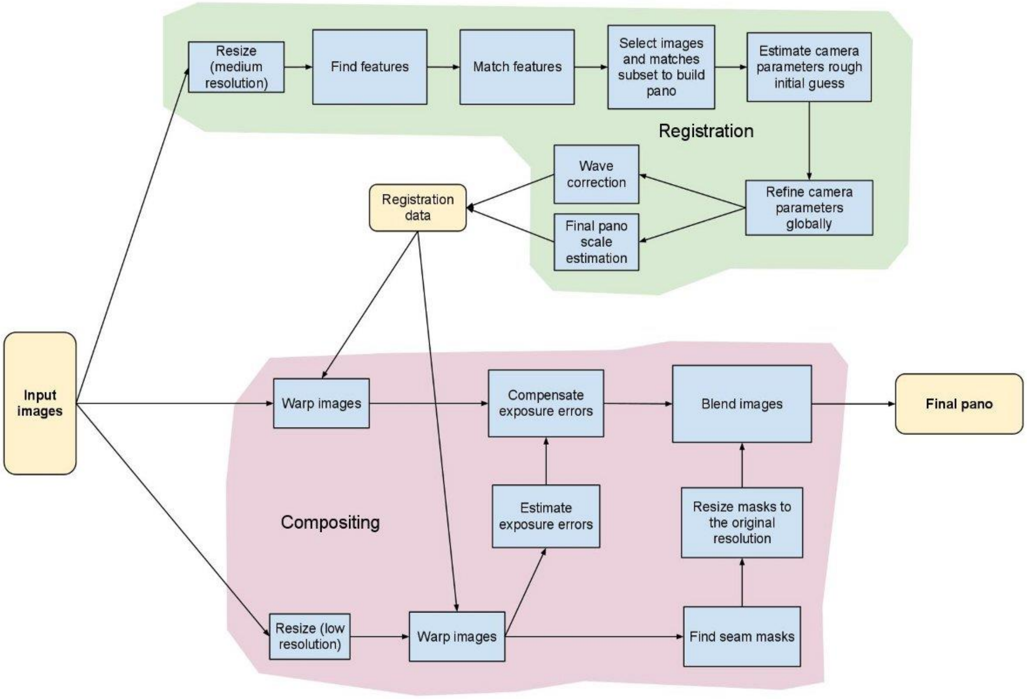

Resize (medium resolution)
Find features
Match features
Select images and matches subset to build pano
Estimate camera parameters rough initial guess
Registration
Wave correction
Refine camera parameters globally
Final pano scale estimation
Registration data
Input images
Warp images
Compensate exposure errors
Blend images
Final pano
Compositing
Estimate exposure errors
Resize masks to the original resolution
Resize (low resolution)
Warp images
Find seam masks

Figure 19 OpenCV stitcher class pipeline [Brown & Lowe, 2007].

3.4.2 Image correspondence

Another approach to the problem is to process both images as separate images in terms of tracking but relay the information between cameras when a pedestrian is crossing to the other image. This is done by finding common points between both images, done either by matching features or by manually matching enough points between images. Based on the correspondence it is possible to compute the transformation matrix necessary to match the points of one image with the points of the other.

The transformation matrix is a matrix that contains all the information related to translation, rotation and scaling applied to a given point to get its correspondence on the other image This is also called the affine transform and it contains the map of operations that need to be applied to a given set of points for those points to be matched on a different image. This operation is computed on OpenCV using the method findHomography [Kaehler & Bradski, 2016].

Once the transformation matrix is calculated, relaying information between cameras is straightforward. We only check if the corresponding point of the centroid of a bounding box is a point in the other image. If it is, then that point is inside the overlap area and it will be visible in both images.

3.5 Additional Data Visualization Features

Besides the algorithm, we also implement some tools to compute some statistics and show some information of the tracking results, namely tools for pedestrian counting and path visualization.

3.5.1 Pedestrian Counting

Counting pedestrian is a very common feature on many tracking systems nowadays. Knowing the number of visitors that walked by a store or that chose to walk towards a certain direction is valuable information in various fields of study. It can be used, for example, to manage article placements in stores, manage traffic flow and surveillance.

In terms of tracking, we can compare the count obtained by the algorithm with the real number of pedestrian passages within an image to see if the chosen tracking approach has wielded decent results or not.

To count pedestrian in this study the number is incremented as soon as a pedestrian enters the image, and since the areas of interest were already calculated and correlated between frames, the count only goes up if there are areas of interest in the new frame that have no correlation with the areas of interest from the previous one. We do not decrement the count when a pedestrian leaves in order to check the reliability of our tracking algorithm.

Because in this study the only relevant information in terms of counting is to be used for error rates there is never a need to decrement the value and the number of pedestrians currently on a given image is always the number of areas of interest (**Figure 20**).

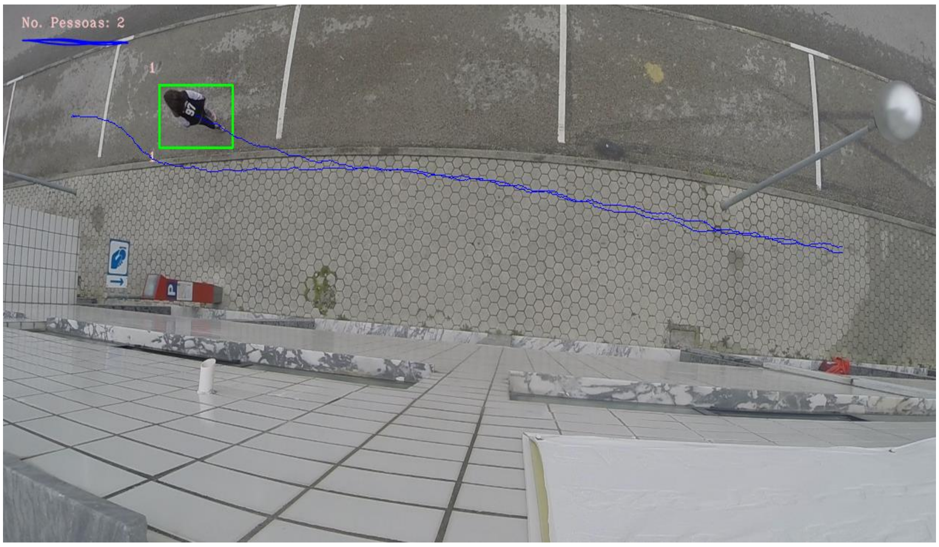


Figure 20 Count example.

3.5.2 Heat-Map or Point-by-point visualization

Another usual way to visually represent pedestrian movements is showing either the connection between the points where a given pedestrian passed or showing the heat map of the whole area.

The point-by-point visualization connects the centre of a pedestrian bounding box across multiple frames, as we can see in (**Figure 21**) where the paths travelled by pedestrians are represented as blue lines.

The heat map is more of a representation of the density of movement in a given area, not focusing on the individual pedestrian, but instead focusing on the number of times that a certain sub-area of the image had movement. To represent the heat map, we only need to use the result of the background subtraction, since the mask extracted will be enough, for us to change all the pixels where pedestrians are present and incrementing a certain value to those pixels in a mask that will be applied on top of the original image. Each colour in the heatmap image (**Figure 21**) does not represent a fixed amount of time or a number of passages, the colour representation is defined based on the number of frames where a certain area is exposed to movement compared to the total amount of frames that the captured video has. The colour map used was JET (**Figure 22**).

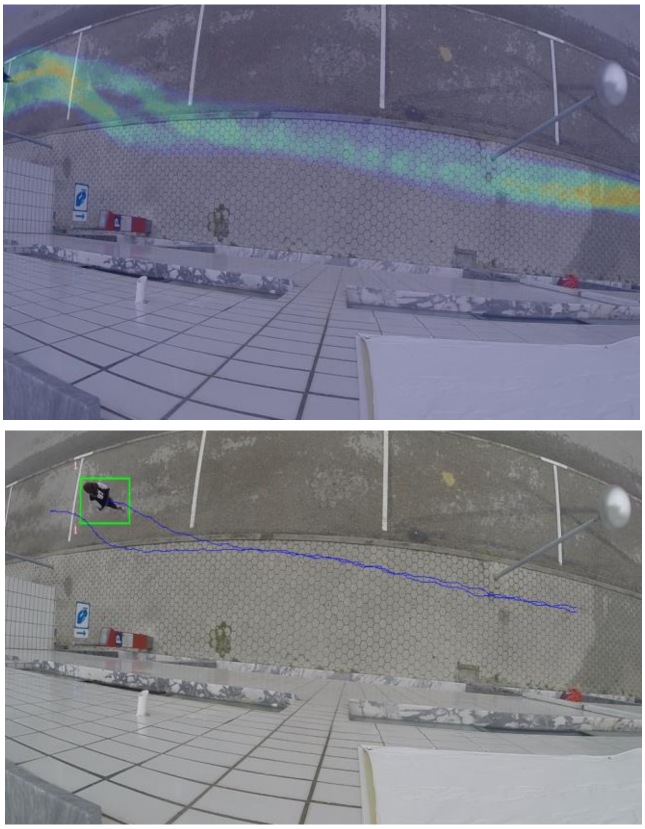


Figure 21 Heat map and Point passage representation.

Figure 22 OpenCV colormap JET, taken from https://docs.opencv.org/2.4/_images/colorscale_jet.jpg

3.6 Preliminary Single Image and Overlap Results

Some preliminary tests were done and here we will present the results that we obtained in the two different phases of our study.

In these tests our cameras were positioned 7 meters above ground and we had a scenario with some light variation, since the videos were captured on the street. Two individuals with different clothes were walking in and out of the two cameras areas of coverage.

In the first test we only considered a single image and we only tracked and counted the pedestrians that passed in that image.

The second test was the same as the first one, but this time we had a second camera that had an overlap area with the first camera. This main purpose of the test was to be able to relay between cameras the information that a pedestrian is appearing in both cameras at the same time, and that way only counts as one and his identification will not change, and he will be tracked across both cameras.

In these tests we were not dealing with re-identification yet, we were only testing the validity of our counting and tracking algorithms.

3.6.1 Single image

If we only consider a single image, with only single pedestrians (meaning no pedestrians are close together, avoiding the situation where a blob can represent multiple pedestrians), our implementation wields interesting results. Since, for now, we have a small number of people, the results were as expected, and we managed to be able to count every pedestrian that passed on the image without any error (**Figure 20**).

The tracking was also flawless since the only possible factor that could interfere with the results would be pedestrian overlapping (two or more people getting very close to each other) (**Figure 21**).

3.6.2 Two Images with Overlap Area

This time we were considering two images with a common area, also called overlap area, and pedestrians crossing from one image to another.

Since we want to track the same person through the two images, we want to make sure that when that if the pedestrian is within the overlap area (which means that he is appearing in both images at the same time), he will not add to the count since she already was counted when she entered the first image. Also, we want to keep the same identification number in both images.

The calibration was done by manually corresponding points between both images and calculating the corresponding homography. To test the quality of the homography there was an additional step, where we clicked on points on one image and the algorithm, based on the calculated homography, would show us the corresponding point on the other image (**Figure 23**).

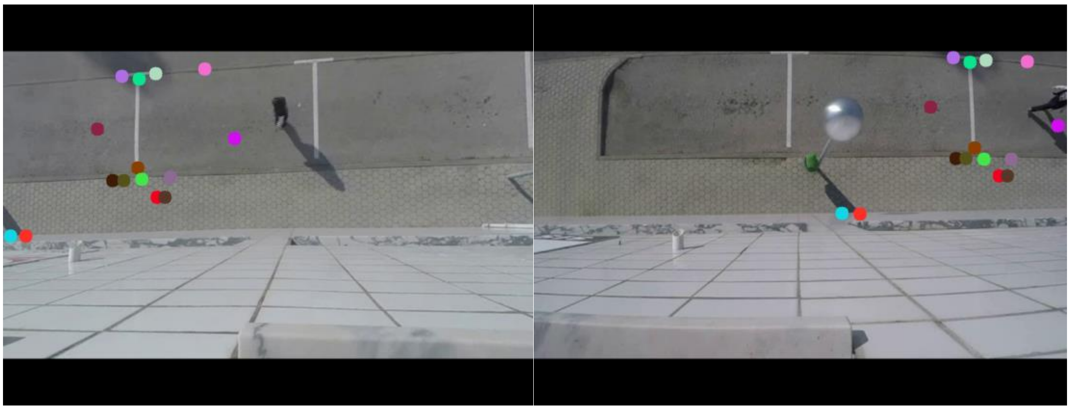
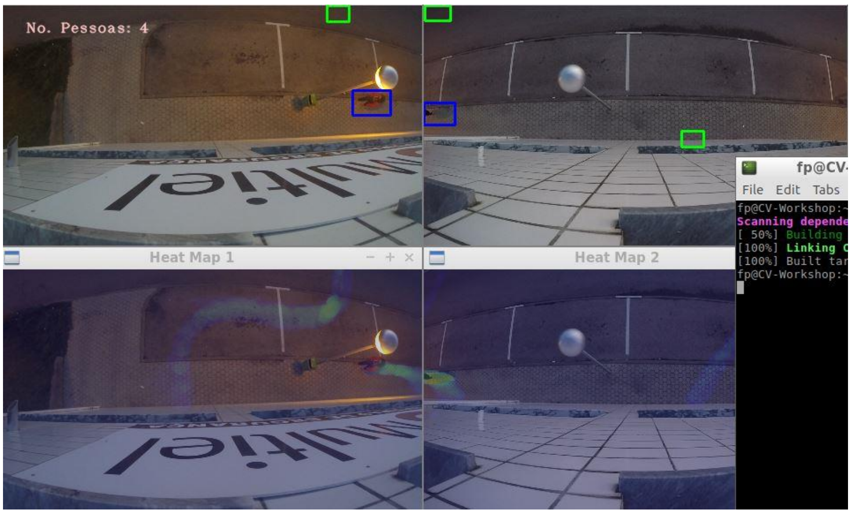


Figure 23 Point association between images with overlap.

To better understand the results presented (**Figure 24**), the green rectangles are representations of the zone where a pedestrian left the image. The blue rectangle represents that a pedestrian is within the overlap area and has been found in both images. If the pedestrian is within the overlap area and is not found in both images, that rectangle will also be green.

Once again, the counting and tracking algorithms perform well, but as stated before, we are dealing with a low number of pedestrians moving across images and the results might change if that number increases by a significant amount.



```
fp@CV-Workshop:~
Scanning depende
[ 50%] Building
[100%] Linking C
[100%] Built tar
fp@CV-Workshop:~
```

Figure 24 Counting, matching the same person in an overlap area, and tracking.

4 Pedestrian Re-Identification

The main novelty of our system compared to the existing systems focuses on pedestrian re-identification using only computer vision.

In this chapter we will describe the developed process of pedestrian re-identification as well as other alternatives. We start by looking at ways to extract relevant information that can be matched between cameras, and between frames in the same camera.

This chapter will end with a demonstration of the preliminary results obtained from our first tests.

4.1 Feature Extraction

Aside from the colour histograms there are also other features that can be useful to re-identify a person in a different image, for example, the direction a person is moving, the size of the person, and other features that can be extracted recurring to OpenCV's feature extractors.

Direction is probably the most useful additional feature to be extracted in our scenario, since depending on the direction a person is moving we can know if that person is showing the front part of the body or the back part of the body. This can lead to additional useful information since that person can be wearing an opened jacket in front causing the colour extracted from the back and from the front to be different.

Pedestrian size also seems to be an interesting feature for re-identification but is the least relevant one since various factors can influence the size of the area corresponding to a person, for example camera distortion and the pedestrian's position in the image, and since the captured images will be from a significant distance that information becomes even less relevant.

OpenCV's SIFT and SURF feature extractors can also be useful, but not by themselves. In order for this information to be useful it needs to be coupled with the direction feature, because we need to compare features between similar images in order to obtain the most accurate results. Comparing an image where a person is with her back turned to the camera with another image where the same person is with her front turned to the camera, will wield completely different results.

4.2 Colour Histograms and Colours Spaces

Colour histograms are a measure that allows us to evaluate the number of pixels in a given range of values in an image. Histograms are frequently used to extract information related to colour from an image and compare images. Since they represent the number of colour in an image, the results are highly influenced by the colour space that is being used in the image.

Histograms can be computed in different colour spaces: RGB (representing the Red, Green and Blue channels) is one of the most frequently used. HSV (Hue, Saturation and Value) is also often used since it can represent colour in a single channel (H). (**Figure 25**).

The main limitation of using RGB is the fact that if we want to extract information related to the most predominant colours found in the image, we need to extract the histogram of the combined channels, and that is very complicated, affecting performance, since we must go pixel by pixel in search of those values. With HSV we can get that information from a single channel.

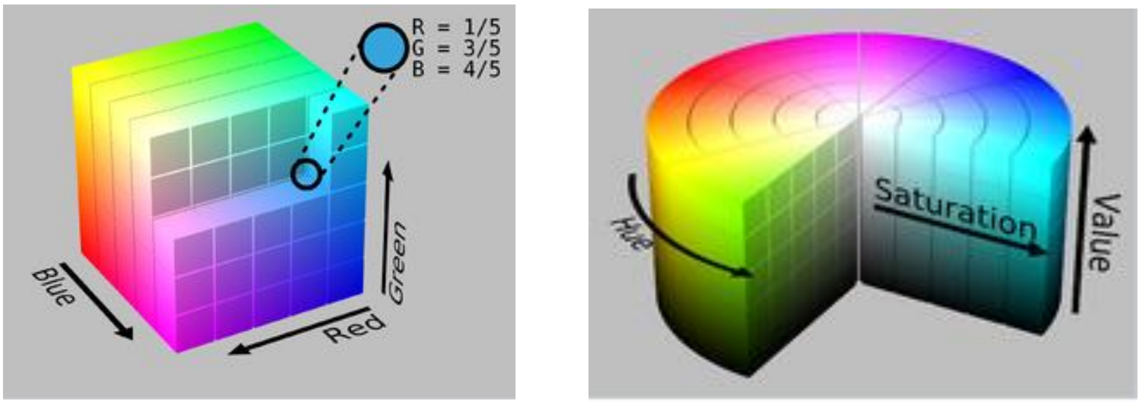


Figure 25 RGB [Wikipedia, 2018] and (b) HSV [Wikipedia, 2018], colour representations.

4.3 Selected Approach

For the reason given in the previous chapter, we opted for the colour histogram approach using HSV colour space, since it allows us to work only with the H channel in order to acquire all the information necessary for the re-identification approach. Using only the H channel also means that in situations where there are small light variations between images, the histogram comparison of both images H channel will still wield good results. This happens because the variation in light will be represented in the V channel, and the H channel will have similar values for the same colours.

It is important to remember that the colour histogram will be calculated not from the whole image but from a sub-image obtained by only checking the pixels that are within a bounding rectangle (that defines a pedestrian) which is also part of the result from the background subtraction. (**Figure 26**).

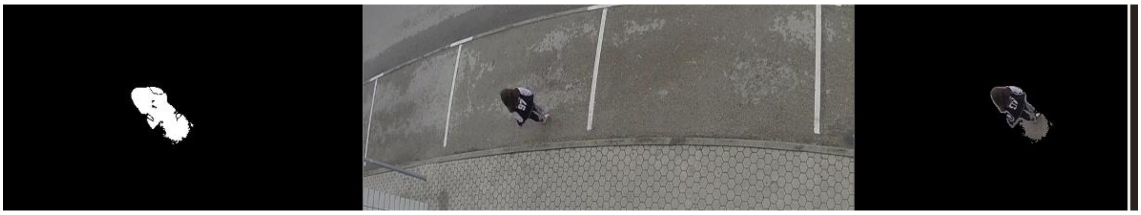


Figure 26 (Centre) Person on the extracted image, (Left) Person after being separated from the background, (Right) relevant area where the histogram will be calculated from.

4.4 Dominant Colours

Extracting information regarding the most common colours present in a given person can be very useful in the process of re-identifying the same person in a different image.

The process of calculating the most common colours is an easy one, since it is already done when we calculate the colour histogram of the extracted image. Since the colour histogram of the H channel is nothing more than the number of pixels where a given colour is present, we just have to identify which columns of the histogram have the highest values. Those columns are the dominant colours (**Figure 27**).

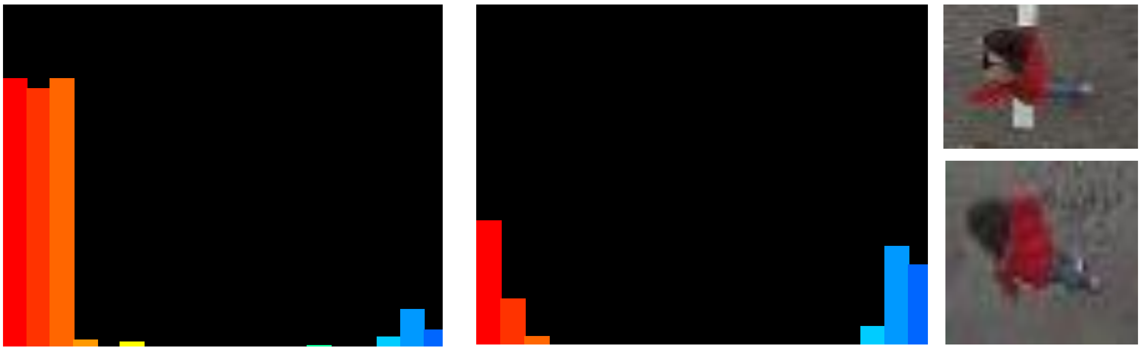


Figure 27 RGB coloured representations of H channel histograms of shown images.

4.5 Histogram comparison

To re-identify an object from an image on a different image, we need to extract the colour information of a certain area of the image (a pedestrian) and compare it with the histogram extracted in another slot from a different image, so that we can correlate people that left the camera view with people that are entering another camera view. If we find a person entering the new image that has the same patterns of colour as one that left the previous image we can assume that there is a certain probability that they are the same person.

OpenCV already provides four different histogram comparison algorithms [Ahmadi Tazehkandi et al., 2018]:

- Correlation;
- Intersection;
- Chi-Square;
- Bhattacharyya distance;

The comparison algorithm chosen by us was the Bhattacharyya distance [Bhattacharyya, 1943], which as the name implies, calculates a distance metric between the two histograms, and returns a result between 0 and 1, where 0 is a perfect match, and 1 is a complete mismatch. The more similar two histograms are, the closer the Bhattacharyya distance will be from 0. This metric is calculated by going to each bin value of both histograms and calculating the distance between both bins, and then averaging the distance between all the bins. From the four alternatives only two do bin-to-bin comparison, the intersection method and the Bhattacharyya distance, which is very important for the next step.

Bin-to-bin comparison allows us to make sure that two pedestrians are only considered similar if they have the same bins with the highest values, since if that is not the case, the calculated distance on those bins will strongly influence the final result of the Bhattacharyya distance.

4.6 Influence of pedestrian position in images

There are several studies already made related to histogram comparison and people re-identification using colour histograms, but as stated all the studies we found used side cameras to capture the images. On the other hand, our scenario is based on top view cameras.

This changes the approach necessary to study the problem, since the information obtained from a side camera is very different from the one obtained from a top view camera.

In a scenario where an image is captured from a side camera, we have visible information of every colour related to a person, since we can see the lower body colour, the upper body colour, and we can also try, and extract information related to the colours present in the head area. This makes it easy to find the dominant colours of a given person and makes it even easier to try and segment the person into three parts (head area, torso, and lower body), making all the colour extracted from the three parts even more useful as information, since it is probably a very accurate result.

On the other hand, a scenario where the camera is capturing the scene from a top view position, does not capture all that information, making segmentation of the person in different parts, to extract more relevant information, more difficult. To add to that problem, the cameras used have distortion which in turn makes the results different depending on the person's position within the image.

In particular we noticed that the position of the pedestrian in the image will influence its visible part (front or rear for example) (**Figure 28**) and this modification can imply significant variations in the histogram. To minimize these problems, we divided the image in three different areas (**Figure 29**), the centre where there is less distortion, and the right and left side, where there is more distortion but there is also more information to be found.

We calculate five different histograms for each pedestrian based on the area and direction in which the pedestrian is moving. For each side area we calculate a histogram from when the pedestrian is walking in the direction of the camera (showing information related to his front), and one from when he is moving away from the camera (showing information related to his back). In the centre area there is no need to obtain two histograms since we are mostly observing the pedestrian from above and the information is mostly the same and is independent from the direction the pedestrian is moving. To get more accurate histograms we calculate an average of the bin values until the pedestrian leaves the previous area.

For optimal results, this division cannot always be the same, since it should be adapted to the movement patterns.

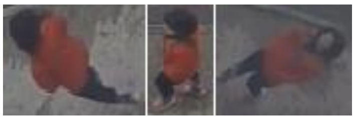


Figure 28 Different views according to pedestrian position in the image.

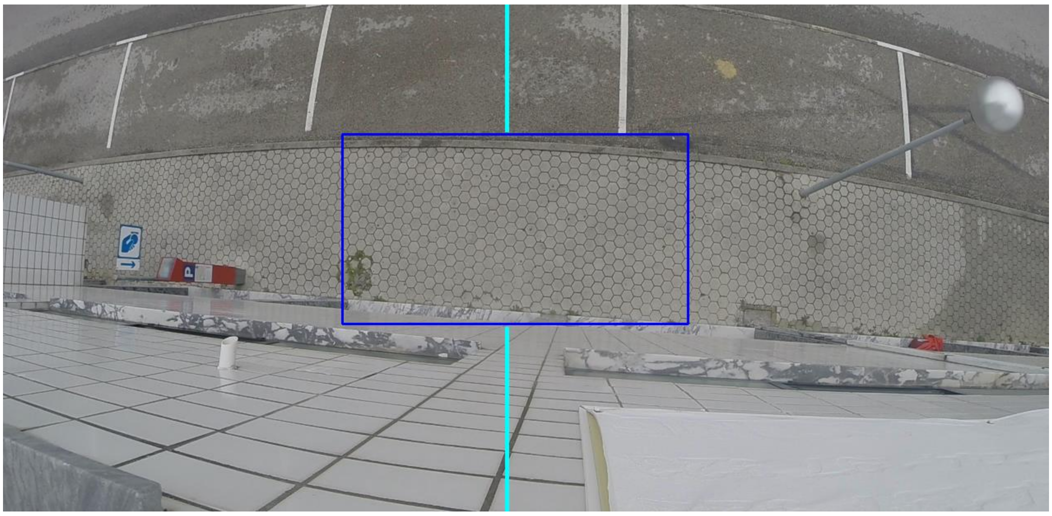


Figure 29 Area segmentation of the image. Left, Right, and centre area.

4.7 Re-identification with tracking

With the tracking algorithm working we can use it to reach a more concise decision regarding the re-identified person.

Since there may be cases where a pedestrian is wrongly detected as a different pedestrian we can use the developed tracking algorithm coupled with a defined frame threshold to be more certain of which person the pedestrian represents. In our case we defined a threshold of 50 frames in which after being detected for 50 frames as a certain person, we assume that the pedestrian is indeed that person.

To validate the results of the coupling we use the id given to each pedestrian by running our tracking algorithm and tagging the number of passages each pedestrian did in our images. In the end the ids of all the passages other than the first passage of each pedestrian will change, if the pedestrian is recognized, to the id of his first passage.

4.8 Preliminary Re-Identification Results

This is a continuation from the tests showed in the chapter where we present the preliminary results for the counting and the overlap between two images.

The test conditions were the same, the cameras were positioned 7 meters above ground and we had a scenario with some light variation, since the videos were captured on the street. Two individuals with different clothes were walking in and out of the two cameras areas of coverage.

In this test the cameras are not overlapping and because of that we must re-identify a pedestrian that leaves the first camera when he enters the second camera.

4.8.1 Two Images with no Overlap

For the final case study, we consider the case in which the two cameras are acquiring different sections with no overlap area. The final objective is to recognize a pedestrian that already appeared in one image in the other image, allowing to track his movements in the area under.

In our tests, the first image is used for information gathering, where we calculate the histogram of a tracked pedestrian across the 3 different areas that we previously defined for that image. (**Figure 29**)

Then, when a pedestrian enters the second image we try to match him with a list of pedestrians that already entered the first image and are not there anymore. (**Figure 30**)

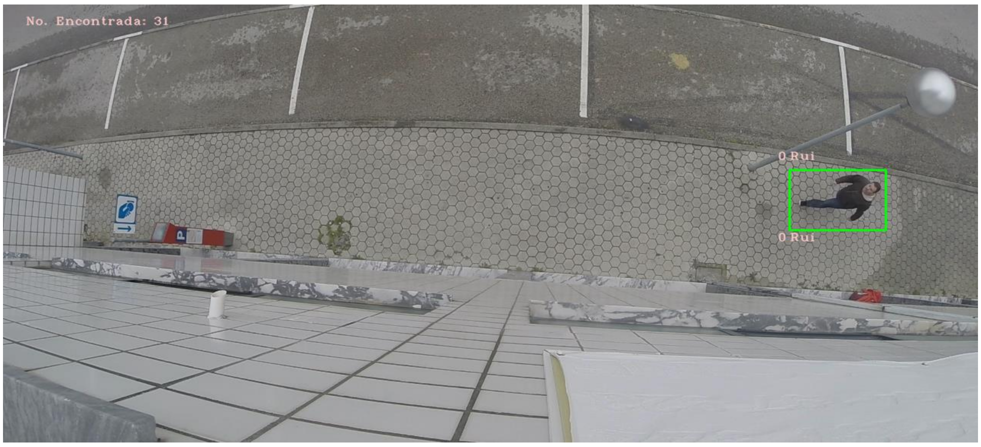
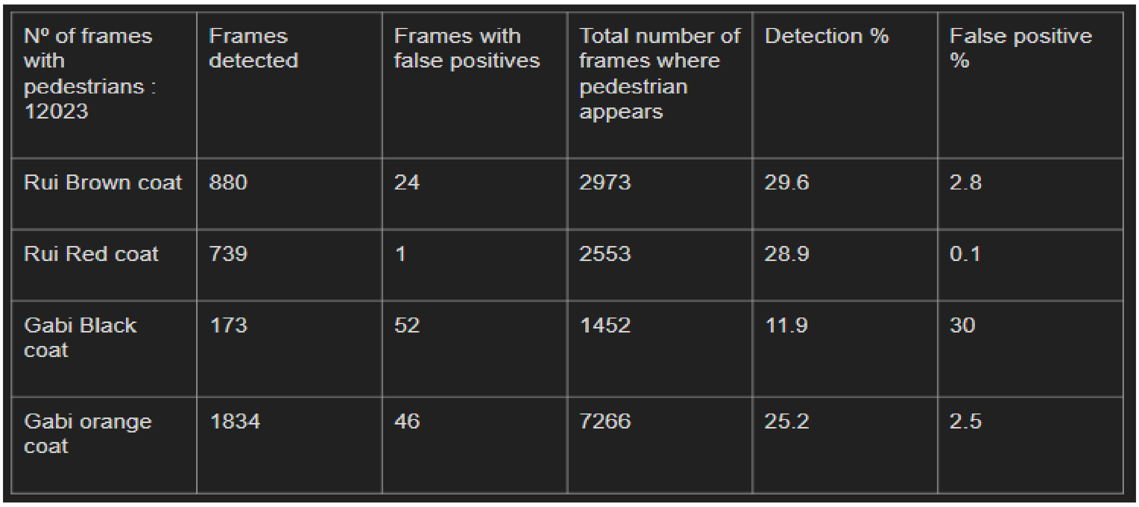


Figure 30 Person found on the second camera, after leaving the first camera.

We performed tests with two different resolutions to see how much the resolution of the captured image would influence the results, and as expected the lower resolution (**Table 1**) wields worst results compared to the higher resolution (**Table 2**).

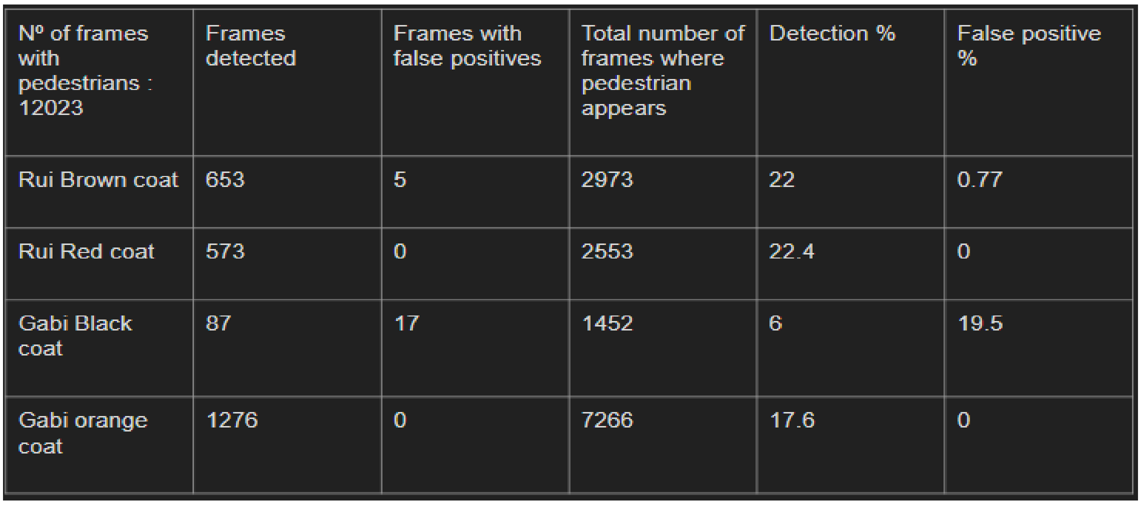
As we can see the results, in terms of a false detection, are much better with a higher resolution than with a lower one. The detection percentage is low in both cases (Detection %), but since we can couple the re-identification with the tracking algorithm already implemented, if we can detect a pedestrian for a certain amount of time and detect him correctly most of the time, we can establish a threshold, in our case 50 frames, and that way the detection percentage becomes less relevant for our solution, only influencing how fast a pedestrian can be re-identified.

With the tracking algorithm and the person re-identification, we can reach a 100% match for this test. Once again, we should be aware that this test was done with a reduced number of pedestrians (four in this case), and that results might be very different if we scale it.



Nº of frames with pedestrians : 12023	Frames detected	Frames with false positives	Total number of frames where pedestrian appears	Detection %	False positive %
Rui Brown coat	880	24	2973	29.6	2.8
Rui Red coat	739	1	2553	28.9	0.1
Gabi Black coat	173	52	1452	11.9	30
Gabi orange coat	1834	46	7266	25.2	2.5

Table 1 Pedestrian re-identification results using a 480x272 resolution.



Nº of frames with pedestrians : 12023	Frames detected	Frames with false positives	Total number of frames where pedestrian appears	Detection %	False positive %
Rui Brown coat	653	5	2973	22	0.77
Rui Red coat	573	0	2553	22.4	0
Gabi Black coat	87	17	1452	6	19.5
Gabi orange coat	1276	0	7266	17.6	0

Table 2 Pedestrian re-identification results using a 1280x720 resolution.

5 Results

This chapter presents some results of our algorithm using more realistic test scenarios in a public space with more users. We will look at our final test results with a single image, with two images with an overlapping area and, finally, with two images without an overlap area.

Once again, we did not allow for pedestrians to be too close to each other since that situation is more complex and we want to check the validity of our approach first.

Our test footage was taken in the library of the Universidade de Aveiro, since it was the only location available to us that had similar conditions to the ones present in a retail scenario. The cameras where positioned 7 meters above the ground and on different ends of the corridor.

There are 8 different pedestrians passing within the counting area, and 2 individuals that were almost static.

5.1 Single Camera

Each person has an id, and when a pedestrian enters the tracking (blue rectangle) he will add to the count on the top-left corner. The path taken by each pedestrian is shown as the blue lines within the tracking area.

Once again, the results related to counting and tracking in a single camera where 100% accurate with no errors both in the counting and tracking (**Figure 31**).

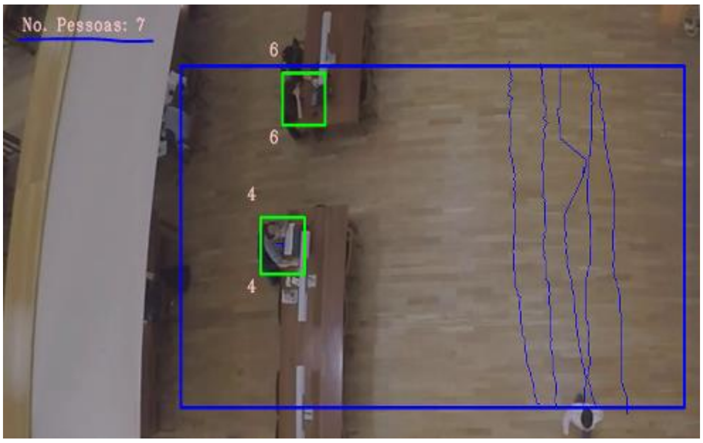


Figure 31 Tracking and Counting on a Single Image

5.2 Multiple Cameras with Overlap

In this scenario we had both cameras with an overlap of around 75%. Once again, the tracking and counting had no errors and all the correspondences were done correctly. The green rectangle signifies a person with no correspondence on the other image, and a blue rectangle (not to be confused with the blue rectangle on the previous image) is a person that is present in both images, counting only as one person.

In our case there are pedestrians sitting that might not have been detected in one of the images and that is why they are shown with a green rectangle (**Figure 32**).

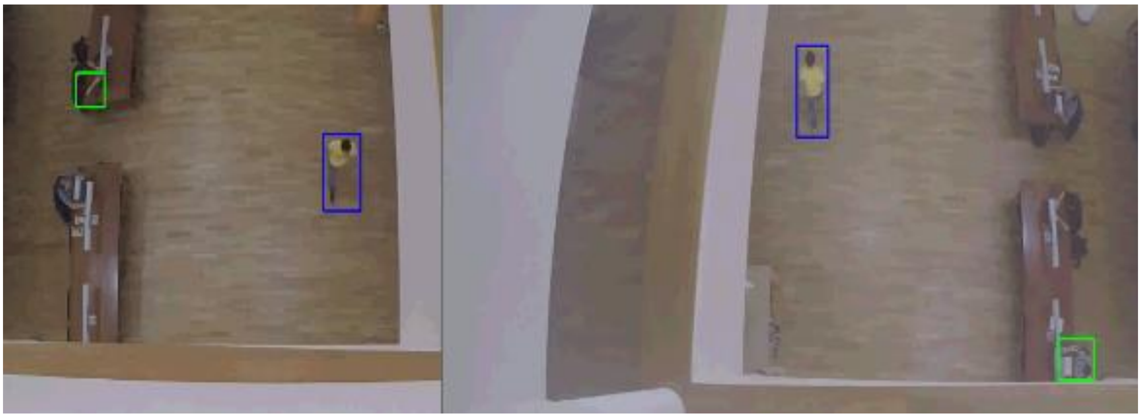


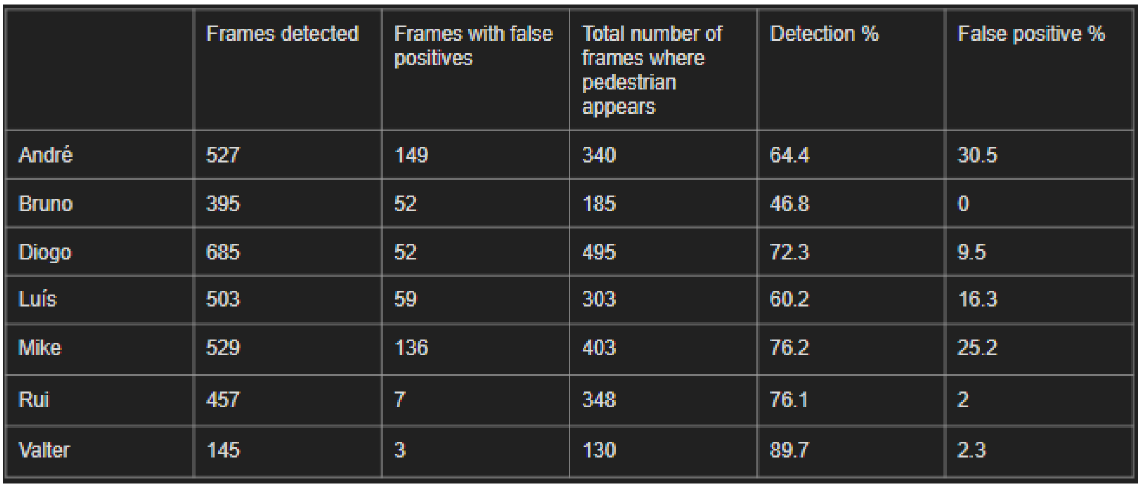
Figure 32 Images with overlap, with one person detected on both images

5.3 No overlap

Our final test was the same as the last preliminary test, where we have two different video feeds and we search for all the pedestrians that were already detected in one of the videos.

We check frame by frame if a certain pedestrian is present within the image, if he was re-identified as any pedestrian, and in case he was re-identified as any pedestrian, we check if he was re-identified as the right pedestrian.

The obtained results show that we can detect accurately most of the pedestrians that appear on the video, except for two cases (**Table 3**).



	Frames detected	Frames with false positives	Total number of frames where pedestrian appears	Detection %	False positive %
André	527	149	340	64.4	30.5
Bruno	395	52	185	46.8	0
Diogo	685	52	495	72.3	9.5
Luís	503	59	303	60.2	16.3
Mike	529	136	403	76.2	25.2
Rui	457	7	348	76.1	2
Valter	145	3	130	89.7	2.3

Table 3 Final Re-Identification results

Overall the errors shown happen because of pedestrians wearing the same dominant colour patterns, which can influence the results. Having said that, the colour pattern of two pedestrians has to be very similar for the algorithm to fail, since some of the pedestrians that show good results also had similar colour patterns between them. However, since the dominant colours were not all the same the algorithm was able to distinguish them well.

We can also obtain better results if we manage to couple the re-identification with the tracking and with the information regarding each camera's position. If we know that a given camera is always the first camera that a pedestrian passes when he enters the area, then we can automatically assume that if a pedestrian enters the area from a certain side, he is a new pedestrian. This way we do not compare him to the other pedestrians that were already in the image. Coupling this with the tracking algorithm (**Figure 33**) it is possible to decrease the error rate of every pedestrian with an error rate lower than 10%, down to 0% (**Table 4**). It is important to note that in (Table 4) the number of passages is the total number of times that pedestrian was seen in the image, the passages with false positive represent the number of passages another pedestrian was recognized as that pedestrian, and finally the number of passages with no error is self-explanatory. Also, when a pedestrian is re-identified we simply change the value of his id to the id of the pedestrian he was re-identified as, the colour of the rectangle is always green.

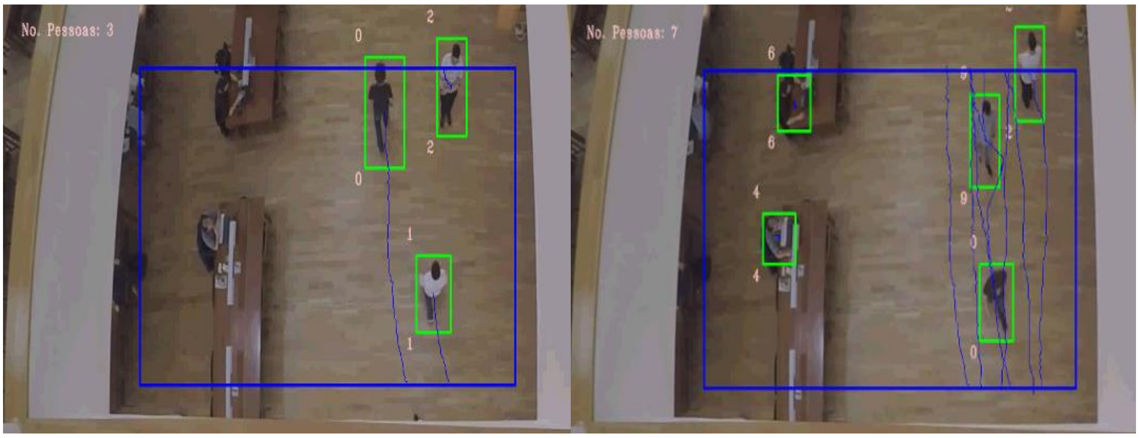
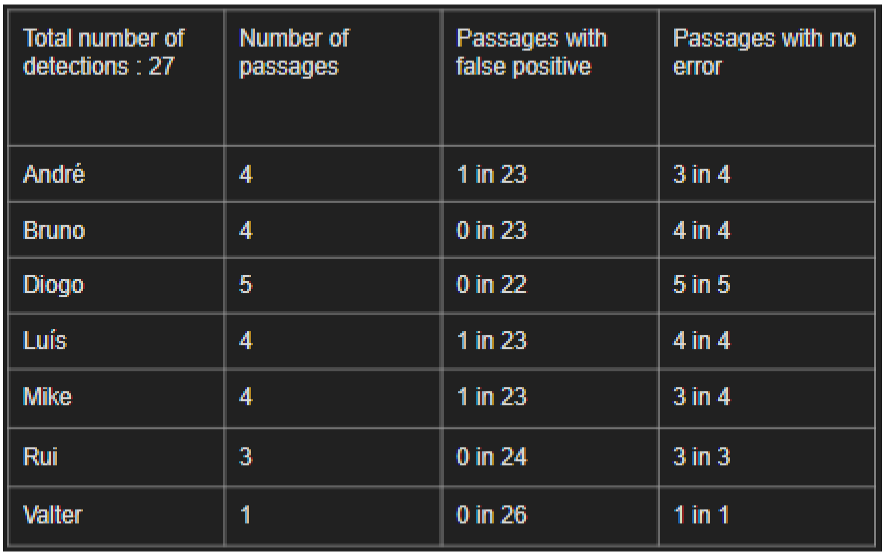


Figure 33 Id 0 and Id 2 re-identified across two different passages



Total number of detections : 27	Number of passages	Passages with false positive	Passages with no error
André	4	1 in 23	3 in 4
Bruno	4	0 in 23	4 in 4
Diogo	5	0 in 22	5 in 5
Luís	4	1 in 23	4 in 4
Mike	4	1 in 23	3 in 4
Rui	3	0 in 24	3 in 3
Valter	1	0 in 26	1 in 1

Table 4 Results of coupling the tracking with the re-identification.

5.4 Additional tests

With the acquired results we decided to test the algorithms with even more people and in a different scenario. Since in the beginning we set out a scenario where the cameras would be positioned between 5-7 meters above the ground, and since all our tests were done in scenarios where the cameras were always 7 meters above ground, this time we wanted to test the results with the cameras positioned 5 meters above ground. We had 11 pedestrians passing within the area covered by our cameras. Because the cameras are closer to the ground it is harder to get a scenario where there are no merged blobs, and because of that we discarded those blobs in this test.

We also changed the visual feedback when a pedestrian is re-identified because we felt that just changing the id of the re-identified pedestrian to the id of his first passage was not a good feedback. Now the id is changed, and the bounding box is drawn in a red colour (**Figure 34**).

It is also important to note that we limited the area where the re-identification is done because of light variation which creates shadows in certain parts of the screen, influencing the results. The area however is not as small as it seems having 7 meters in width and 3 meters in height.

The counting and tracking with a single image or with overlapping images continued to show good results, but the re-identification showed different results. Since we are capturing images closer than in the previous tests, we have more information and more detail captured for each person and because of that the results show no false positives and for pedestrians that were not caught within blob merging scenarios (blobs that represent multiple pedestrians) the re-identification was always done correctly (**Table 5**).

All the errors that occurred in the re-identification happened because the pedestrians got too close and the blobs merged. When the blobs separated again, there were not enough frames for the pedestrian to be re-identified again. These results were surprising since we did not expect to have no false positives, but it makes sense because, as mentioned before, we are capturing images closer to the ground giving us more detail that could not be captured 7 meters above ground. Of course, there are drawbacks to this since capturing closer to the ground also gives us more situations where blobs merge and less area covered.

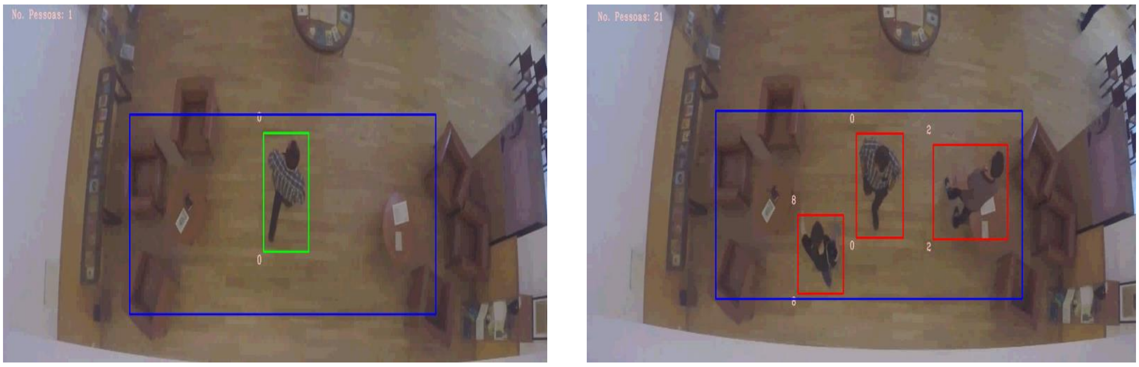
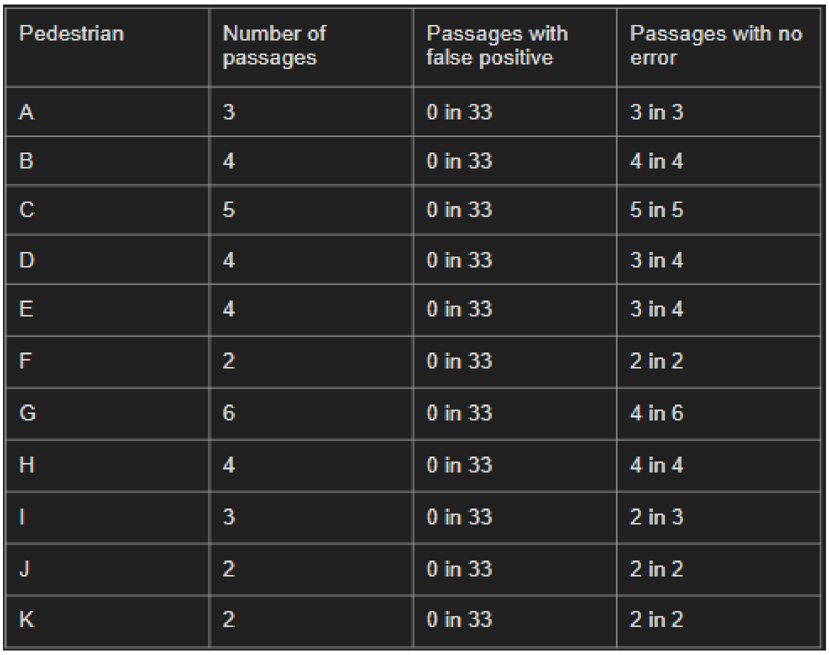


Figure 34 Pedestrian re-identification



Pedestrian	Number of passages	Passages with false positive	Passages with no error
A	3	0 in 33	3 in 3
B	4	0 in 33	4 in 4
C	5	0 in 33	5 in 5
D	4	0 in 33	3 in 4
E	4	0 in 33	3 in 4
F	2	0 in 33	2 in 2
G	6	0 in 33	4 in 6
H	4	0 in 33	4 in 4
I	3	0 in 33	2 in 3
J	2	0 in 33	2 in 2
K	2	0 in 33	2 in 2

Table 5 Results of re-identification coupled with tracking with 11 pedestrians.

6 Conclusions and future work

In this dissertation, we propose and describe a possible approach to pedestrian tracking across multiple cameras with or without overlap. We focus on the re-identification of pedestrians since that is the main purpose of this thesis.

The work was divided in four main stages: The study of pedestrian tracking systems in the market, choosing the methods to use on our application, capturing videos according to the specifications defined beforehand, and validating the results obtained by our approach.

We chose the background removal method to extract the pedestrians moving within the image, since that is the go-to algorithm for extracting moving objects from an image, especially if the cameras are in a fixed position. To re-identify pedestrians between different images we opted for the use of colour histograms since that is the most common method used to match pedestrians in different scenarios from ours, and we adapted that approach to our scenario.

In the end we managed to obtain very good results in tracking and counting within a single image and images with an overlap area, even though we did not deal with pedestrians passing too close together. For the re-identification we can conclude that for a reduced number of pedestrians the results are decent, but for a big number of pedestrians there are some adaptations that need to be done to the algorithm.

However, there are some steps that need to be worked on in the future.

The test footage that we worked with had a limited number of pedestrians and it was a controlled scenario, since we did not manage to find a good location to capture a high number of pedestrians without limitations. Because of that, our approach works for a reduced number of pedestrians, but we are not sure if it is scalable as it is.

Since the cameras used have a big distortion, we split the areas of interest within the image based on the observed movement. Those areas of interest influence the obtained results a lot and because of that an additional study on the optimal split areas of the image might be necessary to improve the results. Also, depending on the most common movement pattern those areas might not be always the same, making a tool to do the split automatically and according to the movement pattern might also be a very good way to improve the results.

Once again, because we did not deal with pedestrians passing very close together, opting to use either the re-identification method proposed in this thesis to fix that situation or using feature extraction, is a very important next step for scalability.

Also, it is important to note that using only the H channel can cause problems in situations where the colour is either White or Black, since the H channel doesn't have a representation for those cases. An alternative is either to use another colour space, or we can use a 2D histogram with information regarding the H and V channels.

Another possible improvement is to extract patterns or texture from the pedestrians and use it as extra information for the re-identification process.

Because there are situations where the area covered may have big lighting variations which can cause shadows to be visible in the area where there is more light and no visible shadows where there is less light, there is a need to deal with this situation in order to cover areas with that variation in light. A possible solution is to deal with the two areas, created by the light variation, separately. Since the background removal algorithm already allows for shadow removal, it might be interesting to check if the obtained results are the similar if we apply the background removal differently in each area.

7 References

Stahlschmidt, Carsten & Gavriilidis, Alexandros & Velten, Joerg & Kummert, Anton. (2013). People Detection and Tracking from a Top-View Position Using a Time-of-Flight Camera. 368. 213-223. 10.1007/978-3-642-38559-9_19.

Kaewtrakulpong, P., & Bowden, R. (2001). An Improved Adaptive Background Mixture Model for Real- time Tracking with Shadow Detection. *Advanced Video Based Surveillance Systems*, 1–5. https://doi.org/10.1.1.12.3705

Zivkovic, Z. (2004). Improved adaptive Gaussian mixture model for background subtraction. *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, *2*(2), 28–31. https://doi.org/10.1109/ICPR.2004.1333992

Zivkovic, Z., & Van Der Heijden, F. (2006). Efficient adaptive density estimation per image pixel for the task of background subtraction. *Pattern Recognition Letters*, *27*(7), 773–780. https://doi.org/10.1016/j.patrec.2005.11.005

Godbehere, A. B., & Goldberg, K. (2014). Algorithms for visual tracking of visitors under variable-lighting conditions for a responsive audio art installation. In *Controls and Art: Inquiries at the Intersection of the Subjective and the Objective* (pp. 181–204). Springer International Publishing. https://doi.org/10.1007/978-3-319-03904-6_8

Lucas, B. D., & Kanade, T. (1981). An Iterative Image Registration Technique with an Application to Stereo Vision. *Imaging*, *130*(x), 674–679. https://doi.org/10.1109/HPDC.2004.1323531

Brown, M., & Lowe, D. G. (2007). Automatic panoramic image stitching using invariant features. In *International Journal of Computer Vision* (Vol. 74, pp. 59–73). https://doi.org/10.1007/s11263-006-0002-3

Kaehler, A. and Bradski, G. (2016). Learning OpenCV 3. O'Reilly Media.

Ahmadi Tazehkandi, A., Godoy, V. and Buhr, K. (2018). Computer Vision with OpenCV 3 and Qt5. Birmingham: Packt Publishing.

Bhattacharyya, A. (1943). On A Measure of Divergence Between Two Statistical Populations Defined by their Probability Distributions. *Bulletin of the Calcutta Methematical Society*, *35*(1), 99–109. https://doi.org/10.1038/157869b0

Kwangchol Jang, Sokmin Han and Insong Kim on their paper "Person Re-identification Based on Color Histogram and Spatial Configuration of Dominant Color Regions", https://arxiv.org/ftp/arxiv/papers/1411/1411.3410.pdf .

Nakatani, Ryota & Kouno, Daichi & Shimada, Kazutaka & Endo, Tsutomu. (2012). A Person Identification Method Using a Top-view Head Image from an Overhead Camera. Journal of Advanced Computational Intelligence and Intelligent Informatics. 16.

8 Links

Lebied, M. (2017). Top 10 Analytics & Business Intelligence Trends for 2018. [online] BI Blog | Data Visualization & Analytics Blog | datapine. Available at: https://www.datapine.com/blog/business-intelligence-trends/ [Accessed 14 Dec. 2017].

Business Reporter. (2016). Movement intelligence: Retail's transformational opportunity - Business Reporter. [online] Available at: https://business-reporter.co.uk/2016/05/16/movement-intelligence-retails-transformational-opportunity/#gsc.tab=0 [Accessed 15 Aug. 2017].

Dwoskin, E. and Timberg, C. (2017). Google's new program to track shoppers sparks a federal privacy complaint. [online] Washington Post. Available at: https://www.washingtonpost.com/news/the-switch/wp/2017/07/30/googles-new-program-to-track-shoppers-sparks-a-federal-privacy-complaint/?noredirect=on&utm_term=.275d54233809 [Accessed 15 Aug. 2017].

McWilliams, A. (2013). How a Depth Sensor Works - in 5 Minutes | Andrew McWilliams. [online] Jahya.net. Available at: https://jahya.net/blog/how-depth-sensor-works-in-5-minutes/ [Accessed 8 Sep. 2017].

GSMArena.com. (2017). The TrueDepth camera of the iPhone X works like the Xbox Kinect. [online] Available at: https://www.gsmarena.com/truedepth_camera_of_iphone_x_works_like_the_kinect-news-27242.php [Accessed 29 Oct. 2017].

Multipix Imaging. (2017). Basler Time-of-Flight Camera - Multipix Imaging. [online] Available at: https://multipix.com/product/basler-time-of-flight-camera/ [Accessed 26 Oct. 2017].

Multipix Imaging. (2017). Basler Time-of-Flight Camera. [online] Available at: https://www.baslerweb.com/en/products/cameras/3d-cameras/time-of-flight-camera/ [Accessed 28 Oct. 2017].

Docs.opencv.org. (2015). OpenCV: Background Subtraction. [online] Available at: https://docs.opencv.org/3.1.0/db/d5c/tutorial_py_bg_subtraction.html [Accessed 3 Jan. 2018].

Forums.ni.com. (2016). Ball Tracking. [online] Available at: http://forums.ni.com/legacyfs/online/183400_test1.png [Accessed 13 Feb. 2018].

Tlantic.com. (2017). Tlantic InStore Tracking - - Tlantic - software for a fitter retail. [online] Available at: http://www.tlantic.com/pt/software/software-mobilidade/tlantic-instore-tracking/ [Accessed 18 May 2018].

Amazon.com. (2016). Amazon.com: : Amazon Go. [online] Available at: https://www.amazon.com/b?ie=UTF8&node=16008589011 [Accessed 9 May 2018].

YouTube. (2016). Introducing Amazon Go and the world's most advanced shopping technology. [online] Available at: https://www.youtube.com/watch?v=NrmMk1Myrxc [Accessed 9 May 2018].

Xovis.com. (2013). Person Tracking Technology | XOVIS. [online] Available at: https://www.xovis.com/en/xovis/ [Accessed 29 Jul. 2017].

ShopperTrak. (2010). ShopperTrak: Retail intelligence, Contagem de pessoas. [online] Available at: https://pt.shoppertrak.com/ [Accessed 29 Jul. 2017].

YouTube. (2010). ShopperTrak Introduction. [online] Available at: https://www.youtube.com/watch?v=ShmE7Dx4cxU [Accessed 29 Jul. 2017].

Artsensor, S. (2013). MOBOTIX - Soluções de vídeovigilância de alta resolução Artsensor. [online] Slideshare.net. Available at: https://www.slideshare.net/ARTSENSOR/mobotix-artsensor [Accessed 30 Jul. 2017].

Youtube.com. (2013). Mobotix MxAnalytics. [online] Available at: https://www.youtube.com/watch?v=Kwl6LVCwKfY [Accessed 30 Jul. 2017].

Image-sensors-world.blogspot.com. (2016). CCD vs CMOS Infographic. [online] Available at: http://image-sensors-world.blogspot.com/2016/05/ccd-vs-cmos-infographic.html [Accessed 12 Feb. 2018].

PetaPixel. (2013). A Mathematical Look at Focal Length and Crop Factor. [online] Available at: https://petapixel.com/2013/06/15/a-mathematical-look-at-focal-length-and-crop-factor/ [Accessed 7 Oct. 2017].

GoPro. (2017). HERO3 Black Edition Field-of-View FOV Information. [online] Available at: https://pt.gopro.com/help/articles/question_answer/HERO3-Black-Edition-Field-of-View-FOV-Information [Accessed 8 Feb. 2018].

Cameras, T. (2017). GoPro Hero 3+ Plus Silver. [online] Ted's Cameras. Available at: https://www.teds.com.au/gopro-hero-3-plus-silver [Accessed 8 Feb. 2018].

ebayimg (2017). GoPro Hero 4. [online] Available at: http://i.ebayimg.com/images/g/HIsAAOSwSclXLLII/s-l1600.jpg [Accessed 8 Jan. 2018].

YouTube. (2015). OpenCV 3.0 Tracking API Results. [online] Available at: https://www.youtube.com/watch?v=pj-QuE6pdEQ [Accessed 11 Sep. 2017].

Wikipedia (2018). RGB color space. [online] Available at: https://en.wikipedia.org/wiki/RGB_color_space [Accessed 11 Feb. 2018].

Wikipedia (2018). HSL and HSV. [online] Available at: https://en.wikipedia.org/wiki/HSL_and_HSV [Accessed 11 Feb. 2018].