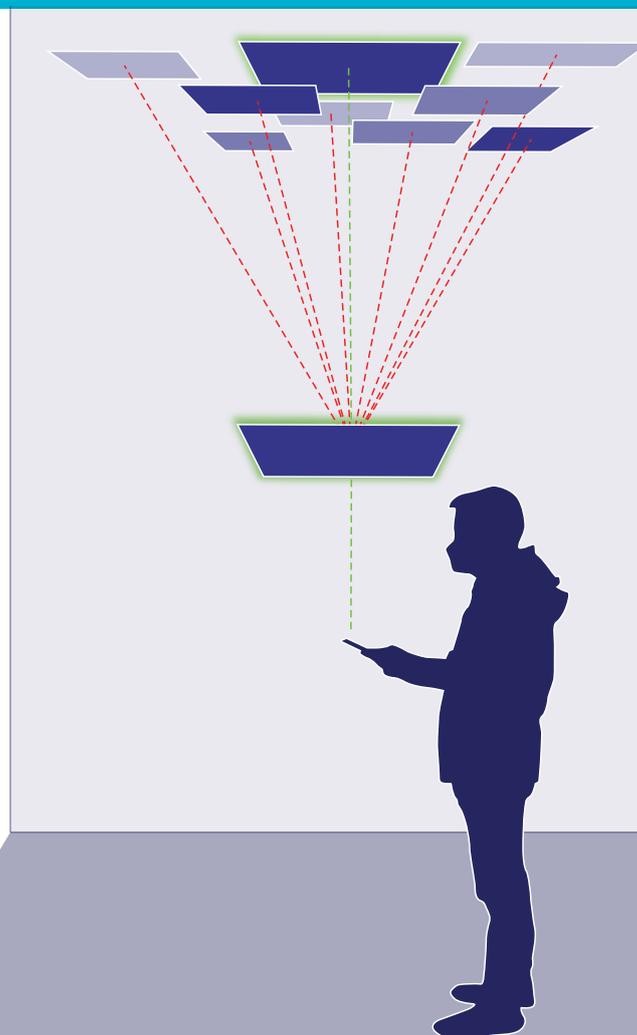


MSc thesis in Geomatics

# Indoor localisation and location tracking in semi-public buildings based on LiDAR point clouds and images of the ceilings

Ioannis Dardavesis

2022





MSc thesis in Geomatics

**Indoor localisation and location tracking in  
semi-public buildings based on LiDAR point  
clouds and images of the ceilings**

Ioannis Dardavesis

June 2022

A thesis submitted to the Delft University of Technology in  
partial fulfillment of the requirements for the degree of Master of  
Science in Geomatics

Ioannis Dardavesis: *Indoor localisation and location tracking in semi-public buildings based on LiDAR point clouds and images of the ceilings* (2022)

© This work is licensed under a Creative Commons Attribution 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.

The work in this thesis was carried out in the:



GIS Technology group  
Faculty of Architecture and the Built Environment  
Delft University of Technology

Supervisors: Ir. E. Verbree  
Dr. A. Rafiee  
Co-reader: Dr. L. Díaz-Vilariño

# Abstract

Nowadays, the evolution of localisation and navigation technologies is vast, aiding towards facilitating users' guidance in various environments. Outdoor positioning can be easily achieved, with the widely used Global Navigation Satellite Systems (GNSS), which comprise a universal standard for positioning and are included in every person's mobile device. However, due to the presence of high buildings in dense urban environments and bad reception in indoor environments, the performance of GNSS is significantly degraded. Therefore, alternative ways of positioning and localisation respectively, need to be explored. In indoor environments, unlike outdoors, there is no universal standard, as the different indoor localisation techniques, that are currently implemented have their own bottlenecks. The most widely used Wi-Fi fingerprinting, requires a constantly up-to-date radio map of the signals from the Wi-Fi access points, whose creation is also a heavy and time-consuming technique. Additionally, other techniques require an installation of costly sensors or either equipment.

Therefore, this thesis investigates the possibility of the ceilings in public or semi-public buildings, being used for indoor localisation, by using features that are included in a simple mobile device. The research additionally involves location tracking of different users, in order to discover different movement patterns in an indoor facility. Indoor localisation is achieved based on the comparison of user and reference data, that can be both point clouds and images, using the Light detection and ranging (LiDAR) of an iPad 12 pro and camera sensors of an Android device. The point cloud-based localisation is implemented based on different combinations of global and local registration techniques, while the image-based approach involves different feature detection, description and matching techniques. Using a web-application to visualise the indoor localisation results, an indoor model and a network graph of the Faculty of Architecture and the Built Environment, location tracking of different users is implemented and visualised in a heat-map. Additionally, a dashboard is created that can be used by a facility manager to translate the user paths to valuable information and reveal different movement patterns in an indoor facility.

The followed methodology showed promising results, concerning the reliability of ceilings for real-time indoor localisation, based on LiDAR and camera sensors, that are incorporated in up-to-date mobile devices. The robustness of Colored Iterative Closest Point (ICP) algorithm for indoor localisation based on point clouds was revealed, both in terms of time efficiency and quality, while the combination of Speeded-Up Robust Features (SURF) feature detector and Scale Invariant Feature Transform (SIFT) descriptor provides the optimal indoor localisation results with image data. The proposed pipeline revealed encouraging results for use in emergency situations, based on static data acquisition of a user, while it is also suitable for dynamic applications, in case a sensor is mounted on an automated device for indoor mapping operations.

**Keywords:** Indoor localisation, Location tracking, LiDAR, iPad 12 pro, Point cloud registration, Feature matching, Ceilings



# Executive Summary

## Introduction

Nowadays, the evolution of localisation and navigation technologies is vast, aiding towards facilitating users' guidance in various environments. Outdoor positioning can be easily achieved, with the widely used GNSS, which are included in every person's mobile device. However, due to the presence of high buildings in dense urban environments and bad reception in indoor environments, the performance of GNSS is significantly degraded. Therefore, alternative ways of positioning and localisation respectively, need to be explored.

In indoor environments, there is lower landmark density and an absence of outstanding elements that can frequently result to easier loss of orientation, compared to outdoors [Michon and Denis, 2001]. As Wadden and Scheff mentioned, people spend around 80 % of their time indoors, thus localisation comprises an important problem, in public buildings, such as airports or train stations, that usually consist of chaotic spaces, leading to a higher chance that an individual will become disoriented. Therefore, the necessity of a dynamic indoor localisation system is apparent, so that a person can navigate in an indoor facility, especially if that person is exploring it for the first time. In that scenario, a location provider tool with adequate precision could be a significant aid. This tool could be applied in various indoor spaces, such as museums and art galleries [Gupta et al., 2016].

Indoor localisation could also be applied in emergency situations in complex indoor spaces. Persons in need could access the name of their current location, based on an indoor localisation application and transmit this information to the first-aid responders. The latter need guidance, related to the location of the person in need, as well as a way to reach that location [Yang and Worboys, 2011]. Additionally, other applications of indoor localisation include the use of mobile autonomous units, in an effort to establish an indoor intelligent environment. Mobile service robots can be exploited for assisted living, setting up a smart living environment for elderly people and perform transportation and human interaction tasks.

Currently, various sensors, such as Wi-Fi fingerprinting and Bluetooth, are widely used for precise localisation in indoor environments. These technologies work by performing triangulation and trilateration of different users and become the basis of different indoor localisation applications. Other applications use optic sensors to achieve object recognition (Google Lens). Recently some new applications have emerged, that use Augmented Reality (AR) combined with the Simultaneous localisation and mapping (SLAM) algorithm, which works by scanning an

indoor environment in order to find a person's location in this space [Oostwegel, 2020].

Compared to outdoor environments where GNSS comprises a universal standard for positioning, the same cannot be claimed for indoor environments [Lymberopoulos et al., 2015]. The most widely used technology, Wi-Fi fingerprinting, is based on the comparison of Received signal strength (RSS) values with a reference radio map that translates signal values to positions [Pérez-Navarro et al., 2019]. Creating this radiomap is a heavy and time-consuming task. Many fingerprints are required and a single change in the Wi-Fi infrastructure, requires the map to be created again. Consequently, an up-to-date radiomap of the signals in an indoor facility is mandatory for this technique to function. Moreover, the availability of Wi-Fi signals in indoor environments might be irregular, which could be an outcome of poor Wireless local area network (WLAN) planning in the facility. Similarly, alternatives that are based on wireless technologies, such as Bluetooth, that requires the installation of costly Bluetooth hotspots, are affected by the aforementioned problems.

Other indoor localisation solutions combine SLAM and AR technologies by taking into account different sensors, such as RGB and depth cameras. Simple AR applications can be ran in a basic smartphone, however additional devices are usually required, to offer a solution with a deeper understanding of the indoor environment.

The problems that emerge in the existing techniques and the lack of a universal solution [Pérez-Navarro et al., 2019], provide a space for innovation by utilising other techniques, that take into advantage camera and LiDAR sensors. While cameras sensors exist in every mobile device, the use of LiDAR sensors is exponentially increased, as they were also recently included in the latest releases of Apple's iPhone and iPad devices, showing that they will be a major part of the mobile devices that will follow in the upcoming years. Therefore, the challenge of achieving indoor localisation in different environments, is to find a technique that does not depend on costly and hard to access indoor sensor networks, but to use features that are accessible to everyone in their mobile device [Willems, 2017].

This research will explore the possibility of the ceilings in public or semi-public buildings, being used for indoor localisation purposes, by providing an accessible solution that makes use of features that are available in a simple mobile device. The focus includes indoor localisation, as well as near real-time location tracking of different users for the purpose of discovering different movement patterns in an indoor facility.

## Methodology and implementation

The main steps of the pipeline will be first briefly explained. First, the ceiling data acquisition, was achieved based on three different techniques. Single images of the tested rooms were acquired with camera sensors from different mobile devices, as well as overlapping images of the same sites, to achieve 3D reconstruction of these scenes. Additionally, point clouds were acquired from LiDAR sensors of an iPad, that include this sensor. For both the point clouds and the images, indoor localisation was achieved by comparing user data and reference data, that were uploaded in a database. Regarding single images, their features were matched based on different matching techniques. Multiple overlapping images of a ceiling were first reconstructed in three dimensions, both from user and database side. As a result, point cloud registration techniques were used to compare the two types of point clouds. Furthermore, directly acquired point clouds, were first pre-processed and then co-registered to achieve indoor localisation. Indoor localisation results derived from point cloud comparison were stored in a database and were visualised in a web application. The indoor model of the tested facility and its network graph were combined with the localisation results to provide information on users' current and previous locations. This way, the used paths were revealed and consequently the movement patterns in an indoor facility. The visualisation of this location tracking operation has the form of a heat-map, including user paths during different times of a day. Additionally, a dashboard included statistics about the path usage was created. This pipeline was applied to some rooms of the Faculty of Architecture and the Built Environment of TU Delft, that are shown in the next figure.

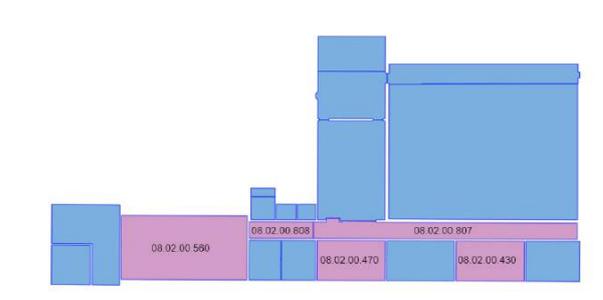


Figure 1.: Selected rooms from the Faculty of Architecture and the Built environment

Regarding point cloud acquisition, two types of point clouds were acquired, from LiDAR sensors. There are point clouds that act as reference and were stored in a database, as well as point clouds that are acquired by a user. The latter will be compared to these reference point clouds, so that indoor localisation is achieved, based on the best match. The acquisition involves the use of two

applications, SiteScape and Pix4D Catch and the LiDAR sensor of an iPad 12 pro. The point cloud acquisition was implemented in two ways: while a person is walking into a room, giving a dynamic perspective to the acquisition and also while staying still, so that it is investigated if the final product of the thesis can be used during emergency situations, in cases where an individual might be unable to move.

Single images of the tested rooms were acquired from camera sensors of a Xiaomi Redmi Note 9s Android phone. As in the case of point clouds, some images were used as reference to represent the room's ceiling in two dimensions. Images of a ceiling acquired by a user were then compared to the reference images of the rooms in order to reveal the user's location based on the optimal match. Furthermore, overlapping images of ceilings were acquired with the same sensors and a minimum overlap, for the reconstruction of the 3D scene. Additionally, concerning the overlapping images, different combinations of feature detectors, descriptors and feature matching techniques were tested, in an effort to discover their optimal combination, for indoor localisation based on images of the ceilings.

Pre-processing of the point clouds included voxel down-sampling in order to reduce processing time by manipulating a point cloud of smaller size [Miknis et al., 2016]. However, this operation has to be implemented carefully and until a certain threshold, because further down-sampling might result to important loss of information. Furthermore, when acquiring ceiling data, it is possible that the point cloud includes adjacent wall parts, that need to be excluded from the upcoming operations. These parts can be considered as outliers [Han et al., 2017] for the purposes of this research. To achieve their removal, a smaller part of the acquired point cloud was used, in order to discard the wall parts that might exist in the corners of the point clouds. Additionally, some outliers were located and removed based on the number of their neighbours, to further improve the point cloud's quality and reduce processing time. Last but not least, plane segmentation based on the Random sample consensus (RANSAC) algorithm was performed, in order to differentiate the flat surface of the ceiling with its protruding objects, such as lamps and other installations, which comprise the characteristic details of each room's ceiling.

After acquiring and pre-processing reference and user point clouds, the next step was to create an algorithm that would aid towards comparing them. The main idea behind this is, that each point cloud taken by a user, would be compared with all the point clouds in the database and the best match will reveal the room where the user is located. This procedure works as follows for both types of point clouds. The comparison first included a global registration, so that the user and the reference point clouds obtain an initial alignment and afterwards a local registration algorithm to refine the point cloud registration.

First, the normal vectors of all the points were computed. Furthermore, points with a unique and descriptive neighbourhood were detected. The detection and description



Figure 2.: Pre-processing steps

of these unique points for each point cloud was implemented based on Fast Point Feature Histogram (FPFH) feature calculation. Two different global registration techniques were implemented and compared.

The first technique includes the aforementioned steps and then RANSAC, in order to select some random points from the reference point cloud and then find the corresponding points in the user point cloud, using a nearest neighbor query in the 33-dimensional FPFH feature space [Li et al., 2021]. Aside from the distance of the corresponding points in the compared point clouds, the similarity between two edges between the compared point clouds and the vertex normal affinity of the correspondences are also checked. In case the points satisfy the selected thresholds, the transformation of the user point clouds towards the reference point clouds is implemented.

The second technique was implemented, based on the fast global registration proposed by Zhou et al. This method follows the steps of the global registration described above, however it does not use RANSAC to choose point correspondences between the two point clouds, but finds the nearest neighbour of every point in the user point cloud among the reference point cloud, based on distance analysis in feature space.

Based on the results of the global registration, an attempt of improving the quality and time efficiency of the algorithm includes different variations of the ICP algorithm. The further minimisation of the point cloud differences was performed by keeping one point cloud fixed, while the other is transformed towards it. Specifically, each point of the user point cloud was matched to the closest point of each reference point cloud. Then, rotation and translation were estimated and this process is iterated until the results converge [Li et al., 2021]. The user point cloud was compared to all the reference point clouds, based on the fitness and the Root Mean Squared Error (RMSE) value, which will result in the indoor localisation. Different variations of ICP were implemented and compared and more specifically Generalised, Point-to-Point, Point-to-Plane and Colored ICP. These steps were implemented both to directly acquired point clouds, as well as point clouds that were reconstructed from overlapping images of a ceiling.

For each of the selected rooms, one image of a ceiling was acquired and acted as reference. For testing purposes, different user images were additionally acquired from different viewpoints and were compared with the reference images. This comparison included the use of different feature descriptors and detectors, such as Oriented FAST and Rotated BRIEF (ORB), SIFT, and also two different feature matching techniques, brute-force and Fast Library for Approximate Nearest Neighbors (FLANN). The number of matches between the user and the reference images was used to reveal the location of the user.

An important step towards 3D reconstruction is the feature matching between the overlapping images. Therefore, some subsets of images were selected, in order to evaluate different combination of feature detectors, descriptors and matching techniques from acquired images of ceilings. These images were acquired based on different overlaps, in an attempt to examine how the percentage of coverage affects the time complexity of the 3D reconstruction and the quality of the results. The quality of the results was examined based on the number of matches between the overlapping images, as well as a ground truth which was manually set for several images, so that the results are protected against false matches. Graphical representations that show the true to false matches were created to enhance the results. Following these steps, the overlapping images sets were used for 3D reconstructions of the ceilings.

The setup of the whole system was organised in an online database, part of the ArcGIS Online Server. This database includes the indoor model of the case study and a network graph, that connects all the rooms of the tested area to each other. Except for the geometry of the rooms in the indoor model, each of them includes one pre-processed point cloud and an image that acts as reference for the point cloud registration and feature matching operations respectively. Moreover, this indoor model serves as an embedded map in web-application that was created, allowing the users to have a visual insight of their location.

The indoor localisation results were visualised in a web application that has the form of a minimum viable product. The app works by requesting the reference point clouds from the database, so that they can be compared

based on the discussed algorithms to the user data in near-real time. Users are able to post their data in the application and after a few seconds the room they are located in is revealed. Additionally, the app includes the indoor model of the case study, so that aside from the name of the room, the app also highlights the polygon that represents the room in the indoor model of the indoor facility and zooms in it.

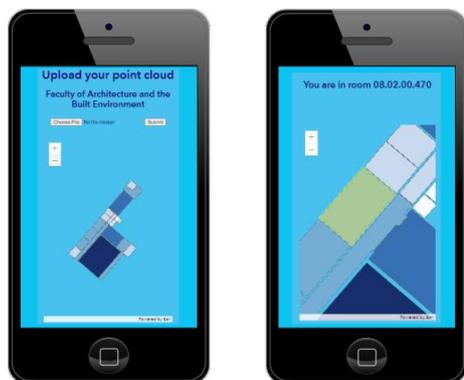


Figure 3.: Web-app interface

Each time the web application is used, the users' current and previous locations are stored in the ArcGIS Online Server, under an encrypted id. When users move between different rooms, it means that they used a certain path to achieve that. Based on the network graph of the indoor space that reveals all the connections between adjacent rooms, the current and previous locations of the users were translated to a line in the network graph, representing a specific route. The availability of this information is near-real time as the results appear in the online server after a few seconds. Based on the unique id of each user, a heat map that is based on the network graph was used to visualise the used routes.

Additionally, this information was used to reveal different movement patterns, during different times of a day. The visualisation is accomplished in the form of a heat-map, where based on the usage of each path, different colors and width were applied to the corresponding line of the network graph. Consequently, this information can reveal how much a path is used during a daily, weekly or even monthly time span. Acquiring this knowledge is valuable, especially during the COVID-19 era, because it can be exploited by a building manager, who can achieve the optimal distribution of people in an indoor facility [Spinoza Andreo et al., 2021].

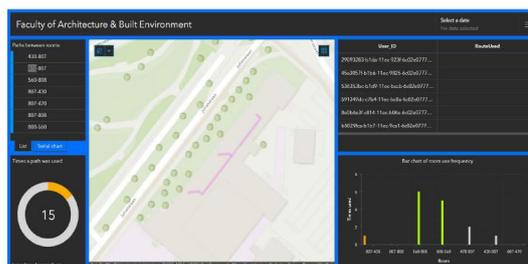
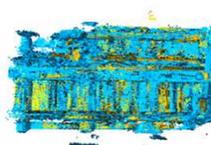


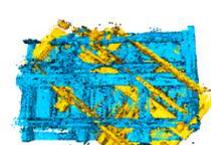
Figure 4.: ArcGIS Dashboard

## Results

The first results emerge from point clouds acquired with the SiteScape app, with a 10 cm distance between each point.



(a) Colored ICP  
(fitness=0.974,  
RMSE=0.08)



(b) Point-to-Point ICP  
(fitness=0.645, RMSE=0.2)

The results for room 08.02.00.560 are promising, as in most cases all the point cloud registration methods match the tested room to its reference equivalent. However in Figure 5b where fast global registration was combined with the Generalised ICP, room 08.02.00.430 resulted in higher fitness than the correct room 08.02.00.560. However, the correct room had the lower RMSE value. The most accurate results are achieved when Colored ICP was involved, producing accurate results when it was combined with global registration algorithms, as Figure 5a and indicates. It has to be noted, that the number of fitness is not important by itself, but it has to be higher compared to the reference point clouds of the remaining rooms.

| Point clouds | RANSAC Global Registration |                |             |                 |
|--------------|----------------------------|----------------|-------------|-----------------|
|              | Point-to-Point             | Point-to-Plane | Colored ICP | Generalised ICP |
| Dynamic      | 4/5                        | 4/5            | 5/5         | 4/5             |
| Static       | 4/5                        | 4/5            | 5/5         | 4/5             |
| Point clouds | Fast Global Registration   |                |             |                 |
|              | Point-to-Point             | Point-to-Plane | Colored ICP | Generalised ICP |
| Dynamic      | 5/5                        | 5/5            | 5/5         | 5/5             |
| Static       | 3/5                        | 4/5            | 5/5         | 3/5             |

Table 1.: Number of correct matches per point cloud registration algorithm

Table 1 shows the number of correct matches for each combination of global and local registration algorithms that were applied. The testing includes ten point clouds per method and specifically five for the ceilings that a user

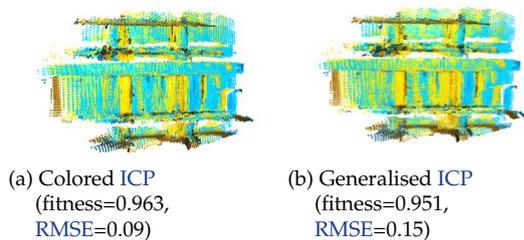
acquired while walking, and five more while the user remained static. In total, both **RANSAC** and fast global registration algorithms have similar results, when combined with different local refinement algorithms. **RANSAC** is a non-deterministic algorithm, however the high number of iterations that was selected, increases the probability that the result is reasonable. On the contrary, the fast registration algorithm, which does not have the same non-deterministic nature, performs slightly faster and produces results with a similar consistency, compared to **RANSAC**, each time it is executed.

The results are better, when users are walking inside a room during data acquisition, in contrast to when they remain static. This is a reasonable outcome, as while a user is walking, the entire ceiling of a room can be captured. On the contrary, while users remain static, they can only capture a specific part of a room’s ceiling, in case the room is considerably large, since the range of the **LiDAR** sensor is approximately five meters. Therefore, in cases where users are unable to move, there are higher chances that the localisation is correct when they capture a part of a ceiling that has characteristic details.

The wrong point cloud matches for some registration techniques, appear between rooms 08.02.00.430 and 08.02.00.470. This confusion arises from the fact that these rooms have almost identical size in squared meters and similar characteristic details in their ceilings, as they are both lecture rooms. Additionally, the second wrong set is mostly between rooms 08.02.00.808 and 08.02.00.807. This happens, due to the fact that they are both corridors and room 08.02.00.808 is significantly smaller than room 08.02.00.807. Thus, it is possible that this room is wrongly matched as a part of 807. Some rooms, such as 08.02.00.807 which is a long corridor, has a significantly different shape than the common rectangular rooms, hence the possibility that the localisation is wrong is significantly reduced.

Concerning, wall parts that were acquired along with ceilings, small areas did not affect the results, as some minor wall parts remained in the tested point clouds even after the pre-processing operations. However, in cases where a significant part of a wall is captured, the plane segmentation could be implemented in a wrong way, as the main plane that is computed, might be the wall instead of the ceiling’s upper flat part.

Additionally, point clouds were also acquired with the Pix4D Catch app, with a distance of 30 cm between each point.



Similarly to the previous test case, the results concerning the room 08.02.00.560 are promising, as the combination of global and local registration methods produces the correct result in most cases. The combination of **Colored ICP** and **RANSAC** provides the higher fitness value, however testing for room 08.02.00.560 provided similarly good results in every combination.

| Point clouds | RANSAC Global Registration |                |             |                 |
|--------------|----------------------------|----------------|-------------|-----------------|
|              | Point-to-Point             | Point-to-Plane | Colored ICP | Generalised ICP |
| User         | 5/5                        | 4/5            | 5/5         | 4/5             |
| Point clouds | Fast Global Registration   |                |             |                 |
|              | Point-to-Point             | Point-to-Plane | Colored ICP | Generalised ICP |
| User         | 4/5                        | 3/5            | 4/5         | 3/5             |

Table 2.: Number of correct matches per point cloud registration algorithm

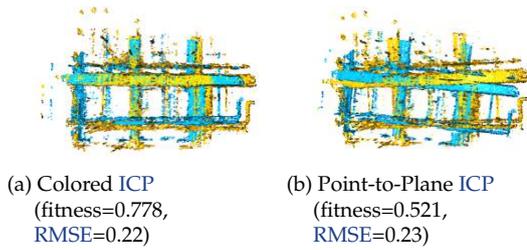
Table 2 shows that the quality of the indoor localisation is slightly worse compared to the results produced by SiteScape. The result is reasonable, due to the lower density of point clouds that was chosen for the acquisition. However, regarding the **Colored ICP** its results are at a similar level as before, showing the importance of adding color information that the other algorithms do not include.

Concerning global registration techniques, **RANSAC** shows better results compared to fast, with 18/20 correct room matches, while fast at the same time results into 14/20 correct indoor localisation results. As it was also mentioned in Section 5.1.1, **RANSAC** is a non-deterministic algorithm, however the high number of iterations that were set in the implementation, increases the probability that the results are more reasonable. The small size of the original point clouds, significantly increases the time efficiency of the algorithm, with a minor time difference between the different methods. The worst results are presented for Point-to-Plane and Generalised **ICP** when they are combined with global registration algorithms, with 7/10 correct indoor localisation results.

The wrong localisation results concern room 08.02.00.430, which is in some cases wrongly mismatched to 08.02.00.470. Their identical size and details, as they are both lecture rooms with similar characteristic details is the reason behind this wrong match. Additionally, while combining fast global registration with Point-to-Plane and Generalised **ICP** algorithms, an other wrong result was observed between rooms 08.02.00.808 and 08.02.00.807. Specifically, room 08.02.00.808 was wrongly localised as 08.02.00.807. This misinterpretation arises due to their difference in size, as the latter is significantly bigger, therefore, it is possible that room 08.02.00.808 is incorrectly considered as a part of 08.02.00.807.

In this part the same testing will be applied for point clouds that were reconstructed from overlapping image sets that were acquired with Pix4d Catch.

In this case, the parts of the ceilings that are behind installations could not be acquired, hence not modelled, as the acquisition involves images. However, that does not comprise a problem in most cases. An important observation



is that the point cloud reconstruction from image sets, may result into point clouds that have a different scale compared to the point clouds that were directly acquired. This is an outcome of the 3D reconstruction process, as only the intrinsic of the cameras are known and the position of the 3D points is computed based on the projections from the 2D space, so the true scale of the scene cannot be accurately recovered. Therefore, the 3D reconstruction of a ceiling is unique up to a scaling factor. For this reason, point clouds that were reconstructed from images have to be used as reference, for the matching results to be accurate. Concerning the results of room 08.02.00.560, the best results in terms of fitness can be once more noticed, where Colored ICP was involved, however Point-to-Point and Generalised ICP combinations also provided high fitness values. A bad matching is observed while combining RANSAC to Point-to-Point ICP with 52% fitness value (Figure B.47a), however this number was higher than the ones of the respective rooms, resulting into a correct localisation.

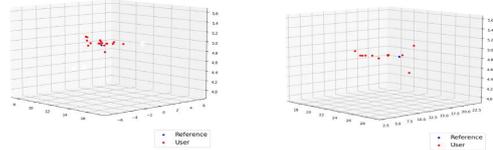
| Point clouds | RANSAC Global Registration |                |             |                 |
|--------------|----------------------------|----------------|-------------|-----------------|
|              | Point-to-Point             | Point-to-Plane | Colored ICP | Generalised ICP |
| User         | 4/5                        | 3/5            | 5/5         | 4/5             |
| Point clouds | Fast Global Registration   |                |             |                 |
|              | Point-to-Point             | Point-to-Plane | Colored ICP | Generalised ICP |
| User         | 4/5                        | 3/5            | 5/5         | 5/5             |

Table 3.: Number of correct matches per point cloud registration algorithm

Both global registration algorithms have similar quality of results and specifically the combination of RANSAC with local refinement algorithms, gives 16/20 correct localisation results, while fast with the same combinations issues indoor localisation correctly, 17/20 times. Concerning local registration algorithms, Colored ICP appears to have the maximum success rate with 10/10 correct localisation results, while the worst results involve Point-to-Plane ICP with 6/10 correct results. Once more, the addition of color information to the existing geometry can significantly improve the point cloud registration results, ensuring a high success indoor localisation rate.

In contrast to the previous datasets, the wrong matches do not include the same rooms as before, however the results in overall are similar. There are different combinations of rooms that were mismatched. This is a result of the scaling factors, during the reconstruction operation that model the third dimension with a scale ambiguity. Thus, rooms that appear to have a different size in reality, might be modelled similarly in terms of size, a fact that could

result into wrong localisation, when the protruding components of the ceilings are not enough to differentiate the rooms between each other.



(a) Scatter plot with centers of reference and user point clouds after point cloud matching in room 08.02.00.808 (b) Scatter plot with centers of reference and user point clouds after point cloud matching in room 08.02.00.807

Figure 5.15a and Figure 5.15b show the centers of the respective reference point cloud with blue color, as well as the centers of different user point clouds after the implementation of the point cloud registration algorithms and specifically RANSAC based global registration and Colored ICP local refinement. The results concerning room 08.02.00.808 reveal good accuracy, as most of the centers of the user point clouds are a few centimeters away from the center of the reference point cloud, while at the same time the precision is adequate, as most of the centers of the user point clouds are close to each other. On the contrary, the same results for room 08.02.00.807 are worse concerning accuracy and also precision, since the centers of the user point clouds are further away from the center of the reference point cloud and at the same time far from each other. This has to do with the size and length of the room 08.02.00.807, that is a corridor with similar and lengthy protruding installations on the ceilings, therefore it is possible that the user point clouds are matched to the reference point cloud on a different part of those installations further away from the center of the point cloud. However, in both cases there is good accuracy and precision regarding the height dimension, which shows that the flat part of the ceilings of the user and reference point clouds is in most cases correctly matched.

| 5 MP Camera |         |           |
|-------------|---------|-----------|
|             | ORB-ORB | SIFT-SIFT |
| Brute-Force | 4/5     | 5/5       |
| FLANN       | 4/5     | 5/5       |
| 8 MP Camera |         |           |
| Brute-Force | 4/5     | 5/5       |
| FLANN       | 4/5     | 5/5       |

Table 4.: Number of correct matches per feature detection, description and matching techniques

The results are based on images that were taken from two different cameras with 5 and 8 MP resolution respectively. Both cameras perform similarly resulting into 18/20 correct room matches. Additionally, the two feature matching techniques have similar efficiency when they are combined with the two different detectors and descriptors, while brute-force performs slightly faster than

**FLANN.** However, the latter can be more efficient than brute-force, when large datasets are involved. **FLANN** results into a higher number of matches between the user image and the reference image of the correct room in most cases. The same can be mentioned about **SIFT**, which results into more matches between the images compared to **ORB**, however the indoor localisation is calculated with worse time efficiency. In terms of quality, the suitability of **SIFT**, lies in the fact that it is scale and rotation invariant, whereas **ORB** is only rotation invariant and robust to noise. As a result, in case **SIFT** is used, the height and angle of the device do not affect the result. The time efficiency of **SIFT**, could be improved, by implementing the **SURF** detector and descriptor. The ratio test that was applied in each experiment was strict, in order to avoid false correspondences, due to the common installations between the different rooms. The most clear results were noticed concerning a test image of room 08.02.00.470, where approximately 400 matches were observed between the user and reference image, a number which is significantly higher compared to the other reference images. This is an outcome of the similarity of the user and reference images, as they were acquired from a similar angle and cover approximately the same part of the ceiling. In other cases where the viewpoints of the user and reference images were different, the indoor localisation results were correct, as the user image had the most matches with its corresponding reference image, however the number of matches was significantly lower, between 50 and 100.

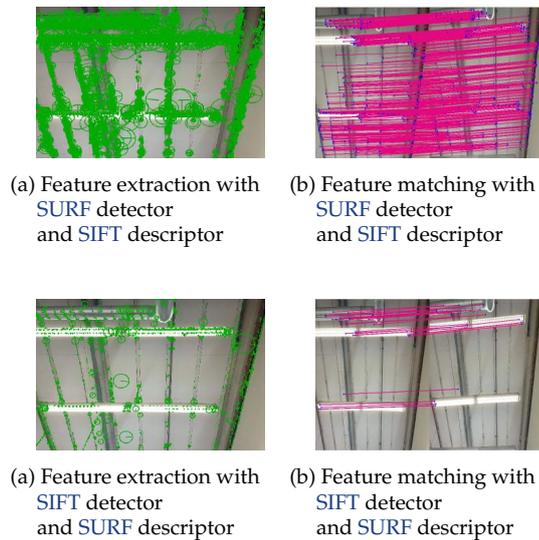
The wrong localisation results, were related to room 08.02.00.807, that cannot be entirely captured from a single image, due to its length. Therefore, in terms of size, it appears to be similar to different rooms of the case study. However, this result can be partially solved, in case the data acquisition is performed, by holding the sensor almost perpendicular to the ceiling, so that a bigger part of the ceiling is captured.

In this testing, there are no differences between the two different cameras regarding the quality of the results. However, certain illumination changes that create blurry areas, may significantly affect the intensity of each pixel of the tested images. In this situation, a high resolution camera could better capture the reality and avoid these blurry parts in the images. However, a drawback of using cameras with high resolution, is that they tend to produce bigger image files, that are not suitable for real-time applications, due to the necessity of a time efficient solution.

Some wrong matches are highly affected by the ceiling lights that are on, during most part of the day in the Faculty of Architecture and the Built environment. These lights tend to create blurry areas around them, tampering with the real intensity values of the pixels. Additionally, the intensity values of these areas might appear similar to the windows, resulting into wrong matches between the windows and the lights, when two images are compared. Hence, during the acquisition, windows should be avoided as much as possible, due to their reflective ability.

Overall, indoor localisation based on the comparison of

the features of an image seems really promising, however additional testing regarding lighting conditions and viewpoints, has to be implemented to produce safe conclusions about this method. Testing in a larger database is also a challenge, as well as the implementation of the **SURF** detector and descriptor, to check the suitability of this indoor localisation method based on images, for real-time applications.



Additionally, some subsets of overlapping images were chosen in order to test additional combinations of feature detectors, descriptors and feature matching techniques. The testing that was performed in an open-source software called Photomatch, showed that the combination of **SURF** as a detector and descriptor detects the maximum number of key-points (5000) with both brute force and **FLANN** matching techniques. The opposite is observed for the combination of **SIFT** and Binary Robust Independent Elementary Features (**BRIEF**) are combined with almost 3500 thousand key-points. The latter happens due to the simplicity of the **BRIEF** descriptor which targets in fast description from simple intensity difference tests. Regarding the percentage of key-points that are used for matching, **SURF** detector with **SIFT** descriptor and **FLANN** matching take into advantage approximately 13% of the detected points, while the combination of **SIFT** detector, **SURF** descriptor and **FLANN** uses less than 1% of the detected key-points for feature matching. This is a result of the size of the vectors of **SIFT** and **SURF** descriptors, which have a size of 128 and 64 elements, showing that **SIFT** entails more details concerning the description of the sub-region of the tested key-points. In most cases, **FLANN** uses a higher percentage of key-points for matching, compared to brute-force except when the **SIFT** detector and **SURF** descriptor are combined, however the difference is minor.

The location tracking results are based on the different indoor locations of different users in different times of a day.

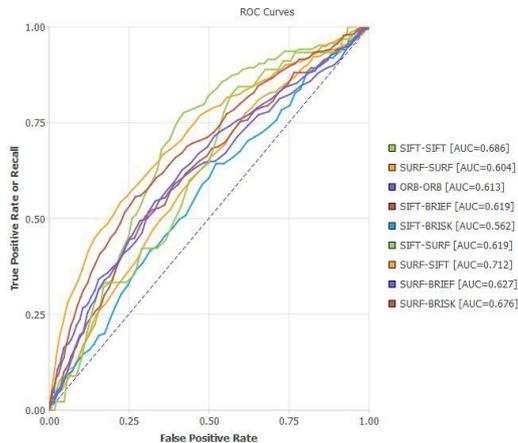


Figure 11.: Receiver Operating Characteristic (ROC) curves between 2 images with Brute-force matching

Therefore, the quality of the followed paths is a direct outcome of the indoor localisation quality. The results are available in the ArcGIS online Server and can be seen in near real-time in a map, that is updated every 30 seconds. To test the accuracy of the location tracking algorithm, a ground truth was set, based on the path that the user originally followed and was compared to the path, as it is visualised in the final product. This is shown in Figure 5.23.

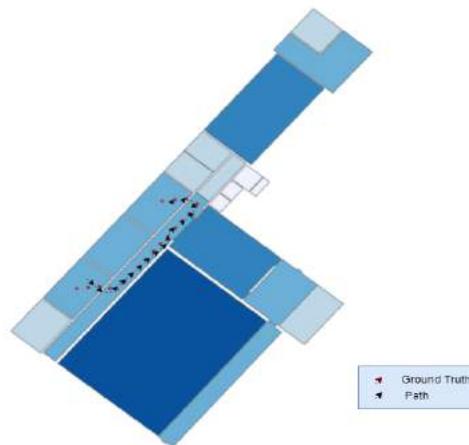


Figure 12.: Ground truth and visualised paths between rooms 08.02.00.430, 08.02.00.807 and 08.02.00.470

Figure 5.23 shows the path of a user that moved between rooms 08.02.00.430, 08.02.00.807 and 08.02.00.470. The indoor localisation was performed correctly for these three rooms, therefore the ground truth is similar to the path as it is visualised in ArcGIS Pro. Some differences exist due to the indoor network that is used to visualise the paths, as the center of each room is the representative node and the

fact that the rooms are connected to each other with lines, therefore small deviations when the user is not moving completely straight cannot be detected.

## Conclusion and discussion

This research aimed to investigate the reliability of ceilings as an alternative way of achieving indoor localisation and in extent location tracking of users, based on LiDAR and camera sensors, which are incorporated in up-to-date mobile devices, in an effort to substitute the varied used localisation methods that mostly involve Wi-Fi fingerprinting and Bluetooth sensors. In that manner, indoor localisation becomes possible for a variety of users, without the need of additional equipment. The only requirement of this pipeline, is the existence of point clouds of ceilings that will act as reference for every room of the indoor facility. The indoor localisation from the different testing that was implemented showed promising results, both in terms of quality as well as time efficiency, as the scope of the thesis was to be able to perform real-time localisation of large indoor environments, focusing on ceilings with characteristic details. Based on the results, a point cloud acquisition of a few seconds is enough to indicate the room that users are in, especially when the whole ceiling can be captured. In case a ceiling is partly acquired, the indoor localisation result depends on the uniqueness of the captured part. Additionally, the point cloud acquisition of ceilings led to promising localisation results while implemented dynamically, during continuous acquisition between different rooms. The range of the current LiDAR sensors is approximately 5 meters, therefore point clouds of ceilings in buildings with high ceilings cannot be captured, except the mobile device is mounted on an extensible monopod or tripod. However, this unavailability in acquisition, can be also translated into information that a person is in a room with a high ceiling.

Data acquisition has to be implemented properly while a person is moving with a steady pace and without sudden movements regarding the measuring angle and height. A monopod or a tripod could be used to solve the small range of the LiDAR sensor. Concerning images, since large rooms cannot be easily captured with a single image, the sensor could be placed almost perpendicularly to the ceiling to capture a larger part of it. Additionally, small wall parts do not affect the localisation results, while larger parts should be avoided. Concerning point clouds, the Colored variation of the ICP provided the most optimal results both in terms of time complexity and quality. Concerning the image-based indoor localisation techniques, ORB takes approximately 10% of the time that SIFT takes, thus it is more suitable for real time applications, however SIFT resulted into more accurate results. The larger variety of feature detectors and descriptors that were included in Photomatch, showed that the combination of SURF feature detector with the SIFT descriptor provides the most optimal results while overlapping images of ceilings are used for feature matching. LiDAR sensors

are currently included only in the recent releases by Apple and some Android devices with a Time of Flight (ToF) camera and ARCore. This new development indicates that more mobile devices will include LiDAR sensors in the future. Pix4D Catch and SiteScape perform with an accuracy of +/-1 inch, showing that the LiDAR sensors in Apple's devices can be also used for construction projects. Promising results were also shown regarding emergency situations, such as fires in indoor environments where users should be able to find the name of the room they are located in, as well as a way to communicate that information with first-aid responders. This information is also important to first-aid responders, which are usually unaware of the number and names of the rooms in an indoor facility. However, some improvements especially concerning time efficiency have to be implemented for the final product to be used for emergency situations. Concerning location tracking, the result quality is based on the succession of the indoor localisation results. The accuracy of location tracking is in room-level as the center of each room was chosen as a representing point.

The implemented pipeline, that includes both indoor localisation from point clouds and images of the ceilings, could be applied in buildings with large rooms, such as airports, and train stations where people can easily lose their orientation. Therefore, localisation can be used as an affirmation that users are on the correct route towards their final destination. The dynamic acquisition by an automated device could help towards the optimal mapping of indoor facilities based on point cloud acquisition. Regarding location tracking, it comprises an extension of indoor localisation, as it is implemented based on two or more localisation results. Information of most used paths is vital in an indoor facility, as its manager, can use daily, weekly or monthly statistics and optimise the distribution of people in an indoor space, based on the noticeable movement patterns, while also respecting user-privacy. The importance of this information is even higher during the COVID-19 era. Last but not least, point clouds of ceilings can be used as reference to Computer-aided Design (CAD) and Building Information Model (BIM) models, in order to help the modelling of the existing utilities and their components in an indoor facility.

Future research could involve the use of machine learning algorithms, which could automatically detect the large wall planes that negatively affect the indoor localisation results based on ceilings. Additionally, feature matching based on monocular depth estimation could be tested, as an alternative way of image-based indoor localisation. The protruding installations of the ceilings, could be used in combination with an AR platform in order to recognise the different utilities in an effort to develop a landmark-based localisation approach. Furthermore, additional research could include the establishment of navigational instructions for humans and also robots, as well as navigation for specific user groups, such as people with partial or severe blindness, by incorporating the braille language in a real time application, or navigational applications that focus on people with movement disorders, who need to follow specific paths as they navigate to their desired des-

tinuation.



# Acknowledgements

First, I would like to express my gratitude to my thesis supervisor, Ir. Edward Verbree for giving me the chance to work on this topic, as well as his valuable guidance and comments during this process. Additionally, I would like to thank my second supervisor, Dr. Azarakhsh Rafiee for her valuable comments and her direct involvement since day one. They both steered me to the right direction, when this was required, while the doors to their offices were always open. I would like to express my appreciation to the co-reader of my thesis, Dr. Lucia Díaz-Vilariño, for her valuable input as an external reader, during the last part of this research.

Furthermore, I would like to acknowledge Ir. Niels van der Vaart and Jan Willem van Eck from ESRI Nederland, that helped me towards acquiring an ArcGIS Professional license, that was required for the completion of this thesis. I am also thankful to Andy Putch, the founder of SiteScape for solving my questions regarding some functionalities of this application.

A special credit belongs to my fellow Geomatics classmates, for the fun moments we had together, as well as our collaboration during the last two years, that helped me improve as an individual, as well as strengthen my technical skills.

Finally, I would like to express my profound gratitude to my family and friends for their continuous support and patience throughout the last two years at TU Delft. Especially, during last year where the pandemic restrictions overwhelmed our everyday lives, their advice and constant encouragement kept me going.

Ioannis Dardavesis



# Contents

|  |           |
|--|-----------|
| <b>1. Introduction</b>                             | <b>1</b>  |
| 1.1. Problem Statement                             | 2         |
| 1.2. Research questions                            | 3         |
| 1.3. Objective and scope                           | 4         |
| 1.4. Reading guide                                 | 4         |
| <b>2. Theoretical background and related work</b>  | <b>5</b>  |
| 2.1. Location and Position                         | 5         |
| 2.2. Localisation/Positioning techniques           | 6         |
| 2.3. Additional research in indoor localisation    | 8         |
| 2.4. Camera sensors and images                     | 8         |
| 2.4.1. Camera models                               | 8         |
| 2.4.2. Camera calibration                          | 9         |
| 2.4.3. Camera parameters                           | 9         |
| 2.5. Feature extraction and matching               | 12        |
| 2.5.1. Key-point detectors                         | 12        |
| 2.5.2. Key-point descriptors                       | 12        |
| 2.5.3. Feature matching                            | 13        |
| 2.5.4. 3D reconstruction                           | 13        |
| 2.6. LiDAR and point clouds                        | 13        |
| 2.6.1. General background                          | 13        |
| 2.6.2. Point cloud registration                    | 14        |
| 2.7. Privacy and location tracking                 | 16        |
| 2.8. Summary of related work                       | 17        |
| <b>3. Methodology</b>                              | <b>19</b> |
| 3.1. Overview                                      | 19        |
| 3.2. Data acquisition                              | 20        |
| 3.2.1. Point clouds                                | 20        |
| 3.2.2. Images                                      | 21        |
| 3.3. Point cloud pre-processing                    | 21        |
| 3.4. Point cloud registration                      | 22        |
| 3.4.1. Global registration                         | 22        |
| 3.4.2. Local refinement                            | 23        |
| 3.5. Feature matching                              | 24        |
| 3.5.1. Feature matching between single images      | 24        |
| 3.5.2. Feature matching between overlapping images | 24        |
| 3.6. Storage                                       | 25        |
| 3.7. Web-app                                       | 25        |
| 3.8. Location tracking                             | 25        |
| 3.9. Case study                                    | 26        |

|  |           |
|--|-----------|
| <b>4. Implementation and experiments</b>                         | <b>29</b> |
| 4.1. Tools   | 29        |
| 4.1.1. Software  | 29        |
| 4.1.2. PhotoMatch  | 31        |
| 4.1.3. COLMAP  | 31        |
| 4.1.4. Hardware  | 31        |
| 4.2. Datasets  | 32        |
| 4.3. Point cloud pre-processing                                  | 34        |
| 4.4. Point cloud registration                                    | 35        |
| 4.4.1. Global registration                                       | 35        |
| 4.4.2. Local refinement  | 36        |
| 4.5. Feature matching  | 37        |
| 4.5.1. Feature matching between single images                    | 37        |
| 4.5.2. Feature matching between overlapping images               | 38        |
| 4.6. Storage   | 39        |
| 4.7. Web-app   | 40        |
| 4.8. Location tracking   | 41        |
| <b>5. Results and analysis</b>                                   | <b>43</b> |
| 5.1. Indoor localisation from point clouds                       | 43        |
| 5.1.1. Point clouds from SiteScape                               | 43        |
| 5.1.2. Point clouds from Pix4D Catch                             | 46        |
| 5.1.3. Point clouds from overlapping image sets                  | 48        |
| 5.1.4. Performance parameters                                    | 50        |
| 5.2. Indoor localisation from images                             | 52        |
| 5.2.1. Single images   | 52        |
| 5.2.2. Overlapping images  | 54        |
| 5.3. Location tracking   | 58        |
| <b>6. Conclusion and discussion</b>                              | <b>61</b> |
| 6.1. Research questions  | 61        |
| 6.1.1. Answer to the secondary questions                         | 61        |
| 6.1.2. Answer to the main research question                      | 64        |
| 6.2. Discussion  | 65        |
| 6.2.1. Contribution  | 65        |
| 6.2.2. Limitations   | 66        |
| 6.3. Recommendations and future work                             | 67        |
| 6.3.1. Recommendations   | 67        |
| 6.3.2. Future work   | 67        |
| <b>A. Reproducibility and self-assessment</b>                    | <b>69</b> |
| A.1. Marks for each of the criteria                              | 69        |
| A.2. Self-reflection   | 70        |
| <b>B. Additional results</b>                                     | <b>71</b> |
| B.1. Point clouds from SiteScape                                 | 71        |
| B.2. Point clouds from Pix4D Catch                               | 74        |
| B.3. Point clouds that are reconstructed from overlapping images | 78        |
| B.4. Accuracy/precision metrics                                  | 81        |
| B.5. Feature detection in images                                 | 82        |

B.6. Feature matching in images . . . . . 84



# List of Figures

|      |   |      |
|------|---|------|
| 1.   | Selected rooms from the Faculty of Architecture and the Built environment . . .   | viii |
| 2.   | Pre-processing steps . . . . .  | ix   |
| 3.   | Web-app interface . . . . .   | x    |
| 4.   | ArcGIS Dashboard . . . . .  | x    |
| 11.  | ROC curves between 2 images with Brute-force matching . . . . .   | xiv  |
| 12.  | Ground truth and visualised paths between rooms 08.02.00.430, 08.02.00.807 and 08.02.00.470 . . . . .   | xiv  |
| 2.1. | Absolute position according to a global reference frame (left) and relative position according to a local reference frame (right) [Sithole and Zlatanova, 2016] . . | 5    |
| 2.2. | Location according to the smallest physically defined space in a building [Sithole and Zlatanova, 2016] . . . . .   | 6    |
| 2.3. | Accuracy of different indoor and outdoor positioning methods [Mautz, 2012] . .  | 7    |
| 2.4. | Pinhole camera model, acquired by [Hata and Savarese, 2022] . . . . .   | 9    |
| 2.5. | Setup of epipolar geometry, acquired by [Hata and Savarese, 2022] . . . . .   | 11   |
| 2.6. | Epipolar lines and corresponding points, acquired by [Hata and Savarese, 2022] .  | 11   |
| 2.7. | Aerial and Terrestrial LiDAR sensors . . . . .  | 14   |
| 3.1. | Methodology overview . . . . .  | 19   |
| 3.2. | LiDAR sensor of iPhone 12 pro . . . . .   | 20   |
| 3.3. | Acquisition of overlapping images . . . . .   | 21   |
| 3.4. | Indoor localisation based on point clouds . . . . .   | 23   |
| 3.5. | Indoor localisation based on feature matching . . . . .   | 24   |
| 3.6. | Indoor localisation based on point clouds that were reconstructed from overlapping images . . . . .   | 25   |
| 3.7. | Selected rooms from the Faculty of Architecture and the Built environment . . .   | 27   |
| 4.1. | CAD drawing of the Faculty of Architecture and the Built environment (West side - ground floor) . . . . .   | 34   |
| 4.2. | Pre-processing steps . . . . .  | 34   |
| 4.3. | Global and local registration . . . . .   | 36   |
| 4.4. | Processing time of different registration algorithms . . . . .  | 37   |
| 4.5. | Processing time of different feature matching combinations . . . . .  | 38   |
| 4.6. | Feature detection (left) and feature matching (right) with SIFT detectors and descriptors . . . . .   | 39   |
| 4.8. | Web-app interface . . . . .   | 41   |
| 4.9. | ArcGIS Dashboard . . . . .  | 42   |
| 5.4. | RANSAC global registration (left) and fast global registration (right) of room 560 from SiteScape . . . . .   | 44   |
| 5.5. | Point cloud of room 08.02.00.430, where the user stumbled during the acquisition  | 46   |

List of Figures

|   |    |
|---|----|
| 5.9. RANSAC global registration (left) and fast global registration (right) of room 560 from Pix4D Catch . . . . .                            | 47 |
| 5.13. RANSAC global registration (left) and fast global registration (right) of room 560 from reconstructed images of ceilings . . . . .      | 49 |
| 5.17. Feature detection with SIFT for room 430 . . . . .  | 54 |
| 5.18. ROC curves between 2 images with Brute-force matching . . . . .   | 56 |
| 5.19. ROC curves between 2 images with FLANN matching . . . . .   | 56 |
| 5.20. Total number of matches and true positive matches . . . . .   | 57 |
| 5.23. Ground truth and visualised paths between rooms 08.02.00.430, 08.02.00.807 and 08.02.00.470 . . . . .                                   | 59 |
| A.1. Reproducibility criteria to be assessed. . . . .   | 69 |
| B.4. RANSAC global registration (left) and fast global registration (right) of room 430, from point clouds acquired by SiteScape . . . . .    | 71 |
| B.8. RANSAC global registration (left) and fast global registration (right) of room 470, from point clouds acquired by SiteScape . . . . .    | 72 |
| B.12. RANSAC global registration (left) and fast global registration (right) of room 807, from point clouds acquired by SiteScape . . . . .   | 73 |
| B.16. RANSAC global registration (left) and fast global registration (right) of room 808, from point clouds acquired by SiteScape . . . . .   | 74 |
| B.20. RANSAC global registration (left) and fast global registration (right) of room 430, from point clouds acquired by Pix4D Catch . . . . . | 75 |
| B.24. RANSAC global registration (left) and fast global registration (right) of room 470, from point clouds acquired by Pix4D Catch . . . . . | 75 |
| B.28. RANSAC global registration (left) and fast global registration (right) of room 807, from point clouds acquired by Pix4D Catch . . . . . | 76 |
| B.32. RANSAC global registration (left) and fast global registration (right) of room 808, from point clouds acquired by Pix4D Catch . . . . . | 77 |
| B.36. RANSAC global registration (left) and fast global registration (right) of room 430 from reconstructed images of ceilings . . . . .      | 78 |
| B.40. RANSAC global registration (left) and fast global registration (right) of room 470 from reconstructed images of ceilings . . . . .      | 79 |
| B.44. RANSAC global registration (left) and fast global registration (right) of room 807 from reconstructed images of ceilings . . . . .      | 80 |
| B.48. RANSAC global registration (left) and fast global registration (right) of room 808 from reconstructed images of ceilings . . . . .      | 81 |
| B.50. Scatter plot with centers of reference and user point clouds after point cloud matching in room 08.02.00.560 . . . . .                  | 81 |
| B.51. Feature detection in room 08.02.00.430 . . . . .  | 82 |
| B.52. Feature detection in room 08.02.00.470 . . . . .  | 82 |
| B.53. Feature detection in room 08.02.00.560 . . . . .  | 83 |
| B.54. Feature detection in room 08.02.00.807 . . . . .  | 83 |
| B.55. Feature detection in room 08.02.00.808 . . . . .  | 84 |
| B.57. Feature matching in room 08.02.00.430 . . . . .   | 85 |
| B.59. Feature matching in room 08.02.00.470 . . . . .   | 86 |
| B.61. Feature matching in room 08.02.00.560 . . . . .   | 87 |
| B.63. Feature matching in room 08.02.00.807 . . . . .   | 88 |
| B.65. Feature matching in room 08.02.00.808 . . . . .   | 89 |

# List of Tables

|      |  |     |
|------|--|-----|
| 1.   | Number of correct matches per point cloud registration algorithm . . . . .                     | x   |
| 2.   | Number of correct matches per point cloud registration algorithm . . . . .                     | xi  |
| 3.   | Number of correct matches per point cloud registration algorithm . . . . .                     | xii |
| 4.   | Number of correct matches per feature detection, description and matching techniques . . . . . | xii |
| 4.1. | Number of points per room, acquired with SiteScape . . . . .                                   | 33  |
| 4.2. | Number of points per room, acquired with Pix4D Catch . . . . .                                 | 33  |
| 4.3. | Number of overlapping images per room . . . . .  | 33  |
| 4.4. | Number of points before and after pre-processing, acquired with SiteScape . . . . .            | 35  |
| 4.5. | Number of points before and after pre-processing, acquired with Pix4D Catch . . . . .          | 35  |
| 5.1. | Number of correct matches per point cloud registration algorithm . . . . .                     | 45  |
| 5.2. | Number of correct matches per point cloud registration algorithm . . . . .                     | 47  |
| 5.3. | Number of correct matches per point cloud registration algorithm . . . . .                     | 49  |
| 5.4. | Number of correct matches per feature detection, description and matching techniques . . . . . | 52  |
| 5.5. | Number of keypoints and their percentage used for matching per combination . . . . .           | 55  |
| A.1. | Evaluation of reproducibility criteria . . . . .   | 69  |



# List of Algorithms

|  |    |
|--|----|
| 3.1. Point cloud preprocessing . . . . . | 22 |
| 3.2. Location tracking . . . . .         | 26 |



# Acronyms

|   |      |
|---|------|
| <b>LiDAR</b> Light detection and ranging . . . . .                    | v    |
| <b>SIFT</b> Scale Invariant Feature Transform . . . . .               | v    |
| <b>GNSS</b> Global Navigation Satellite Systems . . . . .             | v    |
| <b>ICP</b> Iterative Closest Point . . . . .                          | v    |
| <b>SURF</b> Speeded-Up Robust Features . . . . .                      | v    |
| <b>SLAM</b> Simultaneous localisation and mapping . . . . .           | vii  |
| <b>AR</b> Augmented Reality . . . . .                                 | vii  |
| <b>RSS</b> Received signal strength . . . . .                         | vii  |
| <b>WLAN</b> Wireless local area network . . . . .                     | vii  |
| <b>RANSAC</b> Random sample consensus . . . . .                       | viii |
| <b>ORB</b> Oriented FAST and Rotated BRIEF . . . . .                  | ix   |
| <b>FPFH</b> Fast Point Feature Histogram . . . . .                    | ix   |
| <b>FLANN</b> Fast Library for Approximate Nearest Neighbors . . . . . | ix   |
| <b>RMSE</b> Root Mean Squared Error . . . . .                         | ix   |
| <b>BRIEF</b> Binary Robust Independent Elementary Features . . . . .  | xiii |
| <b>ROC</b> Receiver Operating Characteristic . . . . .                | xiv  |

*LIST OF ALGORITHMS*

|  |    |
|--|----|
| <b>CAD</b> Computer-aided Design . . . . .                   | xv |
| <b>ToF</b> Time of Flight . . . . .                          | xv |
| <b>BIM</b> Building Information Model . . . . .              | xv |
| <b>RFID</b> Radio frequency identification . . . . .         | 2  |
| <b>GPS</b> Global Positioning System . . . . .               | 5  |
| <b>RSSI</b> Received signal strength information . . . . .   | 6  |
| <b>ToA</b> Time of Arrival . . . . .                         | 7  |
| <b>AoA</b> Angle of Arrival . . . . .                        | 7  |
| <b>IMU</b> Inertial Measurement Unit . . . . .               | 7  |
| <b>INS</b> Inertial Navigation System . . . . .              | 7  |
| <b>UWB</b> Ultra-Wideband . . . . .                          | 7  |
| <b>FAST</b> Features from Accelerated Segment Test . . . . . | 12 |
| <b>UAV</b> Unmanned Aerial Vehicle . . . . .                 | 14 |
| <b>PFH</b> Point Feature Histogram . . . . .                 | 15 |
| <b>GDPR</b> General Data Protection Regulation . . . . .     | 16 |
| <b>LBS</b> Location Based Services . . . . .                 | 16 |
| <b>IPS</b> Inertial Positioning Systems . . . . .            | 16 |
| <b>HTML</b> HyperText Markup Language . . . . .              | 29 |
| <b>UUID</b> Universal Unique Identifier . . . . .            | 30 |

*LIST OF ALGORITHMS*

|   |    |
|---|----|
| <b>CSS</b> Cascading Style Sheets . . . . .                       | 30 |
| <b>GIS</b> Geographic Information System . . . . .                | 30 |
| <b>SfM</b> Structure from Motion . . . . .                        | 31 |
| <b>MVS</b> Multi-View Stereo . . . . .                            | 31 |
| <b>PLY</b> Polygon File Format . . . . .                          | 32 |
| <b>URL</b> Uniform Resource Locator . . . . .                     | 40 |
| <b>BRISK</b> Binary Robust Invariant Scalable Keypoints . . . . . | 57 |



# 1. Introduction

Nowadays, the evolution of localisation and navigation technologies is vast, aiding towards facilitating users' guidance in various environments. Outdoor positioning can be easily achieved, with the widely used GNSS, which are included in every person's mobile device. However, due to the presence of high buildings in dense urban environments and bad reception in indoor environments, the performance of GNSS is significantly degraded. Therefore, alternative ways of positioning and localisation respectively, need to be explored.

Even during the early centuries, when the recent technological discoveries were not available, localisation and navigation were important fields, that could not be adequately tackled, even by the most renowned scientists of that era. The calculation of a relative position within a coordinate reference system, based on star observations was an initial attempt to comprehend a person's location in space, as sailors needed to navigate towards their final destination. The maps of those eras were giving a sense of location, as identifiable objects were linked to relative positions of the users [Sobel, 1998]. In both of the aforementioned examples, landmarks are introduced in an attempt to accomplish localisation.

Landmarks play a significant role in both outdoor and indoor space. Salient objects, such as high-rise buildings facilitate guidance in outdoor environments. In these environments, landmarks can be both local and distant depending on the visibility, in contrast to the indoor space where corners and walls tend to block the user's vision. Indoor spaces contain regular geometries, such as room boundaries, as opposed to the outdoor space [Yang and Worboys, 2011]. These points have to be taken into consideration, indicating that different strategies have to be followed, as a means to accomplish indoor and outdoor landmark-based localisation.

Regarding indoor environments, the lower density of landmarks and the absence of outstanding elements frequently result to easier loss of orientation, compared to outdoors [Michon and Denis, 2001]. As Wadden and Scheff mentioned, people spend around 80 % of their time indoors, or even higher according to other researchers, therefore localisation comprises an important problem, in public buildings, such as airports or train stations, that usually consist of chaotic spaces, leading to a higher chance that an individual will become disoriented. Therefore, the necessity of a dynamic indoor localisation system is apparent, so that a person can navigate in an indoor facility, especially if that person is exploring it for the first time. In that scenario, a location provider tool with adequate precision could be a significant aid. This tool could be applied in various indoor spaces, such as museums and art galleries [Gupta et al., 2016]. In that context, artificial landmarks (signs) as well as natural landmarks (plants) can help users retrieve their location. In addition, landmarks can also be used as an affirmation that individuals are on the correct route towards their final destination [Hile and Borriello, 2008].

## 1. Introduction

Indoor localisation could also be applied to deal with emergency situations, such as fires in complex indoor spaces. Persons in need could access the name of their current location, based on an indoor localisation application and transmit this information to the first-aid responders. The latter need guidance, related to the location of the person in need, as well as a way to reach that location [Yang and Worboys, 2011]. Additionally, other applications of indoor localisation include the use of mobile autonomous units, in an effort to establish an indoor intelligent environment. Mobile service robots can be exploited for assisted living, setting up a smart living environment for elderly people and perform transportation and human interaction tasks. These types of robots use artificial landmarks in the form of a grid of passive Radio frequency identification (RFID) in the floor [Koch et al., 2007]. Robotic systems could also be used in working environments, by deriving a topological map based on the room geometries [Schmidt et al., 2006].

Currently, various sensors, such as Wi-Fi fingerprinting and Bluetooth, are widely used for precise localisation in indoor environments. These technologies, as well as Arduino [Mitolinos et al., 2010] and Raspberry Pi, work by performing triangulation or trilateration of different users and become the basis of different indoor localisation applications. Other applications use optic sensors to achieve object recognition (Google Lens). Recently some new applications have emerged, that use AR combined with the SLAM algorithm, which works by scanning an indoor environment in order to find a person's location in this space [Oostwegel, 2020]. It has to be mentioned that these applications work on a local level compared to outdoors, meaning that indoor localisation provides contextual information on a person's location in a sub-room or room level, while the exact coordinates cannot be derived.

### 1.1. Problem Statement

Compared to outdoor environments where GNSS comprises a universal standard for positioning, the same cannot be claimed for indoor environments [Lymberopoulos et al., 2015]. This occurs because each of the different indoor localisation techniques that are currently implemented, has its own bottlenecks.

The most widely used technology, Wi-Fi fingerprinting, is based on the comparison of RSS values with a reference radio map that translates signal values to positions [Pérez-Navarro et al., 2019]. Creating this radiomap is a heavy and time-consuming task. Many fingerprints are required and a single change in the Wi-Fi infrastructure, requires the map to be created again. Therefore an up-to-date radio-map of the signals in an indoor facility is mandatory for this technique to function. Moreover, the availability of Wi-Fi signals in indoor environments might be irregular, which could be an outcome of poor WLAN planning in the facility. An additional aspect that needs to be taken into account is the fluctuation in the indoor population, which as was proved by [Garcia-Villalonga and Perez-Navarro, 2015] can significantly affect the RSS and therefore the accuracy of the results. Similarly, alternatives that are based on wireless technologies, such as Bluetooth, that requires the installation of costly Bluetooth hotspots, are affected by the aforementioned problems.

Other indoor localisation solutions combine SLAM and AR technologies by taking into account different sensors, such as RGB and depth cameras. Simple AR applications can be ran in a basic smartphone, however additional devices are usually required, to offer a solution with a deeper understanding of the indoor environment. Google Glass and Microsoft Hololens, which is a head-mounted display AR device and has the ability of visualising 3D models, while it includes its own processing unit [Kim et al., 2017], are some examples of AR devices.

A lot of research has been implemented in the field of outdoor navigation and localisation, in contrast to indoor environments, where the developments are relatively new and mostly involve Bluetooth sensors and Wi-Fi fingerprinting, in an effort to achieve localisation. The problems that emerge in the existing techniques and the lack of a universal solution [Pérez-Navarro et al., 2019], provide a space for innovation by utilising other techniques, that take into advantage camera and LiDAR sensors. While cameras sensors exist in every mobile device, the use of LiDAR sensors is exponentially increased, as they were also recently included in the latest releases of Apple's iPhone and iPad devices, showing that they will be a major part of the mobile devices that will follow in the upcoming years. Therefore, the challenge of achieving indoor localisation in different environments, is to find a technique that does not depend on costly and hard to access indoor sensor networks, but to use features that are accessible to everyone in their mobile device [Willems, 2017].

## 1.2. Research questions

Defining the main and secondary research questions, is a crucial part of the thesis, aiming to address indoor localisation and ensure the concreteness of this project. Therefore, the primary research question is formed as follows:

*"To what extent can ceilings with characteristic details be used for indoor localisation purposes?"*

In order to obtain a better understanding of the concept and be able to answer the main research question robustly, some complementary research questions are formed.

1. *"Which parameters (measuring angle, height, part of the room) should the user take into account while acquiring point clouds and images of ceilings?"*
2. *"Which is the optimal point cloud registration algorithm to achieve indoor localisation from ceiling data?"*
3. *"Which is the optimal image matching algorithm to achieve indoor localisation from ceiling data?"*
4. *"Are LiDAR point clouds acquired by an iPhone device an accurate and accessible solution towards indoor localisation?"*
5. *"Can the proposed pipeline aid towards facilitating localisation in emergency situations?"*
6. *"How accurate is location tracking and does it respect user privacy?"*

## 1. Introduction

### 1.3. Objective and scope

This research is implemented, in order to explore the possibility of the ceilings in public or semi-public buildings, being used for indoor localisation purposes, by providing an accessible solution that makes use of features that are available in a simple mobile device. Indoor localisation is a fundamental step and is considered as the basis towards achieving indoor navigation. Therefore, this research focuses on localisation, as well as near real-time location tracking of different users' locations for the purpose of discovering different movement patterns and not on navigation, which will be a part of the future work recommendations of this thesis. Additionally, the case study will take place in the Faculty of Architecture and the Built Environment at TU Delft, that consists of rooms, whose ceilings include installations, that can aid towards revealing the unique identity of each room. As a result, the recommended pipeline will not be applied to ceilings that do not consist of characteristic details. Data acquisition and specific instructions on how a user can perform it, in order to achieve optimal results will be also included. LiDAR sensors incorporated in the latest iPhone devices and non-commercial applications will be used to acquire and manipulate point cloud data, while camera sensors will be used to acquire images of the ceilings. The thesis will delve into the use of ceilings for indoor localisation, whereas their automatic detection will not be discussed. An additional focus of this research will be the implementation and assessment of different image matching and point cloud registration techniques, as a means to obtain the optimal localisation results. Last but not least, a web application is produced in order to visualise the localisation results and a dashboard that will serve as a location tracking platform, which can include daily, weekly or monthly statistics about the use of different paths in the Faculty of Architecture and the Built Environment. The goal of the latter is to aid towards the discovery of different movement patterns, an information that can help finding the optimal distribution of people in an indoor space.

### 1.4. Reading guide

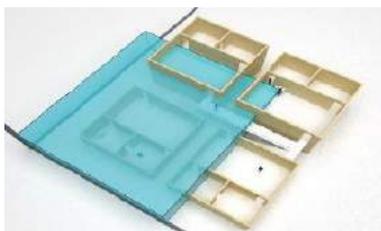
This thesis research consists of six main chapters. Except for this introduction, [Chapter 2](#) provides an overview and analysis of the theoretical concepts, that are required in order to understand this thesis in depth. The related scientific research that is fundamental to the current thesis and an overview of the different algorithms that are implemented, are included in this chapter. The overview and the analysis of the followed methodology and the specific steps that are followed, are discussed in [Chapter 3](#). [Chapter 4](#) focuses on the technical specifications and details of how the proposed methodology was implemented. [Chapter 5](#) includes the implementation results and their extensive analysis. Furthermore, a discussion regarding the outcome of this thesis and its conclusions are presented in [Chapter 6](#), as well as the answers to the aforementioned research questions. Additionally, the contribution of this thesis in the research community, as well as the limitations of the current approach are presented in the same chapter. Last but not least, potential future research based on the conclusions of the current thesis is also proposed.

## 2. Theoretical background and related work

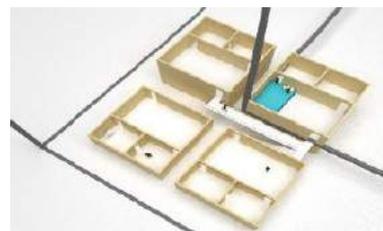
This chapter focuses on the theoretical explanation of the concepts that are applied in this thesis and are required, so that the reader obtains full knowledge of the thesis. Some theoretical aspects of this chapter might be considered general knowledge by someone in the GEO-ICT department, however they have to be explained, so that a less related reader is introduced to the concepts of the thesis. An overview of the implemented research and its relevance to this thesis research will be included in this chapter.

### 2.1. Location and Position

To begin with, it is highly important to explain the differences between two key concepts: localisation and positioning. Position refers to the exact coordinates of a person or an object in a reference coordinate system. The position of individuals or objects in an indoor environment can be specified as a pin-point placement according to a global reference system of Cartesian coordinates that is specified for a building (Figure 2.1a). Position can be also considered as relative, when it is relative to a local reference frame (Figure 2.1b) [Sithole and Zlatanova, 2016]. GNSS are widely used in outdoor environments and are included in all recent mobile devices, enabling the user to find his current position within a certain accuracy, by using for example Google Maps. Global Positioning System (GPS) is a term that is widely used, however it is only a category of GNSS. While GNSS are considerably accurate in outdoor environments, their accuracy might be significantly decreased in presence of high-rise structures, where there is no line-of-sight between the satellites and the receiver, leading to signal attenuation. This phenomenon also called urban canyon effect, leads to inaccurate positioning results in high-dense urban environments. Moreover, GNSS is vulnerable to jammer devices, which started for military purposes but their use has been extended to the public, and also spoofing devices, that transmit fake signals, leading to false solutions from user equipment [Groves, 2013].



(a) Absolute position



(b) Relative position

Figure 2.1.: Absolute position according to a global reference frame (left) and relative position according to a local reference frame (right) [Sithole and Zlatanova, 2016]

## 2. Theoretical background and related work

Concerning indoor environments, GNSS signals are typically between 15 and 40 dB weaker compared to outdoors. A combination of different factors, such as the material of the building and the multipath interference can lead to signal blockage [Groves, 2013], creating the need for alternative solutions in indoor environments. The term localisation comes into life in order to bridge the gap between outdoor and indoor environments. In contrast to position, location does not refer to exact coordinates related to a global or local reference system, but defines a general placement relative to the smallest defined physical space in an indoor facility, that could be a room, stairs or a corridor (Figure 2.2). The uncertainty in the position of an individual is determined by the extent of the room [Sithole and Zlatanova, 2016]. In that manner, in an indoor environment, localisation operations could provide contextual information about a person's or object's location in space, meaning the room or section of an indoor facility. Therefore, an indoor map of the corresponding facility is required, in order to acquire a location.

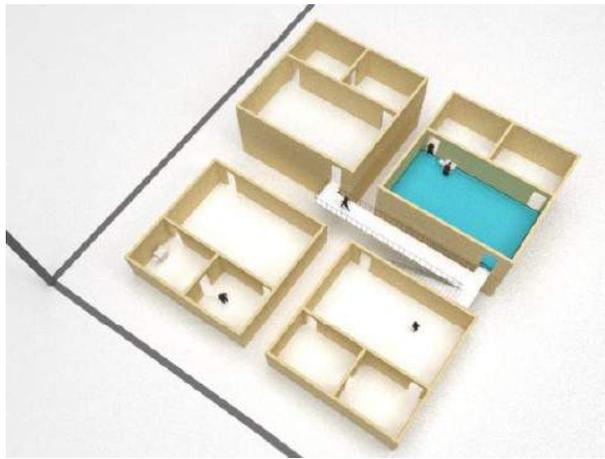


Figure 2.2.: Location according to the smallest physically defined space in a building [Sithole and Zlatanova, 2016]

### 2.2. Localisation/Positioning techniques

The following techniques that are based on the categorisation of Pérez-Navarro et al. can be used either for positioning or localisation depending on the use case. The differentiation between these two terms is explained in section Section 2.1 based on the research of Sithole and Zlatanova.

- Proximity: Proximity methods can only provide relative location information. They take into advantage the ability of mobile cellular networks to identify the approximate position of a mobile device, by finding the cell site that the device is using at the time. It relies upon a dense antenna grid, with known positions. In case the mobile device is using more than one cells, the one with the strongest signal is considered the better solution [Liu et al., 2007].
- Distance: These types of mechanisms are based on measuring the distance between transmitters and receivers, exploiting the Received signal strength information (RSSI) to acquire location in space, using triangulation and trilateration techniques [Koyuncu and

## 2.2. Localisation/Positioning techniques

Yang, 2010]. According to Bose and Foh, information such as the location of the emitters is required. Except for the distance between the transmitter and receiver, signal strength in indoor environments is also affected by the different obstacles. Concerning the transmitters, they should be strategically placed in a building, with proper distance between each other.

- **Time of Arrival (ToA):** These methods are based on calculating the absolute travel time of a signal from a transmitter to a receiver. This euclidean distance is calculated by multiplying the travel time of the signal to the wave speed. Required knowledge is the dielectric constant that depends on the building material. These methods are affected by the multi-path problem, therefore they cannot be adequately applied in indoor environments. This problem can be addressed by using different frequency bands [Mautz, 2012].
- **Angle of Arrival (AoA):** This localisation technique uses triangulation, in order to calculate the angle between a wireless access point and a mobile device using the received signal from a multiple antenna ray. The accuracy of this technique is degraded while distance is increasing [Wong et al., 2008].
- **Inertial:** This method requires a sensor that is carried by a user and can provide information about orientation and speed. A mounted Inertial Measurement Unit (IMU), usually includes accelerometers and gyroscopes, as well as other sensors. Inertial methods are used by taking into account the aforementioned sensors, in order to estimate a person's position while walking, combined with an Inertial Navigation System (INS) [Jiménez et al., 2010].
- **Fingerprinting:** One of the most common indoor localisation techniques, that compares RSS values with a reference radio map of RSS, that needs to be constantly up-to-date, so that signal values are associated with location in space [Honkavirta et al., 2009; Kae-marungsi and Krishnamurthy, 2004]. Wi-Fi fingerprinting is an example of one of the most common fingerprinting methods.

The aforementioned techniques can be used with different technologies, dedicated to indoor localisation such as Ultra-Wideband (UWB), RFID and Bluetooth, or Wi-Fi. Several techniques, include the integration of different technologies in the same system [Pérez-Navarro et al., 2019].

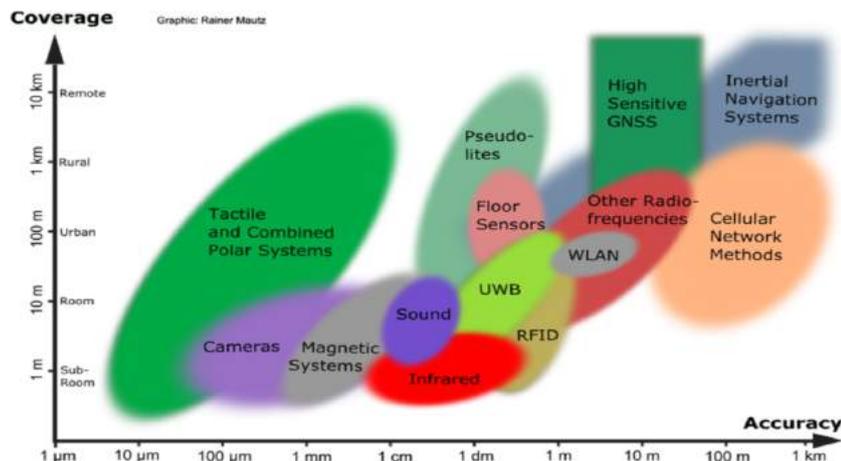


Figure 2.3.: Accuracy of different indoor and outdoor positioning methods [Mautz, 2012]

## 2. Theoretical background and related work

### 2.3. Additional research in indoor localisation

Various research has been implemented for indoor localisation purposes, with different techniques that were presented in [Section 2.2](#). A pure landmark based approach for indoor localisation was proposed by [Willems](#), where he discussed the reliability of landmarks for indoor localisation and how the constellation and number of landmarks affects the results. Furthermore, another complementary research has been implemented by [Fratzeskou et al.](#), as part of the "Synthesis Project" course, which is included in the MSc Geomatics course syllabus of TU Delft. Specifically, an indoor localisation method is described, where users acquire data, such as images, videos and point clouds of the ceilings to retrieve their current location. The authors discuss a methodology where point clouds of ceilings are uploaded in a database, to represent different rooms of the indoor space. The comparison of these reference point clouds, with the ones acquired by users and the calculation of the best match, will reveal the room where the user is located. The signatures of the point clouds are calculated, based on two methods: feature and histogram matching, as well as the integration of these methods. This thesis focuses on feature matching to perform point cloud registration based on feature and not histogram matching, while using LiDAR and camera sensors that are included in mobile devices. At the same time, an important aspect of this thesis is the implementation of a real-time indoor localisation approach and in extent location tracking of users, based on both point clouds and images. Moreover, a dynamic approach that involves continuous point cloud acquisition, while a user or an automated device are moving between rooms is also implemented in the current thesis.

### 2.4. Camera sensors and images

Currently, cameras are widely used by various individuals in order to capture the real world. However camera sensors and the images that they produce can be used in order to solve complex problems related to photogrammetry and computer vision. Photogrammetry, is a science that has as its main outcome the production of topographic maps, based on acquired images that are collected with different techniques, that can be both terrestrial and airborne for small and large scale projects respectively. In parallel, camera sensors are useful in computer vision, in order to support operations, such as object recognition and navigation in autonomous vehicles [[Hartley and Mundy, 1993](#)].

Images incorporate 2D representations of the real-world and include geometric information, such as points and lines, as well as photometric information that could be color or intensity. However, some features of the real world scene may not be preserved [[Kaniouras et al., 2019](#)]. Thus, it is possible to calculate the coordinates of a 3D point, based on the camera's parameters and at least two images with a minimum overlap. This procedure will be thoroughly explained in section [Section 2.5](#).

#### 2.4.1. Camera models

There are various camera models, however pinhole camera model comprises one of the models that represent simple camera geometry and it is commonly used as the reference system for geometrical relations [[Zhang, 2000](#)]. The design of this system includes placing a barrier with a small aperture between the camera and the object. The aperture exists in order to avoid multiple

rays from various points in the object influencing the camera film. In that way, only a few rays pass through the aperture, enabling the possibility of one-to-one mapping between the film and the points of the object [Hata and Savarese, 2022].

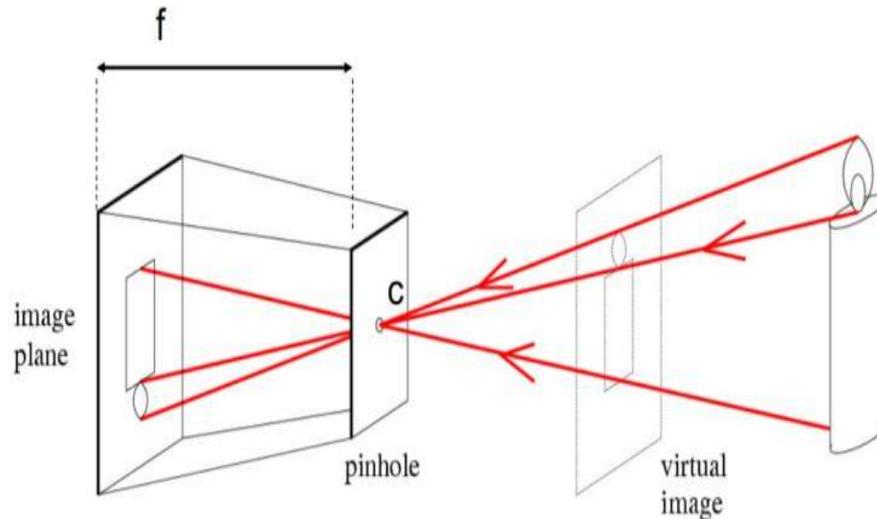


Figure 2.4.: Pinhole camera model, acquired by [Hata and Savarese, 2022]

### 2.4.2. Camera calibration

The first step towards 3D reconstruction of an object or a scene, is the calibration of the camera. Features of the real-world scene can be reconstructed, by calculating the camera parameters. The latter, are specific for each camera and create distortions in an image. Their calculation aims to find these distortions and aid towards an accurate 3D reconstruction [Zhang, 2000].

### 2.4.3. Camera parameters

The main goal of camera calibration is the establishment of parameters that describe dependencies between the camera's and the real coordinate system, as well as parameters that describe physical characteristics of instrument and transformation of the perspective [Kaniouras et al., 2019]. The camera parameters can be divided into intrinsic and extrinsic.

Intrinsic parameters: Describe the internal parameters of a camera and more specifically the focal length, offset and skewness. 2 parameters are needed for focal length, 2 for offset and 1 for skewness. The intrinsic parameters require a theoretical model of the camera to be calculated.

Extrinsic parameters: Describe the position of the camera relatively to the real object that is being captured. Specifically there are 6 extrinsic parameters to be computed, including camera's rotation and translation for every axis of a coordinate system, resulting into 6 extrinsic parameters in total.

## 2. Theoretical background and related work

Even if the camera parameters are unknown, they can be retrieved based on the images that the camera takes. The 11 unknown parameters included in the camera matrix  $K$ , require 6 correspondences to solve this problem. Therefore, 6 equations are created and comprise a linear system. Since there are 12 correspondences the system is overdetermined.

$$\begin{bmatrix} P_1^T & 0^T & -u_1 P_1^T \\ 0^T & P_1^T & -v_1 P_1^T \\ & \vdots & \\ P_n^T & 0^T & -u_n P_n^T \\ 0^T & P_n^T & -v_n P_n^T \end{bmatrix} \begin{bmatrix} m_1^T \\ m_2^T \\ m_3^T \end{bmatrix} = \mathbf{P}m = 0 \quad (2.1)$$

For this linear system,  $m=0$  is always a solution. Therefore, a minimisation is required in order to constrain this problem. Specifically, singular value decomposition is used, so that the intrinsic and extrinsic parameters are calculated.

$$\underset{m}{\text{minimize}} \quad \|\mathbf{P}m\|^2 \quad (2.2)$$

$$P = UDV^T \quad (2.3)$$

$$M = \frac{1}{\rho} \begin{bmatrix} \alpha r_1^T - \alpha \cot \theta r_2^T + c_x r_3^T & \alpha t_x - \alpha \cot \theta t_y + c_x t_z \\ \frac{\beta}{\sin \theta} r_2^T + c_y r_3^T & \frac{\beta}{\sin \theta} t_y + c_y t_z \\ r_3^T & t_z \end{bmatrix} \quad (2.4)$$

Solving the last equation for the intrinsic and extrinsic parameters, can result to their calculation. The exact derivation of this procedure can be found in [Forsyth and Ponce, 2011], while the derivation of the minimisation problem in [Hartley and Zisserman, 2004].

The aforementioned camera calibration procedure takes place based on a single view. However, in order to successfully reconstruct a real-world scene, multiple images are required, so that certain ambiguities are avoided and the loss of information is prevented. For this reason, epipolar geometry is used when multiple cameras are present. Epipolar geometry combines the relationship between the cameras, the 3D points and the observations, meaning the projection of each point in each camera's image plane.

With epipolar geometry, the fundamental matrix is calculated and includes information about the camera matrices  $K$  and  $K'$ , as well as the extrinsic parameters, meaning the relative translation and rotation between the cameras. The importance of this matrix, lies in the fact that by knowing a point in an image, the respective position of the same point in the other image can be found. Therefore, it is possible to establish a relationship between these points without knowing the actual position of the point in 3D space, or the intrinsic and extrinsic characteristics of the used cameras. Using triangulation, the location of a 3D point can be calculated, based on its projections into two images.

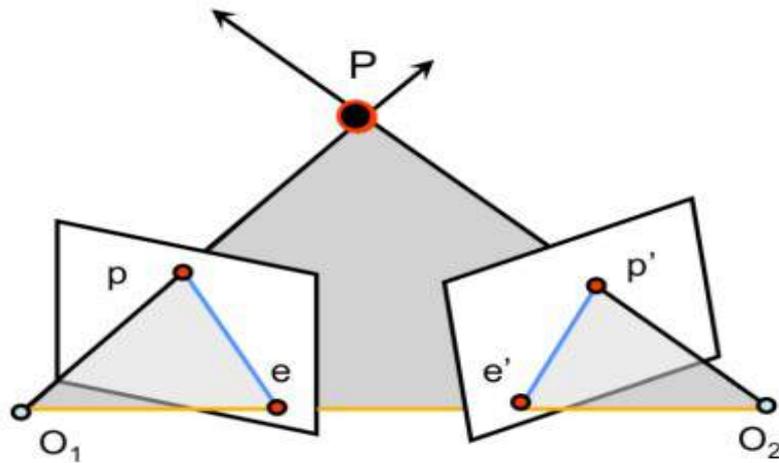


Figure 2.5.: Setup of epipolar geometry, acquired by [Hata and Savarese, 2022]

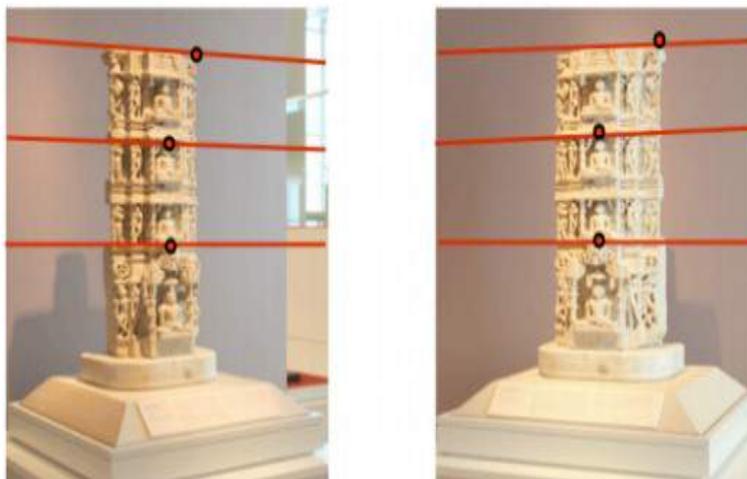


Figure 2.6.: Epipolar lines and corresponding points, acquired by [Hata and Savarese, 2022]

Epipolar geometry can be extended into multiple views, instead of two, with structure from motion, where based on observations from multiple views, the camera parameters and the 3D structure of the scene can be simultaneously determined. The aforementioned process that is based on epipolar geometry, is extended to multiple views by using pairwise cameras. A good method to solve the Structure from motion problem is bundle adjustment, when multiple cameras are combined. Different methods, such as the 8-point algorithm proposed by [Longuet-Higgins](#) where 8 pairs of corresponding points between images are known and its normalised version, can help towards calculating the fundamental matrix, when the camera parameters are unknown. The detection of corresponding points between 2 images is called feature match-

## 2. Theoretical background and related work

ing, where based on different algorithms, key-points of different images are detected and then matched, so that the corresponding points between 2 or more images are found. An additional filtering follows, so that false matches are prevented. The last step is the surface reconstruction, which is out of the scope of this thesis.

### 2.5. Feature extraction and matching

Currently, various techniques have been developed, related to feature matching, using different combinations of detectors, descriptors and matching techniques. An overview of the different techniques is included in this section.

#### 2.5.1. Key-point detectors

Different detectors have been created, in order to find unique points in each image. One of the most characteristic examples is [SIFT](#), which uses differences of Gaussians, where image pyramids are created by using an iterative convolution of the original image with Gaussian kernels. In each pyramid level, a pixel is compared with 8 neighbouring cells in the current image and with its 9 neighbors of its adjacent pyramid images. The key-points are detected where local maxima occur in the differences between the Gaussian images [[Lowe, 2004](#)]. [SURF](#), was developed by [Bay et al.](#) to speed up [SIFT](#) results and uses Fast Hessian to detect key-points. Convolution filters are used, to approximate second order Gaussian derivatives for each image point. The key-points are detected in areas where the determinant is maximal through non-maximal suppression. [ORB](#), created by [Rublee et al.](#) is a corner detection algorithm that uses Features from Accelerated Segment Test ([FAST](#)) in order to detect stable key-points in different scales. It works by selecting the strongest corners using [FAST](#) or the Harris Corner score, and calculates the orientation with first-order moments. [FAST](#) developed by [Rosten and Drummond](#) is an algorithm suitable for real-time applications and is based on the [ast!](#) ([ast!](#)), which distinguishes key-points by checking the intensity values of 16 neighboring pixels around the candidate key-point in a circular pattern.

#### 2.5.2. Key-point descriptors

Descriptors are used in order to describe the detected key-points. [BRIEF](#) is a short binary descriptor that extracts intensities in the form of bit strings, which are compared along the same lines. It uses binary tests on smooth image patches and takes advantage of pre-trained binary tests on classification trees to obtain the signature of arbitrary key-points. However, a drawback is that it is not invariant to scale and rotation. Some aforementioned key-point detectors are also used as descriptors. [ORB](#) is similar to [BRIEF](#), however it is additionally invariant to rotation and is robust to noise. It uses a rotation-aware version of [BRIEF](#) and a learning method to apply binary tests, identifying features with high variance and low correlation [[Rublee et al., 2011](#)]. Furthermore, [SIFT](#) calculates the magnitude and orientation for 16 sub-regions around the selected key-point in a specific scale, constructing a 128 element non-binary vector. It is invariant to certain illumination changes, due to the normalisation of the values. [SURF](#) uses Haar wavelets and calculates responses for 16 sub-regions around the detected feature, which are then further subdivided, resulting into a 64-element descriptor for each feature. Similarly

to SIFT, it is also invariant to certain illumination changes, however it is significantly faster in terms of time efficiency.

### 2.5.3. Feature matching

After choosing and implementing the detectors and descriptors for an image set, an algorithm has to be chosen, so that the required features are matched. Brute-Force, is used by comparing a feature descriptor of an image with all the features in the second set. Afterwards, their distance is calculated and the closest feature is acquired. Geometric tests, such as the calculation of the fundamental matrix using the 8-point algorithm can be used to validate the results [Jakubović and Velagić, 2018]. Another feature matching technique, FLANN uses nearest neighbor search, therefore it is more efficient than Brute-Force when large datasets and high-dimensional features are involved Muja and Lowe [2009]. Additionally, the robustness of the matching algorithms can be tested by calculating the recall (Equation 2.5) that will reveal the number of the matches that were actually true. This calculation requires setting up a ground truth based on common and distinct points in the compared images. A research that tests different combinations of detectors, descriptors and feature matching techniques, resulting in an open source tool was implemented by González-Aguilera et al.

$$\text{Recall} = \frac{TP}{TP + FN} \quad \text{where} \quad TP = \text{True Positive}, \quad FN = \text{False Negative} \quad (2.5)$$

### 2.5.4. 3D reconstruction

Various research has been implemented about scene reconstruction from image sets, that include two or more images. Mohr et al. uses reference points and multiple uncalibrated images to perform 3D reconstruction. The research involves selecting an optimal geometric method towards reference points selection, as well as testing in simulated and real images. Similarly, Stathopoulou and Remondino describe and evaluate different pipelines on how to perform 3D reconstruction from multiple images, even when the camera parameters are unknown. The development of computer vision and deep learning, has led to the development of techniques that require a single image to perform 3D reconstruction. The research of [Ping et al., 2021] includes training a neural network dataset with images and focuses on their boundaries. Specifically, the original image and the edge map are used as input, in order to make a prediction of a point cloud. Additionally, ICP is used to compare the ground truth with the predicted point cloud and calculate the differences between them.

## 2.6. LiDAR and point clouds

### 2.6.1. General background

LiDAR comprises a measurement method that has seen rapid advancement after the invention of laser in 1960. It uses the ToF measuring principle for imaging, which includes depth measurements by counting time delays in cases where light is emitted from a source. Specifically,

## 2. Theoretical background and related work

**LiDAR** is an active non-contact range-finding technique, in which an optical signal is emitted from a sensor to an object and the reflected or back-scattered signal is detected and processed, to determine the distance between the sensor and the object. The distance is calculated based on the time that it takes for the emitted signal to reach the object and find its way back to the detector, multiplied by the speed of light [Royo and Ballesta-Garcia, 2019]. Using the measured distance to an object and the sensor's position in a global coordinate system, the 3D coordinates of the measured location can be computed, allowing the creation of a 3D point cloud [Kaniouras et al., 2019]. Alternatively, the sensor's position could serve as the start of measurements in a local coordinate system.

$$d = \frac{c \times ToF}{2} \quad \text{where } c = 3 \times 10^8 \text{ m/s} \quad (2.6)$$

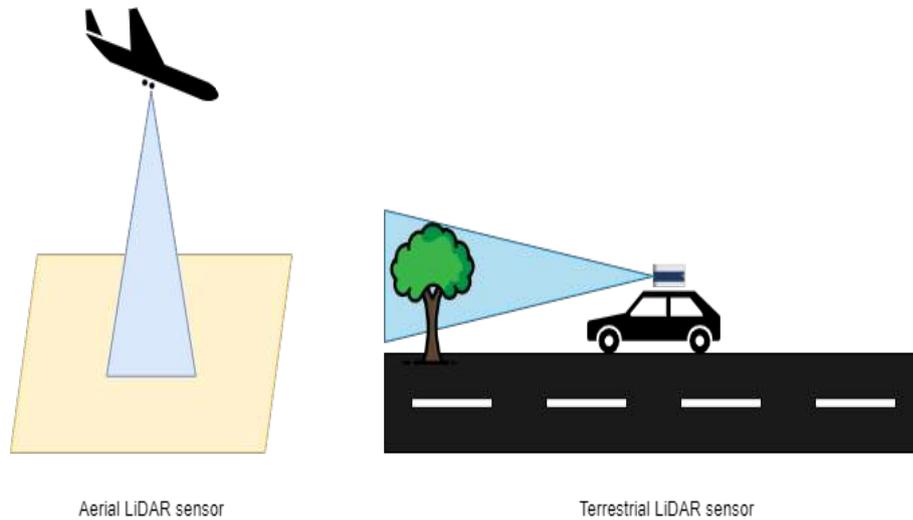


Figure 2.7.: Aerial and Terrestrial **LiDAR** sensors

During the last decades, various techniques have been developed, to address problems in different fields, such as autonomous driving systems, remote sensing, as well as navigation and mapping. **LiDAR** can be applied using various terrestrial sensors that can be static (fixed location) and mobile (mounted on vehicles) and airborne sensors that can be attached in a Unmanned Aerial Vehicle (UAV) or an airplane. The ability to use **LiDAR** in different applications, from construction projects to infrastructure monitoring, makes it an attractive technology to a wide community of professionals [Wang and Menenti, 2021]. **LiDAR** can also be combined with other sensors' measurements to resolve more complex problems.

### 2.6.2. Point cloud registration

A significant process concerning point clouds is point cloud registration, which includes finding a spatial transformation in order to align two point clouds. Different techniques exist that could be divided into global and local.

### Global registration techniques

A global registration technique, is needed when the compared point clouds, do not have a good initial alignment. In global techniques, the tested point clouds are usually first downsampled, to reduce time complexity, while their normal vectors are calculated. Afterwards, points with a unique and descriptive neighbourhood are selected, as well as a descriptor that can be a histogram, a value or a multi-dimensional vector, in order to describe the geometry of these key features. Currently, various descriptors have been developed, such as Point Feature Histogram (PFH) described by Rusu, as a descriptor, that uses the estimated surface normal vectors of all the points in the desired neighbourhood and assembles all the points' relationships in a histogram. FPFH proposed by [Rusu et al., 2009], is a 33-dimensional vector, that describes the local geometric property of a point, and is used by a nearest neighbor query to find the adjacent points with similar geometry. It can be used in real-time or near real-time applications. Additionally, RANSAC can be used to reject some correspondences, that can be considered as outliers and therefore refine the registration.

Moreover, some variations of the feature matching descriptors SIFT and SURF have been created. PointSIFT, created by Jiang et al. works by using a given an  $n \times d$  matrix that describes a point and its neighbourhood  $n$  with dimension  $d$ , and adds a new dimension to every point. Additionally, it is invariant to scale and models different orientations. A research implemented by Tong and Xiang discusses a methodology, where texture information is projected into two dimensional space, matched point pairs are extracted with the SURF operator and finally, curvature information is used to filter out points with low similarity. An extended analysis of the different local and global descriptors is given by Han et al. Zhou et al. implemented a global registration method that shows promising results concerning time complexity and can even be related with the efficiency of local registration algorithms.

### Local registration techniques

In cases where a rough initial alignment exists, local point cloud registration algorithms can be applied, in order to improve the results of global registration. The most common method is ICP, that works by keeping a target point cloud stable, while the source point cloud moves towards it, by comparing the distance between the coordinates of the matched points of each point cloud. Different variations of ICP have been developed, such as Point to Plane ICP, by Chen and Medioni, Point to Point ICP, by Besl and McKay, Generalised ICP produced by Segal et al. and comprises a combination of the two aforementioned algorithms and last but not least, Colored ICP by Park et al..

Point to Point ICP finds correspondences between the compared point clouds based on their transformation matrix. The transformation matrix is iteratively updated in order to minimise the differences between the point clouds.

$$E(T) = \sum_{(p,q) \in K} \| p - Tq \|^2 \quad (2.7)$$

Point-to-plane ICP works in a similar way, however it uses a different function to minimise the differences between the point clouds.

## 2. Theoretical background and related work

$$E(T) = \sum_{(p,q) \in K} ((p - Tq) \times n_p)^2, \quad \text{where } n_p : \text{normal of a point } p \quad (2.8)$$

Colored ICP combines point geometry with the photometric properties of the points. Park et al. additionally developed a multi-scale registration scheme, by iteratively registering point clouds with different voxel radius and iteration values to improve efficiency. Colored ICP consists of two equations, one revealing the geometry and the other the photometric aspects of the point cloud. The final equation of E is:

$$E(\mathbf{T}) = (1 - \delta)E_C(\mathbf{T}) + \delta E_G(\mathbf{T}) \quad (2.9)$$

For all the aforementioned ICP techniques the quality of the registration between two point clouds is checked based on the fitness value, which measures the overlapping area between the compared point clouds and specifically the number of inlier correspondences towards the total number of points of the target point cloud. Additionally the RMSE of the inlier correspondences of the point clouds can be calculated and its value has to be as low as possible, compared to the fitness value which requires to be high.

### 2.7. Privacy and location tracking

Privacy is a really important and up-to-date matter in the 21st century and can be directly linked to location tracking, a topic that is a significant part of this thesis. Currently, different legislative directives exist, such as the General Data Protection Regulation (GDPR), an important component of EU privacy and human rights law, that aims to enhance the control and rights of individuals over their own data. Moreover, the requirements and limits of personal data and their transfer within and outside EU are topics that are addressed in this regulation.

Indoor positioning and localisation services are being used by many people, that are not necessarily aware of that. For instance, Google Maps does not require to be open, as it works in the background, while it also incorporates indoor capabilities for some buildings. Indoor services may continuously track the location of an individual, therefore location tracking can be considered unethical if it takes place without the user's consent [Konstantinidis et al., 2015]. Location Based Services (LBS) and Inertial Positioning Systems (IPS) usually belong to the private sector, where companies may collaborate with third parties by distributing user data. Additionally, LBS and IPS can be infiltrated by hackers that could attempt to steal data of individuals. For instance, in case of automated vehicles, there is a very thin line between wanted location awareness by a service provider and unwanted location awareness by unwelcome persons doing surveillance, as well as hackers [Kaplan and Hegarty, 2017]. Therefore, the optimal goal of privacy preserving techniques is to manipulate data in a way that private information cannot be tracked back to specific individuals, protecting their identity.

Currently, various privacy preserving techniques are being used specifically for location services and the main categories are the following:

## 2.8. Summary of related work

Sanitised locations: This technique involves generating a set of fake locations, therefore sanitised, in order to protect the real location of the user [Kido et al., 2005; Yiu et al., 2008].

Spatial cloaking: Before users submit their position, so that it can be used in a LBS, a spatial cloaking area is calculated, by blurring the users' exact position and protect their privacy [Chow et al., 2006].

Space transformations: In social networks, users tend to upload location based content. Therefore, with this method, locations are redistributed in space, using cryptographic techniques. The keys can only be shared by users, only with their approval, protecting their data from unwanted attacks [Yiu et al., 2009].

k-anonymity: This is the state-of-the art of privacy preserving techniques. It encapsulates masking of the private id of different users, by making them indistinguishable, at least with a probability less than  $1/k$  [Sweeney, 2002].

Several research has been implemented regarding indoor location tracking. Zàruba et al. uses a system for locating wireless nodes in an indoor environment, requiring only one access point. The methodology describes the use of RSSI provided by Wi-Fi or other sensors that may exist in a home environment. Location is computed with Bayesian filtering on data that is derived by Monte Carlo sampling. User tracking is the outcome of combining different locations of a user. Furthermore, in the research of Kim et al., the constant location of elderly people can be known, based on an indoor system that uses RSSI measurements. Their location is analysed in combination with the time of the day, as well as the amount of time they spent in a specific location, revealing movement patterns.

## 2.8. Summary of related work

This chapter includes both the theoretical background and the related work that is the basis of this thesis. Data acquisition in this thesis is based on LiDAR and camera sensors that work based on the ToF principle. Indoor localisation will be implemented based on different point cloud registration and feature matching techniques concerning images, with a variety of them being explained in this chapter. Specifically, point cloud registration will involve both global and local techniques, such as ICP and its variations. Similarly, this thesis will attempt to complement to the research of Fratzeskou et al. that uses ceiling data for indoor localisation. Additionally, this thesis will delve into the implementation and comparison of different feature key-points, descriptors and feature matching techniques that were previously described. Location tracking of different users, based on the indoor localisation results is implemented, while protecting their privacy using a variation of the k-anonymity privacy technique, aiming to discover movement patterns in an indoor facility. Last but not least, a dashboard that includes statistics based on the location tracking results is implemented, so that movement patterns of users can be discovered.



# 3. Methodology

This chapter includes an overview of the methodology and specifically the exact steps that were implemented for the completion of this thesis. This chapter has the aim of providing the reader with a conceptual aspect of the developed methodology and the design of the experiments that were implemented to validate this procedure.

## 3.1. Overview

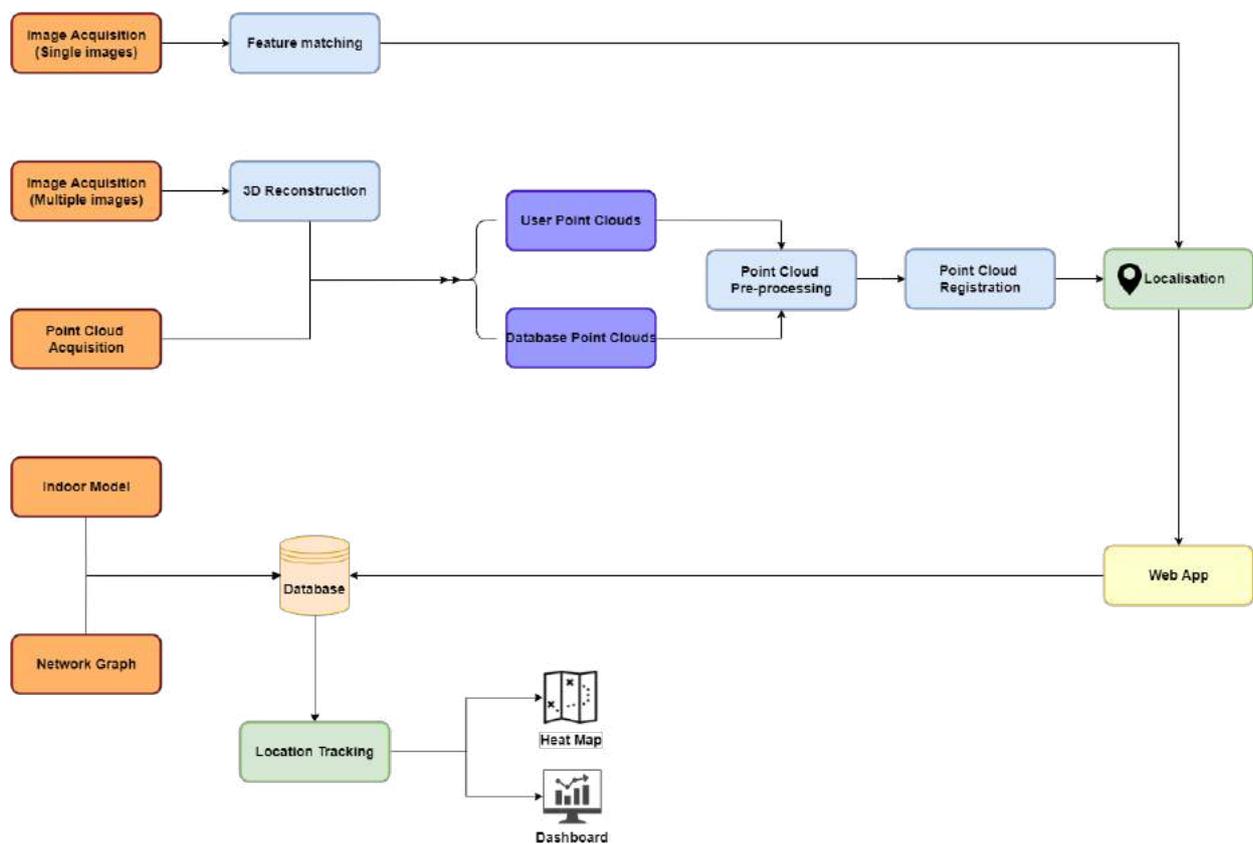


Figure 3.1.: Methodology overview

Based on the methodology presented in the figure above, the main steps of the pipeline will be briefly explained. First, the ceiling data acquisition, was achieved based on three different techniques. Single images of the tested rooms were acquired with camera sensors from different

### 3. Methodology

mobile devices, as well as overlapping images of the same sites, to achieve 3D reconstruction of these scenes. Additionally, point clouds were acquired from LiDAR sensors of an iPad, that include this sensor. For both the point clouds and the images, indoor localisation was achieved by comparing user data and reference data, that were uploaded in a database. Regarding single images, their features were matched based on different matching techniques. Multiple overlapping images of a ceiling were first reconstructed in three dimensions, both from user and database side. As a result, point cloud registration techniques were used to compare the two types of point clouds. Furthermore, directly acquired point clouds, were first pre-processed and then co-registered to achieve indoor localisation. Indoor localisation results derived from point cloud comparison were stored in a database and were visualised in a web application. The indoor model of the tested indoor facility and its network graph were combined with the localisation results to provide information on users' current and previous locations. This way, the used paths were revealed and consequently the movement patterns in an indoor facility. The visualisation of this location tracking operation has the form of a heat-map, including user paths during different times of a day. Additionally, a dashboard included statistics about the path usage was created.

## 3.2. Data acquisition

### 3.2.1. Point clouds

Regarding point cloud acquisition, two types of point clouds were acquired, from LiDAR sensors. There are point clouds that act as reference and were stored in a database, as well as point clouds that are acquired by a user. The latter will be compared to these reference point clouds, so that indoor localisation is achieved, based on the best match.



Figure 3.2.: LiDAR sensor of iPhone 12 pro

The acquisition involves the use of different applications and devices that will be thoroughly discussed in the [Chapter 4](#). Parameters, such as the measuring angle, height of the device, as well as the part of the room that is acquired are significant in order to reveal how they affect the indoor localisation results. The point cloud acquisition was implemented in two ways: while a person is walking into a room, giving a dynamic perspective to the acquisition and also while staying still, so that it is investigated if the final product of the thesis can be used during emergency situations, in cases where an individual might be unable to move.

### 3.2.2. Images

Single images of the tested rooms were acquired from camera sensors of a mobile device. As in the case of point clouds, some images were used as reference to represent the room's ceiling in two dimensions. Images of a ceiling acquired by a user were then compared to the reference images of the rooms in order to reveal the user's location based on the optimal match. Furthermore, overlapping images of ceilings were acquired with the same sensors and a minimum overlap, for the reconstruction of the 3D scene, based on the procedure described in [Chapter 2](#). In both cases, the testing includes image acquisition from different camera sensors and viewpoints, in order to examine how these parameters influence the indoor localisation result. Additionally, concerning the overlapping images, different combinations of feature detectors, descriptors and feature matching techniques that were discussed in [Chapter 2](#) were tested, in an effort to discover their optimal combination, for indoor localisation based on images of the ceilings.

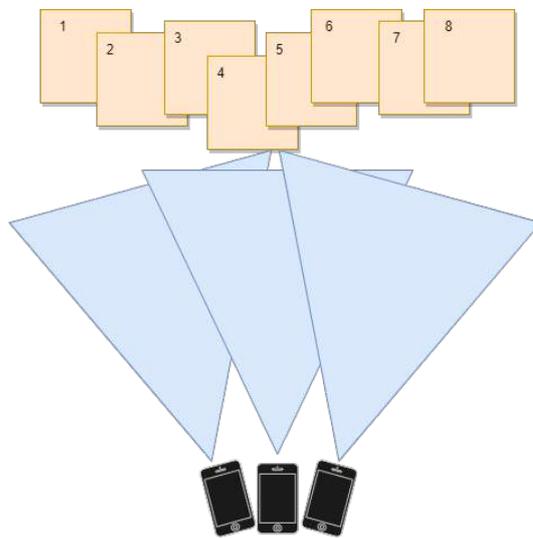


Figure 3.3.: Acquisition of overlapping images

## 3.3. Point cloud pre-processing

The comparison between these two types of point clouds was facilitated by performing some primary operations. Pre-processing of the point clouds included voxel down-sampling, an operation where a regular voxel grid was used, to create a uniform down-sampled point cloud, from an input point cloud. The algorithm was firstly performed by transforming points into voxels and secondly by generating the centroid of each voxel, which is its representing point. This operation is useful, as it aims to reduce processing time by manipulating a point cloud of smaller size [[Miknis et al., 2016](#)]. However, this operation has to be implemented carefully and until a certain threshold, because further down-sampling might result to important loss of information. Furthermore, when acquiring ceiling data, it is possible that the point cloud includes adjacent wall parts, that need to be excluded from the upcoming operations. These parts

### 3. Methodology

can be considered as outliers [Han et al., 2017] for the purposes of this research. To achieve their removal, a smaller part of the acquired point cloud was used, in an effort to discard the wall parts that might exist in the corners of the point clouds. Additionally, some outliers were located and removed based on the number of their neighbours, to further improve the point cloud's quality and reduce processing time. Last but not least, plane segmentation based on the RANSAC algorithm was performed, in order to differentiate the flat surface of the ceiling with its protruding objects, such as lamps and other installations, which comprise the characteristic details of each room's ceiling.

---

**Algorithm 3.1:** Point cloud preprocessing

---

**Input:** Point cloud  $pc$  and voxel size

**Output:** Original point cloud, Cleaned point cloud and FPFH features

```
1  $pc \leftarrow$  bounding box of x,y,z coordinates
2 bounding box  $x_{min} = \max(points_x) - 0.1 * (\max(x) - \min(points_x))$ 
3 bounding box  $x_{max} = \min(points_x) + 0.1 * (\max(x) - \min(points_x))$ 
4 bounding box  $y_{min} = \max(points_y) - 0.2 * (\max(y) - \min(points_y))$ 
5 bounding box  $y_{max} = \min(points_y) + 0.2 * (\max(y) - \min(points_y))$ 
6 bounding box  $z_{min} = \max(points_z) - 0.1 * (\max(z) - \min(points_z))$ 
7 bounding box  $z_{max} = \min(points_z) + 0.1 * (\max(z) - \min(points_z))$ 
8  $pc \rightarrow$  downsample based on voxel size
9  $pc \rightarrow$  remove outliers(nearest neighbours, max standard deviation)
10  $pc \rightarrow$  segment plane (distance threshold, ransac points , iterations)
11 for points in outlier point cloud do
12    $\lfloor$  CalculateFPFHfeatures(radius,nearest neighbours)
13 return  $pc$ , outlier cloud and FPFH
```

---

## 3.4. Point cloud registration

After acquiring and pre-processing reference and user point clouds, the next step was to create an algorithm that would aid towards comparing them. The main idea behind this is, that each point cloud taken by a user, would be compared with all the point clouds in the database and the best match will reveal the room where the user is located. This procedure works as follows for both types of point clouds. The comparison first included a global registration, so that the user and the reference point clouds obtain an initial alignment and afterwards a local registration algorithm to refine the point cloud registration.

### 3.4.1. Global registration

First, the normal vectors of all the points were computed. Furthermore, points with a unique and descriptive neighbourhood were detected. The detection and description of these unique points for each point cloud was implemented based on FPFH features that were described in Chapter 2. Two different global registration techniques were implemented and compared.

The first technique includes the aforementioned steps and then RANSAC, in order to select some random points from the reference point cloud and then find the corresponding points in

the user point cloud, using a nearest neighbor query in the 33-dimensional FPFH feature space [Li et al., 2021]. Aside from the distance of the corresponding points in the compared point clouds, the similarity between two edges between the compared point clouds and the vertex normal affinity of the correspondences are also checked. In case the points satisfy the selected thresholds, the transformation of the user point clouds towards the reference point clouds is implemented.

The second technique was implemented, based on the fast global registration proposed by Zhou et al. This method follows the steps of the global registration described above, however it does not use RANSAC to choose point correspondences between the two point clouds, but finds the nearest neighbour of every point in the user point cloud among the reference point cloud, based on distance analysis in feature space. This implementation does not require an additional local refinement, however in this thesis it was combined with ICP local refinement methods.

### 3.4.2. Local refinement

Based on the results of the global registration, an attempt of improving the quality and time efficiency of the algorithm includes different variations of the ICP algorithm. The further minimisation of the point cloud differences was performed by keeping one point cloud fixed, while the other is transformed towards it. Specifically, each point of the user point cloud was matched to the closest point of each reference point cloud. Then, rotation and translation were estimated and this process is iterated until the results converge [Li et al., 2021]. The user point cloud was compared to all the reference point clouds, based on the fitness and the RMSE value, which will result in the indoor localisation. Different variations of ICP were implemented and compared and more specifically Generalised, Point-to-Plane, Point-to-Point and Colored ICP. These steps were implemented both to directly acquired point clouds, as well as point clouds that were reconstructed from overlapping images of a ceiling.

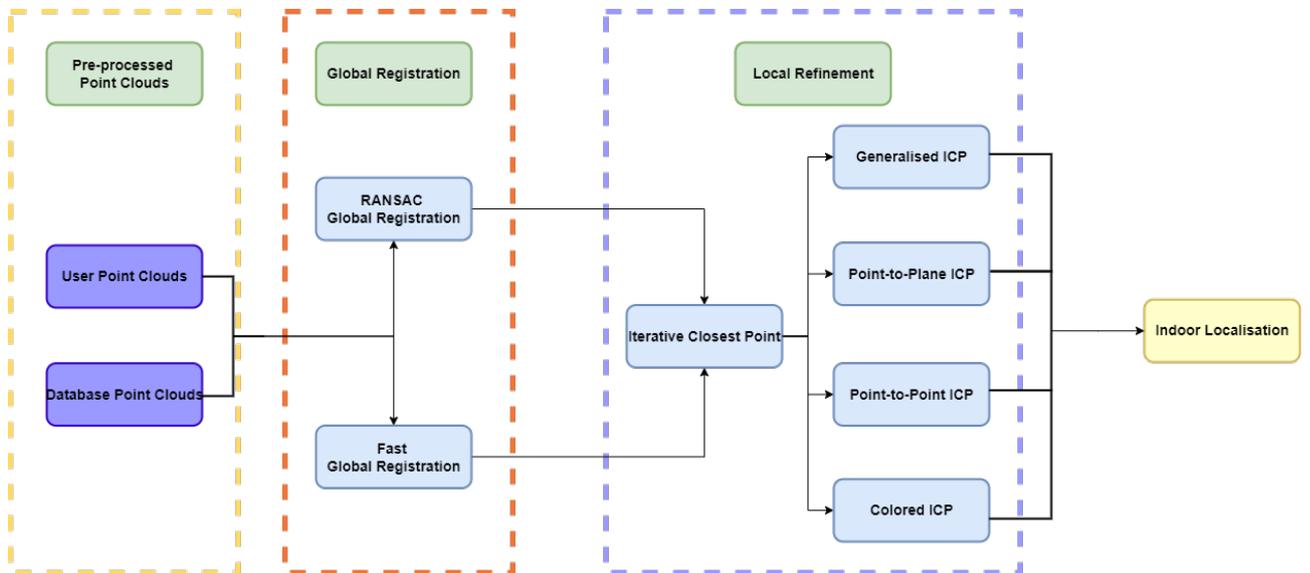


Figure 3.4.: Indoor localisation based on point clouds

### 3. Methodology

## 3.5. Feature matching

In this section feature matching based on the comparison of single images, as well as comparison between overlapping images of a set is explained, in order to answer the research question regarding the optimal combination of feature detectors, descriptors and matching techniques.

### 3.5.1. Feature matching between single images

For each of the selected rooms, one image of a ceiling was acquired and acted as reference. For testing purposes, different user images were additionally acquired from different viewpoints and were compared with the reference images. This comparison included the use of different feature descriptors and detectors, such as [ORB](#), [SIFT](#), and also two different feature matching techniques, brute-force and [FLANN](#). The number of matches between the user and the reference images was used to reveal the location of the user.

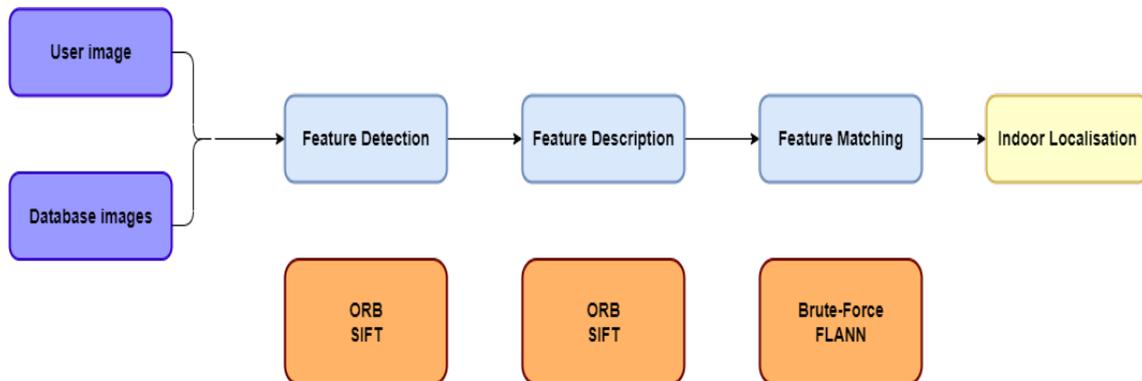


Figure 3.5.: Indoor localisation based on feature matching

### 3.5.2. Feature matching between overlapping images

An important step towards 3D reconstruction is the feature matching between the overlapping images. Some subsets of images were selected, in order to evaluate different combination of feature detectors, descriptors and matching techniques from acquired images of ceilings. Additionally, the images were acquired based on different overlaps, in an attempt to examine how the percentage of coverage affects the time complexity of the 3D reconstruction and the quality of the results. The quality of the results was examined based on the number of matches between the overlapping images, as well as a ground truth which was manually set for several images, so that the results are protected against false matches. Graphical representations that show the true to false matches were created to enhance the results. Following these steps, the overlapping images sets were used for 3D reconstructions of the ceilings.

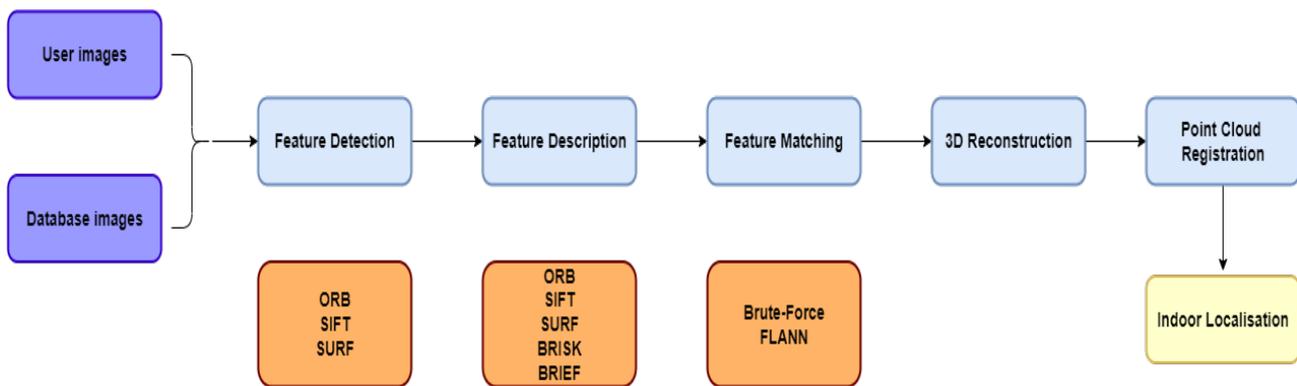


Figure 3.6.: Indoor localisation based on point clouds that were reconstructed from overlapping images

### 3.6. Storage

The setup of the whole system was organised in an online database. This database includes the indoor model of the case study and a network graph, that connects all the rooms of the tested area to each other. Except for the geometry of the rooms in the indoor model, each of them includes one pre-processed point cloud and an image that acts as reference for the point cloud registration and feature matching operations respectively. Moreover, this indoor model serves as an embedded map in web-application that was created, allowing the users to have a visual insight of their location.

### 3.7. Web-app

The indoor localisation results were visualised in a web application that has the form of a minimum viable product. The app works by requesting the reference point clouds from the database, so that they can be compared based on the discussed algorithms to the user data in near-real time. Users are able to post their data in the application and after a few seconds the room they are located in is revealed. Additionally, the app includes the indoor model of the case study, so that aside from the name of the room, the app also highlights the polygon that represents the room in the indoor model of the indoor facility and zooms in it.

### 3.8. Location tracking

Each time the web application is used, the users' current and previous locations are stored in the online database under an encrypted id. When users move between different rooms, it means that they used a certain path to achieve that. Based on the network graph of the indoor space that reveals all the connections between adjacent rooms, the current and previous locations of the users were translated to a line in the network graph, representing a specific route. The availability of this information is near-real time as the results appear in the online server after

### 3. Methodology

a few seconds. Based on the unique id of each user, a heat map that is based on the network graph was used to visualise the used routes.

Additionally, this information was used to reveal different movement patterns, during different times of a day. The visualisation is accomplished in the form of a heat-map, where based on the usage of each path, different colors and width were applied to the corresponding line of the network graph. Consequently, this information can reveal how much a path is used during a daily, weekly or even monthly time span. Acquiring this knowledge is valuable, especially during the COVID-19 era, because it can be exploited by a building manager, who can achieve the optimal distribution of people in an indoor facility [Spinoza Andreo et al., 2021].

---

**Algorithm 3.2:** Location tracking

---

**Input:** Reference point cloud  $ref_{pc}$ , User point cloud  $user_{pc}$  and table with paths

**Output:** List containing visited rooms

```
1 if length(localisation list)==0 then
2   for  $ref_{pc}$  in roomlist do
3     preprocess  $user_{pc}$  and  $ref_{pc}$ 
4     global registration( $user_{pc}$ ,  $ref_{pc}$ )
5     local registration( $user_{pc}$ ,  $ref_{pc}$ , resultglobal) → calculate fitness value
6     return room(max fitness)
7 else if length(localisation list) ≥ 0 then
8   for  $i$  in range(length(result)) do
9     request path table
10    paths → query(previous room, current room)
11    query result → update path usage
12    preprocess  $user_{pc}$  and  $ref_{pc}$ 
13    global registration( $user_{pc}$ ,  $ref_{pc}$ )
14    local registration( $user_{pc}$ ,  $ref_{pc}$ , resultglobal) → calculate fitness value
15    return room(max fitness)
16
```

---

### 3.9. Case study

The case study of this thesis will involve the Faculty of Architecture and the Built Environment of TU Delft. Specifically, the case study area includes a few rooms of the ground floor of this facility, which are selected in order to test the created pipeline, including both point clouds and images. The selected rooms are visualised with a pink color in [Figure 3.7](#).

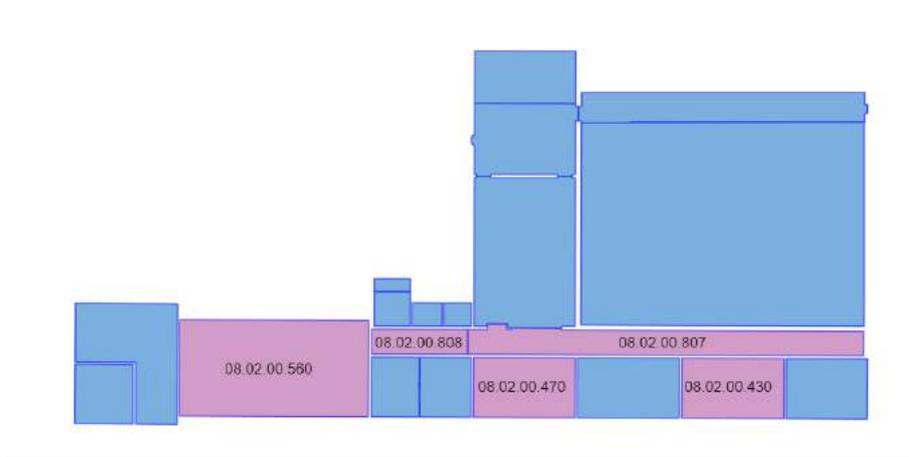


Figure 3.7.: Selected rooms from the Faculty of Architecture and the Built environment



## 4. Implementation and experiments

This chapter aims to explain in detail the steps of the methodology from a technical perspective. The tools that were used, as well as the datasets are firstly explained. Afterwards, details on the implementation steps, such as parameters and how they were chosen, as well as some intermediate results are presented in this chapter.

### 4.1. Tools

In this section first the software that was used for the implementation of this thesis is analysed. In the meantime, the hardware tools, such as the specifications of the devices that were used for data acquisition and the personal computer that was used are also mentioned.

#### 4.1.1. Software

##### Python

The implementation of the methodology of this thesis was completed using the Python programming language. Python was used solely for back-end development, meaning the implementation of the developed algorithms and not the creation of the web-application. The version that was used is 3.8.10, due to the fact that it is one of the newest versions currently available and also due to some incompatibilities between the different libraries, that were faced in other versions that were attempted. At the same time, python code was executed in the Visual Studio Code editor, which has the capability to understand different programming languages and has an user-friendly interface.

arcgis.features: Module of the `arcgis.gis` library that is used to connect maps and location services with an exterior application. In this thesis it is used in order to request network and map layers from the ArcGIS online portal.

open3d: Open source library for 3D data processing. Its python version is used in this thesis in order to read, analyse and visualise different point cloud data.

Flask: Lightweight python web framework, that can be used to create web-applications, by connecting the python back-end to a front-end developed with HyperText Markup Language (HTML) or JavaScript.

#### 4. Implementation and experiments

uuid: Python library that is used to create a unique id, based on the Universal Unique Identifier (UUID) standards that is used to uniquely identify an object or an entity in the internet. Specifically, it contains a string of either 32 or 16 hexadecimal digits. This library is used in order to create a unique identifier for the users of the web-application and protect their personal information, while location-tracking operations are executed.

matplotlib: Python library that is used to create static and interactive visualisations. This library is used in order to visualise images, as well as their keypoints and matches for feature matching operations.

openCV-Python: Open-source computer vision and machine learning library that focuses on real-time applications. Its python binding is used in the current thesis in order to implement different image matching techniques, while using feature detectors and descriptors included in this library.

#### **HTML and Cascading Style Sheets (CSS)**

[HTML](#) is the standard markup languages for web pages, and it helps towards constructing their main interface. Additionally, it is a standard markup language, which means that their content cannot be altered. [CSS](#) is used in order to style an HTML document and describes how each [HTML](#) element is visualised.

#### **JavaScript**

JavaScript is a high-level scripting language, that was introduced for client-side operations in a web browser. Compared to [HTML](#), it is a dynamic language, that also improves the website appearance. In this thesis JavaScript code was embedded inside an [HTML](#) script.

#### **ArcGIS pro**

ArcGIS Pro is the latest commercial desktop Geographic Information System (GIS) application from ESRI. The software incorporates several capabilities including exploration, visualisation and analysis of different data, in order to create different 2D maps and 3D scenes. Using the ArcGIS portal, the work that is implemented in a local computer can be shared online. Additionally, for this project ArcGIS Indoors is a vital extension, which can be used to translate [CAD](#) or [BIM](#) data into floor-aware indoor maps, in order to support facility, space management as well as other operations such as finding the fastest route to a specific location.

#### **CloudCompare**

CloudCompare is an open-source 3D point cloud and mesh processing software. It includes several capabilities, such as point cloud processing and more advanced operations such as point cloud registration and automatic segmentation. For this thesis, CloudCompare was used for visualising some primary results and inspect the acquisition quality.

### 4.1.2. PhotoMatch

PhotoMatch is an open-source software that delves into automatic feature matching, enabling the implementation and evaluation of different feature detection, description and matching techniques. Aside from these, it also includes image enhancement methods to improve the quality of an image dataset [González-Aguilera et al., 2020]. This software was used in this thesis, to investigate the optimal algorithm combination of detectors, descriptors and feature matching techniques, for indoor localisation based on images of ceilings.

### 4.1.3. COLMAP

COLMAP is a Structure from Motion (SfM) and Multi-View Stereo (MVS) pipeline, used through a graphical or command-line interface. Its main purpose is 3D reconstruction from image collections, which is how it was used in this thesis [Schönberger et al., 2016; Schönberger and Frahm, 2016].

### 4.1.4. Hardware

#### Computer

The laptop that was used for the implementation of the current thesis has the following specifications:

- Model: HP OMEN 15-en0135nd - Gaming Laptop - 15.6 Inch
- Operating system: Windows 11
- Processor: AMD Ryzen 7 4800H (8 cores)
- Clock speed: 2.9GHz (max. 4.2GHz)
- Internal memory(RAM): 16 GB
- System: 64-bit operating PC

#### Xiaomi Redmi Note 9s

Two cameras of this Android device were used for image acquisition, in order to test different feature matching algorithms towards indoor localisation. Specifically, an ultra wide camera with 8 MP resolution,  $f/2.2$  aperture and 119 degrees field of view, and a camera with 5 MP resolution and  $f/2.4$  aperture were used for image acquisition.

## 4. Implementation and experiments

### iPad

For point cloud and image measurements, the respective [LiDAR](#) and camera sensors of an iPad pro 12 device were used. The device includes one wide camera with 12MP and  $f/1.8$  aperture and an ultra wide camera with 10 MP,  $f/2.4$  aperture and 125 degrees field of view.

The technical specifications of the [LiDAR](#) sensor are not available online, which makes users unaware of the real applicability of their devices and consequently the level of accuracy and precision [[Díaz Vilariño et al., 2022](#)]. However their maximum range is approximately 5 meters. For data acquisition with the aforementioned sensors two applications were used and compared:

- **SiteScape:** Application available on the Apple Store, that follows a freemium model, meaning that many of its basic capabilities are provided for free. A user-friendly and accessible application, that can help acquire point clouds and export them in Polygon File Format ([PLY](#)) format. The application uses the [LiDAR](#) sensor to acquire the coordinates of the points in space, combined with the colored RGB image from the wide-angle camera of the iPad. Additionally, some tools in ARKit are used, such as plane detection, in order to place the objects in the real world. Based on the documentation, measurements of this software have an accuracy of  $\pm 1$  cm on average.
- **Pix4d Catch:** This application is available on the Apple store, however compared to SiteScape, a 15-day trial is required, so that it can be freely used. It contains more capabilities compared to SiteScape, as it can reveal the different locations of the device during the acquisition. The downsampling scale of the acquired point cloud can also be chosen. An additional aspect is the capability of automatically acquiring multiple overlapping images of the measured area, while a user is using the [LiDAR](#) sensor to acquire a point cloud. Similarly to SiteScape, the depth information is captured by the [LiDAR](#) sensor, while the texture and color of the points are based on the wide-angle camera of the device.

## 4.2. Datasets

The point cloud datasets were acquired using the iPad 12 pro devices and especially their [LiDAR](#) sensor. Two different applications were used to exploit the [LiDAR](#) capabilities and specifically SiteScape and Pix4D Catch. The acquisition on both these applications includes exporting the point cloud into [PLY](#) format. Point clouds were acquired with two different time-spans of 20 seconds while a user is moving inside a room, and 30 seconds while a user remains still. These experiments were implemented in order to investigate the chance of using point clouds of the ceilings for dynamic applications, such as a robotic applications that constantly map an area, as well as static applications, where users need to figure their location in a case of an emergency, such as a fire and might be unable of moving in a room. The point clouds from SiteScape were created with average quality, while the ones from Pix4D Catch with a lower quality, in order to investigate how the results are affected by quality, as well as the time efficiency of the algorithm and its suitability for real-time applications.

| Room         | Original Point Cloud (Dynamic) | Original Point Cloud (Static) | Original Point Cloud (Database) |
|--------------|--------------------------------|-------------------------------|---------------------------------|
| 08.02.00.430 | 558209                         | 472426                        | 1503456                         |
| 08.02.00.470 | 536849                         | 556006                        | 1385289                         |
| 08.02.00.560 | 548031                         | 365033                        | 1064175                         |
| 08.02.00.807 | 473777                         | 184747                        | 844399                          |
| 08.02.00.808 | 195576                         | 211368                        | 474517                          |

Table 4.1.: Number of points per room, acquired with SiteScape

| Room         | Original Point Cloud (User) | Original Point Cloud (Database) |
|--------------|-----------------------------|---------------------------------|
| 08.02.00.430 | 52586                       | 118690                          |
| 08.02.00.470 | 73925                       | 86135                           |
| 08.02.00.560 | 36122                       | 40438                           |
| 08.02.00.807 | 168338                      | 191406                          |
| 08.02.00.808 | 39179                       | 47043                           |

Table 4.2.: Number of points per room, acquired with Pix4D Catch

Tables [Table 4.1](#) and [Table 4.2](#) reveal, that the size of the point clouds that were acquired from Pix4D Catch is significantly smaller than the ones acquired by SiteScape, a factor that might significantly affect the time efficiency of the implemented algorithms.

Image acquisition is implemented with different cameras, such as the iPad pro wide cameras and two cameras of a Xiaomi Redmi Note 9s. The acquisition is implemented from different perspectives, to examine how the image comparison is affected. Additionally, Pix4D Catch was used for acquiring overlapping images of the ceilings, in an automated way, as this operation takes place in parallel to the point cloud acquisition, using the ultra-wide camera of the iPad.

| Room         | Number of overlapping images (User) | Number of overlapping images (Database) |
|--------------|-------------------------------------|---|
| 08.02.00.430 | 87                                  | 161                                     |
| 08.02.00.470 | 129                                 | 140                                     |
| 08.02.00.560 | 43                                  | 47                                      |
| 08.02.00.807 | 232                                 | 242                                     |
| 08.02.00.808 | 43                                  | 47                                      |

Table 4.3.: Number of overlapping images per room

CAD drawings of the Faculty of Architecture were used. Specifically, the CAD drawings of the ground floor of the Faculty of Architecture and the Built environment are used, as an input to ArcGIS Pro, so that the indoor model of the selected part of the Faculty and a network graph are created. The CAD files were provided for the project of [Spinoza Andreo et al.](#) from TU Delft Real Estate.

#### 4. Implementation and experiments



Figure 4.1.: CAD drawing of the Faculty of Architecture and the Built environment (West side - ground floor)

### 4.3. Point cloud pre-processing

Point-cloud pre-processing involves different operations that were performed using python's library open3d. First of all, voxel-downsampling was implemented with this library, by experimenting through different voxel sizes varying from 0.05 to 0.3 in order to figure the optimal size that combines quality and processing time, but at the same time does not omit important details of the ceilings. Additionally, some adjacent wall parts were excluded by keeping a smaller slice of the point cloud, based on algorithm 3.1. Some outliers were statistically removed with the respective open3d module, taking into account 30 neighbours to calculate the average distance of a given point, while the standard deviation ratio was set to 1, a threshold that is based on the standard deviation of the average distances in the point cloud. Furthermore, plane segmentation based on RANSAC was performed in order to separate the upper flat part of a ceiling and the rest of the points that represent the characteristic details. Using the segment plane module of open3d, distance threshold for a point to be considered as an inlier was set to 0.4, number of randomly sampled points to 3 and number of iterations to 1000. The latter indicates how often a random plane is sampled and verified, affecting the processing time as the number becomes higher. The aforementioned numbers were chosen after tweaking the parameters and checking how the results were affected. It has to be noted that the distance threshold may vary per dataset. The pre-processing results can be seen below:



Figure 4.2.: Pre-processing steps

Tables 4.4 and 4.5 show that the chain of pre-processing operation significantly reduces the sizes of the point clouds, which is an important aspect for the time complexity of the pipeline, especially since it involves real-time visualisation of the results. The smaller size of the point

#### 4.4. Point cloud registration

clouds that were acquired with Pix4d Catch, is a result of the app, which allows the user to acquire point clouds with a downsampling factor.

| Room         | Dynamic case         |                           | Static case          |                           |
|--------------|----------------------|---------------------------|----------------------|---------------------------|
|              | Number of points     |                           |                      |                           |
|              | Original Point Cloud | Pre-processed Point Cloud | Original Point Cloud | Pre-processed Point Cloud |
| 08.02.00.430 | 558209               | 3881                      | 472426               | 2860                      |
| 08.02.00.470 | 536849               | 6532                      | 556006               | 5646                      |
| 08.02.00.560 | 548031               | 4961                      | 365033               | 5172                      |
| 08.02.00.807 | 473777               | 5438                      | 184747               | 1866                      |
| 08.02.00.808 | 195576               | 1446                      | 211368               | 1241                      |

Table 4.4.: Number of points before and after pre-processing, acquired with SiteScope

| Room         | User                 |                           | Database             |                           |
|--------------|----------------------|---------------------------|----------------------|---------------------------|
|              | Original Point Cloud | Pre-processed Point Cloud | Original Point Cloud | Pre-processed Point Cloud |
| 08.02.00.430 | 118690               | 5038                      | 52586                | 2965                      |
| 08.02.00.470 | 86135                | 2157                      | 73925                | 4645                      |
| 08.02.00.560 | 40438                | 1057                      | 36122                | 1212                      |
| 08.02.00.807 | 191406               | 6147                      | 168338               | 6272                      |
| 08.02.00.808 | 39179                | 2324                      | 47043                | 2336                      |

Table 4.5.: Number of points before and after pre-processing, acquired with Pix4D Catch

## 4.4. Point cloud registration

Point cloud registration encompasses the comparison of a pre-processed user point cloud and all the point clouds of the database, in order to find the best match, which will lead to the indoor localisation of the user. Different techniques, were implemented and compared, so that the best combination of a global registration and local refinement is selected. Implementation details concerning the computation time are included in this chapter, while the quality of the results will be discussed in [Chapter 5](#).

### 4.4.1. Global registration

As it was discussed in [Chapter 3](#), two global registration methods were implemented: [RANSAC](#) based registration and fast global registration, implemented by [Zhou et al.](#). The latter can be used as a standalone registration method however in this thesis it was combined with a local registration, for direct comparison with the [RANSAC](#) method. [RANSAC](#) based registration involves tweaking of some pruning parameters, that aid towards rejecting some false correspondences. Specifically, the distance between the point clouds was checked, by setting a respective threshold (voxel size \* 3), as well as the similarity of two arbitrary edges in the reference and user point cloud. The distance of these edges in both user and reference point cloud were measured and then their similarity was checked according to [Equation 4.1](#) and [Equation 4.2](#). The convergence criteria of the [RANSAC](#) algorithm were set to 100000 iterations and the confidence

#### 4. Implementation and experiments

probability to 0.999. The aforementioned parameters were chosen, after trial and error, as they led to the most optimal results, based on time complexity and quality of the results. Considering the fast global registration, only a few parameters were required. The distance threshold was set the same as the [RANSAC](#) method for comparison purposes. Both of these methods were created as two separate functions in python, using the library `open3d`.

$$\|edge_{source}\| > 0.9 \cdot \|edge_{target}\| \quad (4.1)$$

$$\|edge_{target}\| > 0.9 \cdot \|edge_{source}\| \quad (4.2)$$

##### 4.4.2. Local refinement

The initial alignment of the [RANSAC](#) global registration was used in order to improve the point cloud matching and time efficiency, based on variations of the [ICP](#) algorithm. The user point cloud was transformed towards each one of the reference point clouds, in an attempt to find the optimal match and localise the user. Point-to-Point, Point-to-Plane and Generalised [ICP](#) were implemented as modules of the `open3d` library and applied with maximum convergence set to 30 iterations. Regarding the Color [ICP](#) registration, the implementation was based on [Park et al.](#) and was executed iteratively three different times with voxel sizes (0.1, 0.15, 0.2) and three iterations (15, 30, 50). For each of these combinations, the compared point clouds were downsampled, the points' normal vectors were estimated and finally the local refinement was achieved, taking into account both the point cloud geometry and the colors of the point clouds, depending on the algorithm.



Figure 4.3.: Global and local registration

Figure 4.4 shows the processing time of different combinations of global and local registration algorithms, based on point clouds acquired with SiteScape, Pix4D catch, as well as the ones that were reconstructed based on overlapping images. The smaller size of the Pix4D point clouds, significantly affects the processing time and results into indoor localisation in approximately 3-6

seconds depending on the algorithm. This time difference is small, therefore the use of different registration combinations does not significantly affect the time efficiency. However, while the size of the point clouds becomes higher, such as in the SiteScape dataset, the time complexity augments exponentially, and reveals the time efficiency of Colored ICP, which produces indoor localisation in approximately 18 seconds when combined with the two different global registration algorithms that were applied. In this case, there is significant difference in the processing time, as other algorithms such as Generalised ICP takes approximately 60 seconds to result into indoor localisation. This difference will be even greater, when a database with a higher number of point clouds is used to perform indoor localisation.

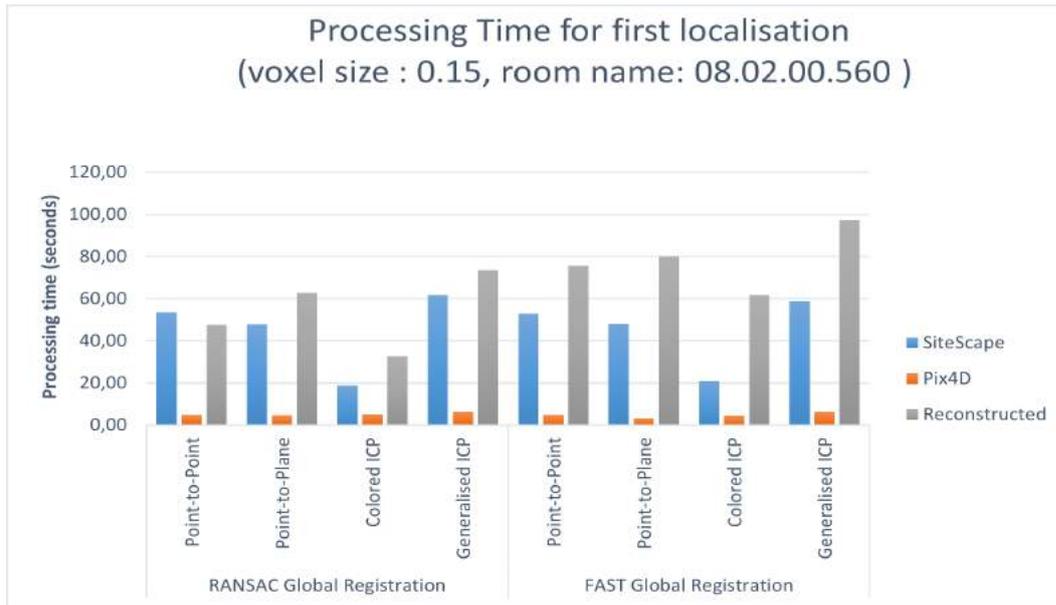


Figure 4.4.: Processing time of different registration algorithms

## 4.5. Feature matching

This section includes the implementation details for the techniques that were described in [Chapter 3](#) and are based on image acquisition.

### 4.5.1. Feature matching between single images

Images that were acquired from the camera of a mobile device, were first converted to gray-scale images. For each user and reference images, unique key-points were calculated, as well as descriptors in order to outline the key-points behavior. Their implementation is based on [ORB](#) and [SIFT](#), in an attempt to achieve indoor localisation based on single images of ceilings. Following, the detection and description of the unique points of each point cloud, brute-force and [FLANN](#) feature matching techniques were implemented, in order to find matches between the user and reference images. These algorithms were implemented based on the python binding

#### 4. Implementation and experiments

of the OpenCV library, while the visualisation of the key-points and the matches on matplotlib. For comparison purposes between the different techniques, the maximum threshold for detected features was set to 5000. A strict distance test was set to reject some false correspondences between the compared images.

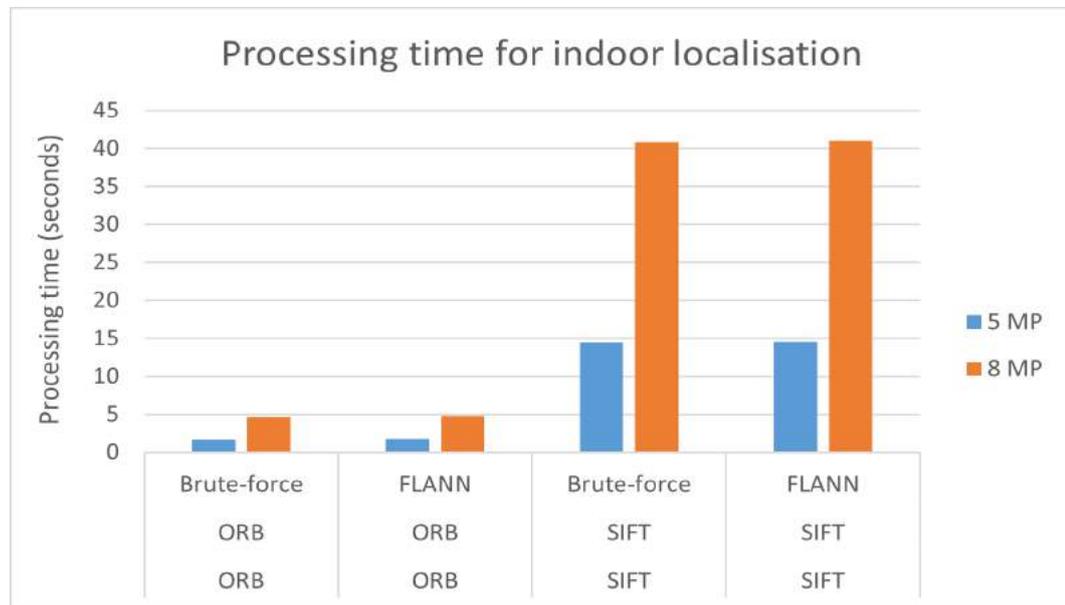


Figure 4.5.: Processing time of different feature matching combinations

Figure 4.5 shows the processing time while using different combinations of feature detection, description and matching algorithms with two different cameras of 5 and 8 MP resolution respectively. The resolution of the camera, affects the time efficiency of the calculation, as the algorithm is executed faster in every combination while the 5 MP camera is used. Additionally, ORB detector and descriptor is faster than acsift, showing the importance of SURF, which could not be applied in python, because its use is patented. Brute-force is faster than FLANN, however in bigger datasets the latter is highly efficient.

#### 4.5.2. Feature matching between overlapping images

Before reconstructing the 3D scene from overlapping images of ceilings, a small subset of ten images was chosen, in order to investigate the optimal combination of feature detectors, descriptors and matching techniques. The PhotoMatch software was used for this purpose. Specifically, the images were first decolorised. Afterwards, feature extraction and detection was applied, with various combinations of algorithms that are provided from this software. As in the case of single images, the maximum threshold for detected features was set to 5000. Two different feature matching techniques were implemented, brute-force and FLANN, which also passed a fundamental matrix geometry test to validate the results. The detected key-points and matches between the compared images, were visualised in the viewers provided by PhotoMatch. Additionally, the ground truth was manually set for the selected images, in order to detect false correspondences. ROC curves were used to visualise the true to false match ratio, based on

the ground truth. This operation is solely experimental and aims to evaluate the different descriptors, detectors and feature matching techniques, before point cloud reconstruction. The 3D reconstruction from overlapping images was completed in COLMAP, a software capable of reconstructing sparse and dense models, using SIFT to find correspondences from overlapping images. After the reconstruction was finalised, the user and reference point clouds were compared based on the point cloud algorithms that were described above with the open3d library.

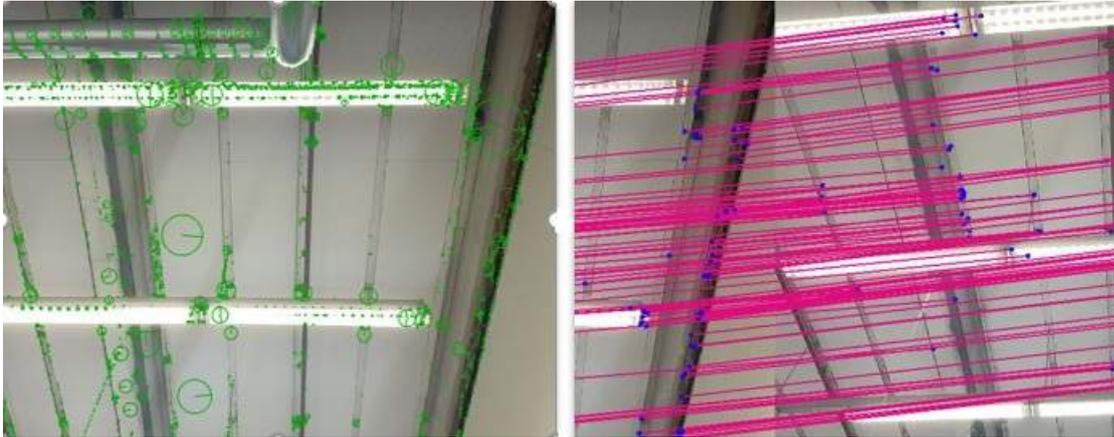
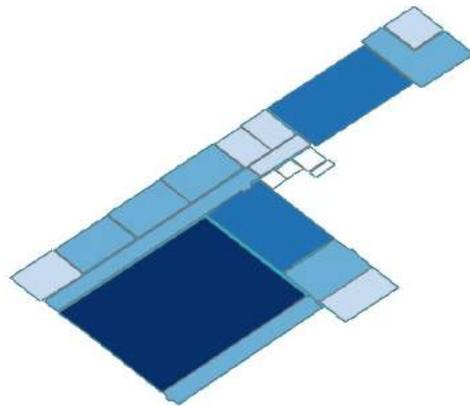


Figure 4.6.: Feature detection (left) and feature matching (right) with SIFT detectors and descriptors

## 4.6. Storage

The indoor model of the case study, which includes the west part of the ground floor in the Faculty of Architecture and the Built environment, was created in ArcGIS pro with indoor capabilities, using the respective CAD files as input. GIS layers were then created and each room of the study area was visualised as a polygon including geometry and location attributes. Additionally, the pre-processed point clouds that act as reference, were attached to the corresponding rooms. Furthermore, based on the indoor package of ArcGIS pro, the network graph, connecting all the adjacent rooms to each other, was created. The indoor model and the network graph were hosted in the ArcGIS online server, as open data so that they could be later used for the web-app.

#### 4. Implementation and experiments



(a) Indoor model of case study



(b) Network graph of case study

### 4.7. Web-app

The creation of the web-app involves requesting the indoor map from ArcGIS online, so that it can serve as a background map. The structure and main functionality of the app were created with [HTML](#). Two [HTML](#) scripts were written, the first so that users can upload their point cloud data and a second where the users are able to see their location as a text, as well as in the indoor model, where the respective room is highlighted and zoomed in, providing a visual insight to the user. The first [HTML](#) page includes two buttons, one where users can browse through their files and add their point cloud and a second one so that they can submit it. The uploaded user point cloud is then compared to all the point clouds of the database, and after a few seconds, the result is returned. The styling of the [HTML](#) pages is implemented with additional [CSS](#) files, that style the buttons, map and background color. JavaScript is also involved in the [HTML](#) files, for the correct set-up of the background indoor map. The connection between the front-end [HTML](#) and the back-end script that was written in Python and includes point cloud pre-processing and registration operations, was implemented using the Flask web-framework, which sets the Uniform Resource Locator ([URL](#)) and the route of the operations. The first time someone uses the web-app, the loaded point cloud is compared against all the point clouds of the database, therefore it takes a few more seconds to show the indoor localisation result, compared to the ones that follow. After the first localisation, the algorithm compares the loaded point cloud to the point clouds of the database that are adjacent to the previous localisation. This operation significantly reduces the time complexity of the algorithm and allows it to take into account from one to maximum four point clouds for each localisation. This number varies depending on the number of rooms that are connected to the examined room.

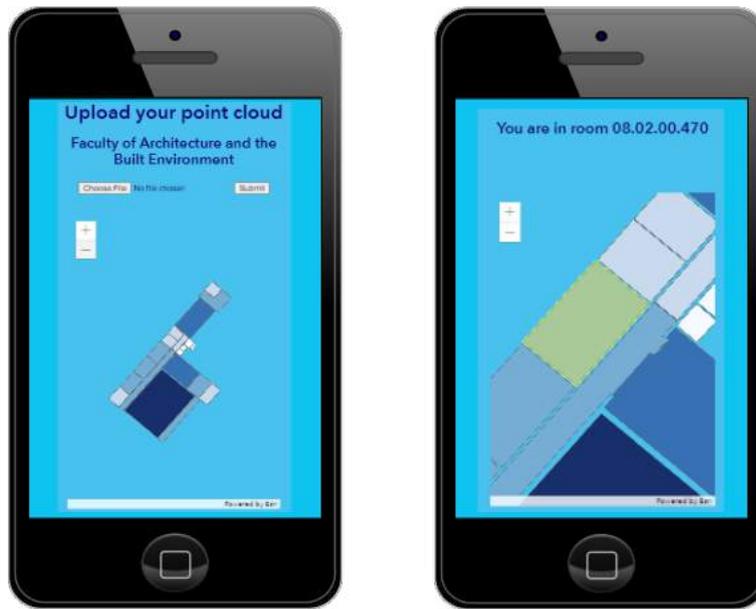


Figure 4.8.: Web-app interface

## 4.8. Location tracking

The indoor localisation results of every user were stored in a list. Therefore, in case users were located in more than one rooms, that means they used a specific path to move from one room to another. Based on the network graph, in the ArcGIS online server, two localisation results starting from a room and ending in another were translated into a specific line of the network graph. Therefore, this information is sent to the ArcGIS online server and the attribute representing path usage is updated by one. Additionally, a unique identity was created for every user based on the urllib python library, in order to protect their privacy by masking their IP address. As a result, there is knowledge that a person followed a specific route in the indoor facility, however it is not possible to track the real identity of this user. The near real-time visualisation of the followed routes, is visualised in ArcGIS online based on the network graph. A heat-map was created, where each user's route is represented by a different color, to differentiate this information between the different users. This information is available near real-time as approximately 30 seconds are needed for it to be shown in the ArcGIS online server. Therefore, information on different movement patterns during a day can be discovered. Furthermore, a dashboard was created in ArcGIS online, so that all the path data are included in a single page. This dashboard is useful for a facility manager, in order to obtain daily, weekly or monthly statistics on how frequently each path is used. Therefore, the different movement patterns that can be retrieved from these statistics, can help the manager optimise operations, such as the optimal distribution of people in an indoor facility, during different times of a day.

#### 4. Implementation and experiments

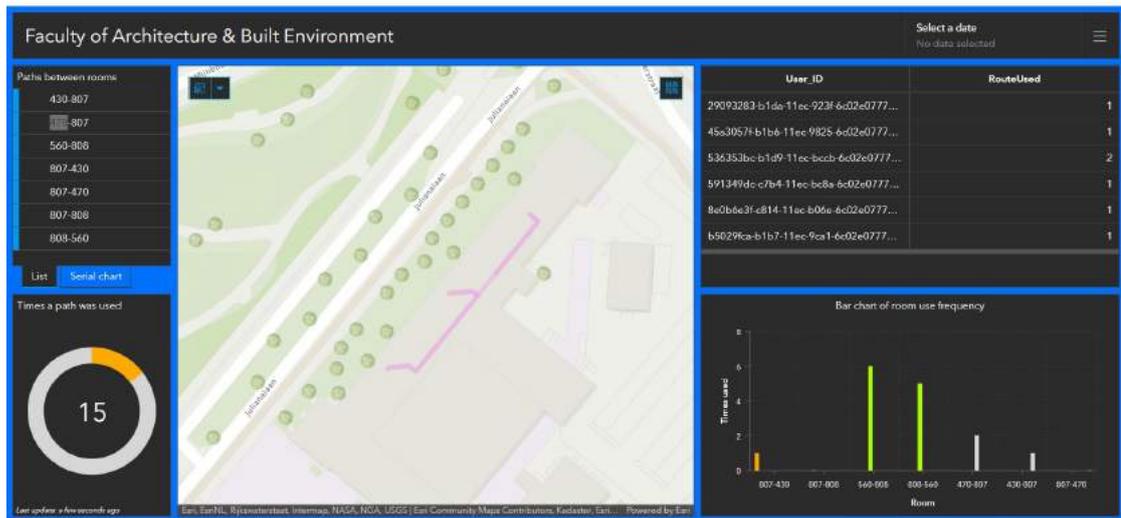


Figure 4.9.: ArcGIS Dashboard

## 5. Results and analysis

In this chapter the results that have occurred from the implementation of the described methodology are presented and analysed. Specifically, the quality of the results is discussed, in parallel to the time efficiency of the proposed algorithms. First, indoor localisation results based on point clouds that were directly acquired and the point clouds that were reconstructed from overlapping images, are discussed. Secondly, indoor localisation results based on image comparison are highlighted and analysed. Afterwards, an analysis and comparison of the different feature detection, description and matching techniques, that can be used for point cloud reconstruction from multiple overlapping images is an additional aspect that is discussed. Last but not least, location tracking, which is based on multiple indoor localisation results is also reviewed.

### 5.1. Indoor localisation from point clouds

In this section the indoor localisation results that were produced based on point cloud acquisition from two different applications, and point clouds that were reconstructed from overlapping images, will be presented and compared.

#### 5.1.1. Point clouds from SiteScape

The first results emerge from point clouds acquired with the SiteScape app. For the acquisition of this dataset, the medium point density setting was selected. This means that the distance between each point is 10 cm.

Room 08.02.00.560 is used as an example to show the point cloud registration with the different algorithms that were used. The remaining registration results are added in [Appendix B](#), to avoid the disruption of the flow of this thesis.



(a) Point-to-Point ICP (fitness=0.943, RMSE=0.06)      (b) Point-to-Point ICP (fitness=0.945, RMSE=0.03)

## 5. Results and analysis

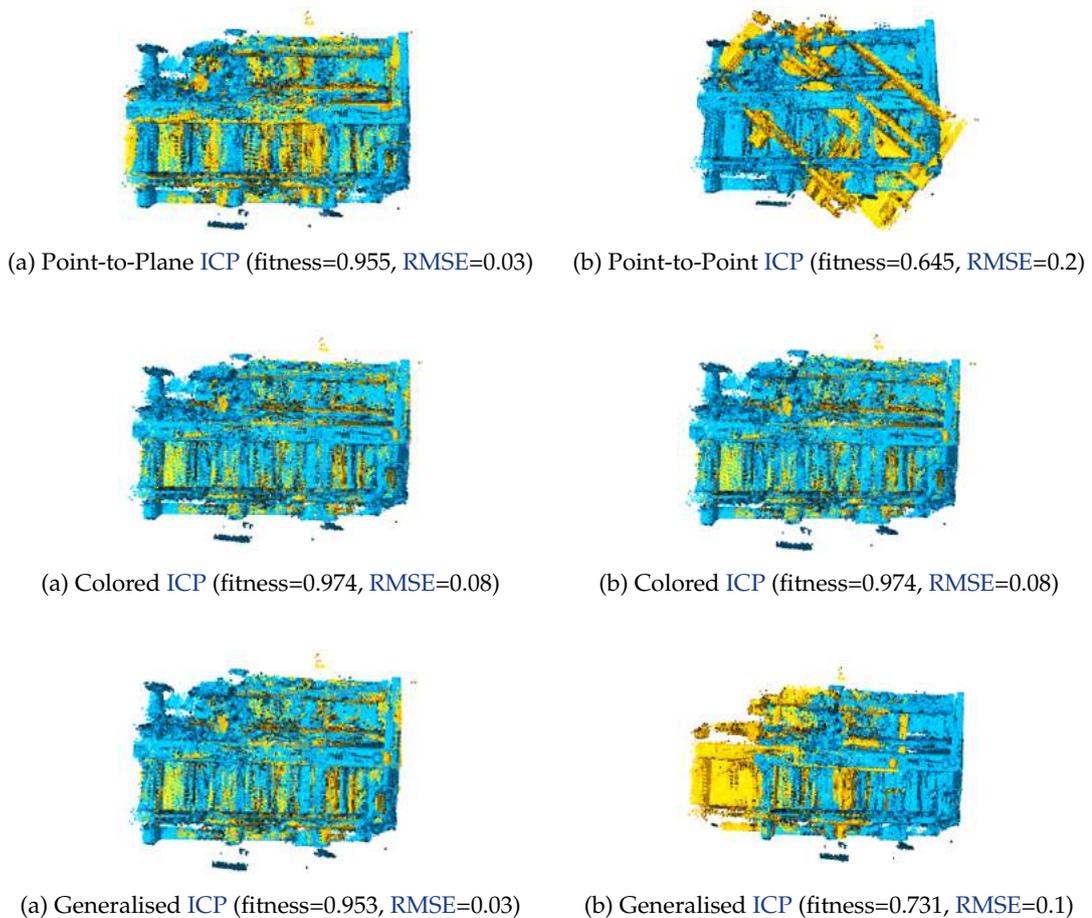


Figure 5.4.: RANSAC global registration (left) and fast global registration (right) of room 560 from SiteScape

The results for room 08.02.00.560 are promising, as in most cases all the point cloud registration methods match the tested room to its reference equivalent. However in Figure 5.2b and Figure 5.4b where fast global registration was combined with the Point-to-Plane and Generalised ICP respectively, room 08.02.00.430 resulted in higher fitness than the correct room 08.02.00.560. However, in both cases the correct rooms had the lower RMSE value. The most accurate results are achieved when Colored ICP was involved, producing similar results when it was combined with fast and RANSAC global registration algorithms, as Figure 5.3a and Figure 5.3b indicate. It has to be noted, that the number of fitness is not important by itself, but it has to be higher compared to the reference point clouds of the remaining rooms.

## 5.1. Indoor localisation from point clouds

| Point clouds | RANSAC Global Registration |                |             |                 |
|--------------|----------------------------|----------------|-------------|-----------------|
|              | Point-to-Point             | Point-to-Plane | Colored ICP | Generalised ICP |
| Dynamic      | 4/5                        | 4/5            | 5/5         | 4/5             |
| Static       | 4/5                        | 4/5            | 5/5         | 4/5             |
| Point clouds | Fast Global Registration   |                |             |                 |
|              | Point-to-Point             | Point-to-Plane | Colored ICP | Generalised ICP |
| Dynamic      | 5/5                        | 5/5            | 5/5         | 5/5             |
| Static       | 3/5                        | 4/5            | 5/5         | 3/5             |

Table 5.1.: Number of correct matches per point cloud registration algorithm

Table 5.1 shows the number of correct matches for each combination of global and local registration algorithms that were applied. The testing includes ten point clouds per method and specifically five for the ceilings that a user acquired while walking, and five more while the user remained static. In total, both RANSAC and fast global registration algorithms have similar results, when combined with different local refinement algorithms. RANSAC is a non-deterministic algorithm, however the high number of iterations that was selected, increases the probability that the result is reasonable. On the contrary, the fast registration algorithm, which does not have the same non-deterministic nature, performs slightly faster, as Figure 4.4 implies.

Concerning, the local registration algorithms, the quality of the results is similar in most of the cases. The worst results are noticed when the fast global registration is combined with Point-to-Point and Generalised ICP, where only 3/5 rooms were matched correctly. Colored ICP produces the best results when it is combined with both global registration techniques, due to the fact that except for the geometry it takes advantage of the color information of each point. Additionally, concerning time complexity, as it was shown in Figure 4.4, it is significantly faster to the other algorithms, due to its multi-scale registration behavior, where it uses downsampled point clouds, unlike the other variations of the ICP.

Furthermore, Table 5.1 shows that the results are better, when users are walking inside a room during data acquisition, in contrast to when they remain static. This is a reasonable outcome, as while a user is walking, the entire ceiling of a room can be captured. On the contrary, while users remain static, they can only capture a specific part of a room's ceiling, in case the room is considerably large, since the range of the LiDAR sensor is approximately five meters. Therefore, in cases where users are unable to move, there are higher chances that the localisation is correct when they capture a part of a ceiling that has characteristic details.

The wrong point cloud matches for some registration techniques, appear between rooms 08.02.00.430 and 08.02.00.470. This confusion arises from the fact that these rooms have almost identical size in squared meters and similar characteristic details in their ceilings, as they are both lecture rooms. Moreover, a mistake in the acquisition of room 08.02.00.430 (Figure 5.17), where the user stumbled during the acquisition, shows why it is significant for the user to move with a steady step. Additionally, the second wrong set is mostly between rooms 08.02.00.808 and 08.02.00.807. This happens, due to the fact that they are both corridors and room 08.02.00.808 is significantly smaller than room 08.02.00.807. Thus, it is possible that this room is wrongly matched as a part of 807. Some rooms, such as 08.02.00.807 which is a long corridor, has a significantly different shape than the common rectangular rooms, hence the possibility that the localisation is wrong is significantly reduced.

## 5. Results and analysis

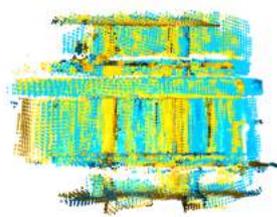


Figure 5.5.: Point cloud of room 08.02.00.430, where the user stumbled during the acquisition

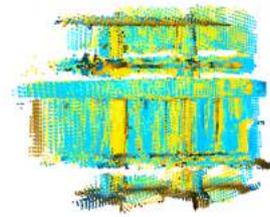
Concerning, wall parts that were acquired along with ceilings, small areas did not affect the results, as some minor wall parts remained in the tested point clouds even after the pre-processing operations. However, in cases where a significant part of a wall is captured, the plane segmentation that was implemented in [Algorithm 3.1](#) could be implemented in a wrong way, as the main plane that is computed, might be the wall instead of the ceiling's upper flat part.

### 5.1.2. Point clouds from Pix4D Catch

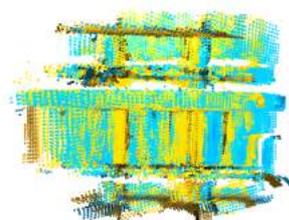
Additionally, point clouds were also acquired with the Pix4D Catch app. During the acquisition with this application, the distance between the points of the acquired point clouds was set to 30 cm. The point clouds will be less dense compared to the ones by SiteScape, in order to examine how the time efficiency of the algorithm and the quality of the results are affected.



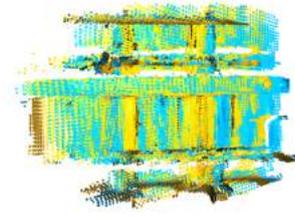
(a) Point-to-Point ICP (fitness=0.954, RMSE=0.15)



(b) Point-to-Point ICP (fitness=0.95, RMSE=0.15)



(a) Point-to-Plane ICP (fitness=0.958, RMSE=0.02)



(b) Point-to-Point ICP (fitness=0.951, RMSE=0.17)

### 5.1. Indoor localisation from point clouds

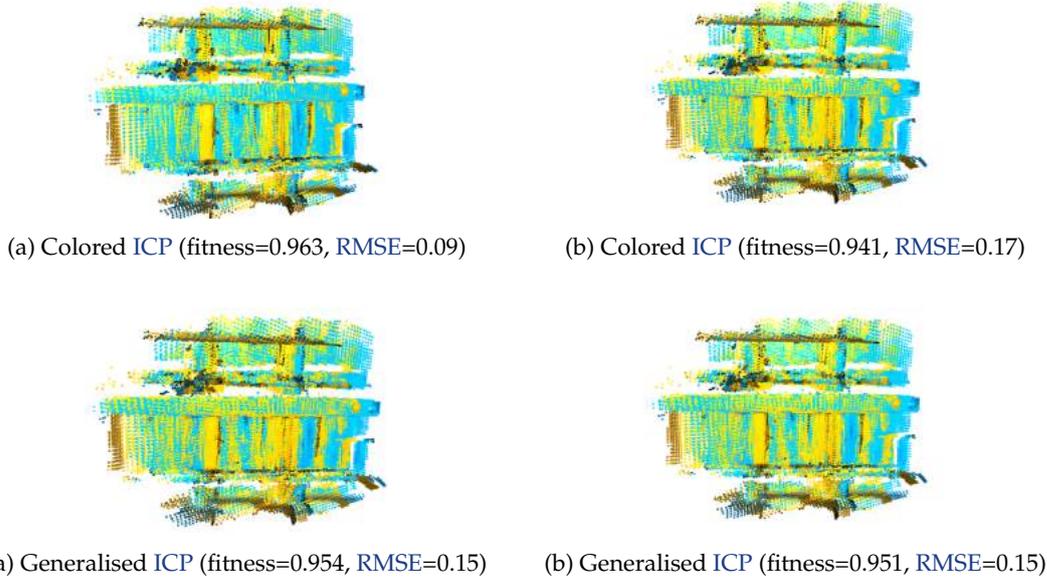


Figure 5.9.: RANSAC global registration (left) and fast global registration (right) of room 560 from Pix4D Catch

Similarly to the previous test case of Section 5.1.1, the results concerning the room 08.02.00.560 are promising, as the combination of global and local registration methods produces the correct result in most cases. Most of the point cloud registration methods have similar results to each other. The combination of Colored ICP and RANSAC provides the higher fitness value, however testing for room 08.02.00.560 provided similarly good results in every combination. However, as it was previously mentioned, it is important that the fitness value of the correct room, is higher compared to the other rooms' value and not as a number by itself. The same is true for the RMSE values, however in that case a lower value is translated to a better result.

| Point clouds | RANSAC Global Registration |                |             |                 |
|--------------|----------------------------|----------------|-------------|-----------------|
|              | Point-to-Point             | Point-to-Plane | Colored ICP | Generalised ICP |
| User         | 5/5                        | 4/5            | 5/5         | 4/5             |
| Point clouds | Fast Global Registration   |                |             |                 |
|              | Point-to-Point             | Point-to-Plane | Colored ICP | Generalised ICP |
| User         | 4/5                        | 3/5            | 4/5         | 3/5             |

Table 5.2.: Number of correct matches per point cloud registration algorithm

In Table 5.2 some overall results regarding the registration of all five rooms are presented, in order to discuss the quality of the registration with all the possible combinations. The quality of the indoor localisation is slightly worse compared to the results produced by SiteScape. The result is reasonable, due to the lower density of point clouds that was chosen for the acquisition, as some small objects on the ceiling might not captured. However, regarding the Colored ICP its results are at a similar level as before, showing the importance of adding color information that the other algorithms do not include.

Concerning global registration techniques, RANSAC shows better results compared to fast, with 18/20 correct room matches, while fast at the same time results into 14/20 correct indoor

## 5. Results and analysis

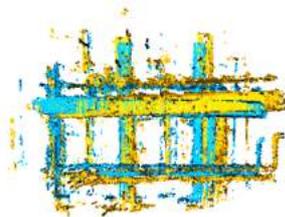
localisation results. As it was also mentioned in Section 5.1.1, RANSAC is a non-deterministic algorithm, however the high number of iterations that were set in the implementation, increases the probability that the results are more reasonable.

As in the previous case, the best results in terms of quality are shown when combining a global registration algorithm to Colored ICP. In this case, the small size of the original point clouds, significantly increases the time efficiency of the algorithm, without significant time difference between the different methods, as it is was shown in Figure 4.4. The worst results are presented for Point-to-Plane and Generalised ICP when they are combined with global registration algorithms, with 7/10 correct indoor localisation results.

The wrong localisation results concern room 08.02.00.430, which is in some cases wrongly mismatched to 08.02.00.470. Their identical size and details, as they are both lecture rooms with similar characteristic details is the reason behind this wrong match. Additionally, while combining fast global registration with Point-to-Plane and Generalised ICP algorithms, an other wrong result was observed between rooms 08.02.00.808 and 08.02.00.807. Specifically, room 08.02.00.808 was wrongly localised as 08.02.00.807. This misinterpretation arises due to their difference in size, as the latter is significantly bigger, therefore, it is possible that room 08.02.00.808 is incorrectly considered as a part of 08.02.00.807.

### 5.1.3. Point clouds from overlapping image sets

In this part the same testing will be applied for point clouds that were reconstructed from overlapping image sets that were acquired with Pix4d Catch.



(a) Point-to-Point ICP (fitness=0.743, RMSE=0.19)



(b) Point-to-Point ICP (fitness=0.774, RMSE=0.2)



(a) Point-to-Plane ICP (fitness=0.521, RMSE=0.23)



(b) Point-to-Plane ICP (fitness=0.771, RMSE=0.2)

## 5.1. Indoor localisation from point clouds

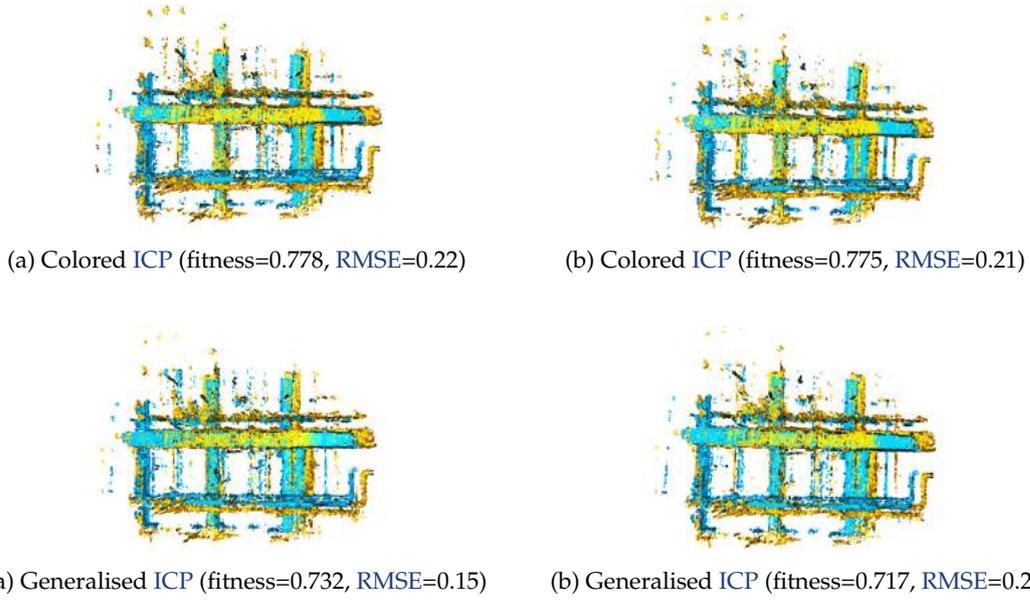


Figure 5.13.: RANSAC global registration (left) and fast global registration (right) of room 560 from reconstructed images of ceilings

Room 08.02.00.560 was also chosen for testing the point clouds that were tested from overlapping image sets. From Figure B.65 it can be noticed that in this case, due to the fact that the acquisition involves images, the parts of the ceilings that are behind installations could not be acquired, hence not modelled. However, that does not comprise a problem in most cases. An important observation is that the point cloud reconstruction from image sets, may result into point clouds that have a different scale compared to the point clouds that were directly acquired. This is an outcome of the 3D reconstruction process, as only the intrinsic of the cameras are known and the position of the 3D points is computed based on the projections from the 2D space, so the true scale of the scene cannot be accurately recovered. Therefore, the 3D reconstruction of a ceiling is unique up to a scaling factor. For this reason, point clouds that were reconstructed from images have to be used as reference, for the matching results to be accurate. Concerning the results of room 08.02.00.560, based on Figure B.65, the best results in terms of fitness can be once more noticed, where Colored ICP was involved, however Point-to-Point and Generalised ICP combinations also provided high fitness values. A bad matching is observed while combining RANSAC to Point-to-Point ICP with 52% fitness value (Figure B.47a), however this number was higher than the ones of the respective rooms, resulting into a correct localisation.

| Point clouds | RANSAC Global Registration |                |             |                 |
|--------------|----------------------------|----------------|-------------|-----------------|
|              | Point-to-Point             | Point-to-Plane | Colored ICP | Generalised ICP |
| User         | 4/5                        | 3/5            | 5/5         | 4/5             |
| Point clouds | Fast Global Registration   |                |             |                 |
|              | Point-to-Point             | Point-to-Plane | Colored ICP | Generalised ICP |
| User         | 4/5                        | 3/5            | 5/5         | 5/5             |

Table 5.3.: Number of correct matches per point cloud registration algorithm

## 5. Results and analysis

Both global registration algorithms have similar quality of results and specifically the combination of **RANSAC** with local refinement algorithms, gives 16/20 correct localisation results, while fast with the same combinations issues indoor localisation correctly, 17/20 times.

Concerning local registration algorithms, **Colored ICP** appears to have the maximum success rate with 10/10 correct localisation results, while the worst results involve **Point-to-Plane ICP** with 6/10 correct results. Once more, the addition of color information to the existing geometry can significantly improve the point cloud registration results, ensuring a high success indoor localisation rate. Except for the quality of the results, the time efficiency of **Color ICP** has to be reminded, establishing it as concrete point cloud registration algorithm.

In contrast to the previous datasets, the wrong matches do not include the same rooms as before, however the results in overall are similar. This time, there are different combinations of different rooms that were matched incorrectly. This is a result of the scaling factors, during the reconstruction operation that model the third dimension with a scale ambiguity. Thus, rooms that appear to have a different size in reality, might be modelled similarly in terms of size, a fact that could result into wrong localisation, when the protruding components of the ceilings are not enough to differentiate the rooms between each other. An example of the scale ambiguity is visualised in [Figure B.50](#) and [Figure B.49b](#). It should be pointed out that the 3D reconstruction is a time-consuming process, even for the most specialised software. Therefore, indoor localisation based on 3D reconstruction that takes place based on the acquisition of overlapping images, cannot be used for real-time applications.



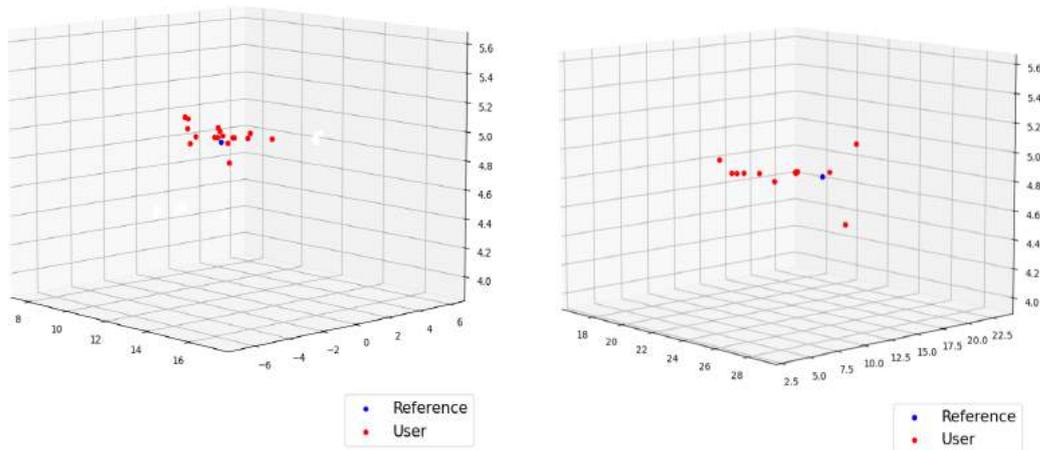
(a) Point cloud of room 08.02.00.560, acquired by SiteScape



(b) Point cloud of room 08.02.00.560, reconstructed from overlapping images

### 5.1.4. Performance parameters

This section presents some performance parameters that were calculated in order to test the robustness of the results. A discussion concerning the availability and yield of the current localisation approach, as well as some accuracy and precision metrics, for the results of the point-cloud based indoor localisation approach.

**Accuracy/Precision**

(a) Scatter plot with centers of reference and user point clouds after point cloud matching in room 08.02.00.808      (b) Scatter plot with centers of reference and user point clouds after point cloud matching in room 08.02.00.807

Figure 5.15a and Figure 5.15b show the centers of the respective reference point cloud with blue color, as well as the centers of different user point clouds after the implementation of the point cloud registration algorithms and specifically RANSAC based global registration and Colored ICP local refinement. The results concerning room 08.02.00.808 reveal good accuracy, as most of the centers of the user point clouds are a few centimeters away from the center of the reference point cloud, while at the same time the precision is adequate, as most of the centers of the user point clouds are close to each other. On the contrary, the same results for room 08.02.00.807 are worse concerning accuracy and precision, since the centers of the user point clouds are further away from the center of the reference point cloud and at the same time far from each other. This has to do with the size and length of the room 08.02.00.807, that is a corridor with similar and lengthy protruding installations on the ceilings, therefore it is possible that the user point clouds are matched to the reference point cloud on a different part of those installations, further away from the center of the point cloud. However, in both cases there is good accuracy and precision regarding the height dimension, which shows that the flat part of the ceilings of the user and reference point clouds is in most cases correctly matched. It has to be noted that even in cases where the center of the user point cloud is not very close to the one of the reference point cloud, that does not necessarily affect the indoor localisation results, as the fitness of the compared point clouds of the same room is still higher compared to different rooms.

**Availability and yield**

The developed point cloud based localisation method is possible, in buildings that include a database of reference point clouds of ceilings for each room of the indoor facility. Devices such as the iPad 12 pro, that was used in this thesis and incorporates a LiDAR sensor are required for the acquisition of reference point clouds, as well as for a user to capture a point cloud of a ceiling and achieve indoor localisation. If an Apple device with a LiDAR sensor is not available, a laser scanner could be used to perform the acquisition of reference point clouds. In general, the

## 5. Results and analysis

range of the LiDAR sensor is approximately 5 meters, therefore ceilings such as the Orange Hall in the Faculty of Architecture and the Built Environment cannot be captured without mounting the sensor on a monopod or a tripod. The latter does not apply to image acquisition, since camera sensors can acquire images in larger distances. The created pipeline provides satisfactory solutions for ceilings with characteristic details, such as the ones in the Faculty of Architecture and the Built Environment. However, the quality of the solution, might not be the same when applied to ceilings that are primarily flat, with fewer characteristic details, or ceilings that include glass, whose reflection abilities might affect the indoor localisation result.

### Cost

The creation of a database with reference point clouds and images for every ceiling of an indoor facility requires some devices. Regarding point cloud acquisition, an Apple device such as an iPhone 12 pro or an iPad 12 pro are required, as well as some non-commercial software. The cost for these devices is approximately 1000 euros. Alternatively, a laser-scanner could be rented in order to perform the acquisition. Concerning image acquisition, camera sensors are included in each mobile device, so no further devices are required.

## 5.2. Indoor localisation from images

This subsection includes techniques that were implemented based on image acquisition. First, indoor localisation results based on the comparison of single images is performed. The indoor localisation result is based on the number of matches between the user and the reference images. Additionally, different combinations of feature detection, description and matching techniques, which are a vital step towards 3d reconstruction, are analysed.

### 5.2.1. Single images

Table 5.4 shows the indoor localisation results, based on ORB and SIFT feature detectors, as well as the homonymous descriptors and two feature matching techniques, brute-force and FLANN.

| 5 MP Camera |         |           |
|-------------|---------|-----------|
|             | ORB-ORB | SIFT-SIFT |
| Brute-Force | 4/5     | 5/5       |
| FLANN       | 4/5     | 5/5       |
| 8 MP Camera |         |           |
| Brute-Force | 4/5     | 5/5       |
| FLANN       | 4/5     | 5/5       |

Table 5.4.: Number of correct matches per feature detection, description and matching techniques

The results are based on images that were taken from two different cameras with 5 and 8 MP resolution respectively. Both cameras perform similarly resulting into 18/20 correct room matches

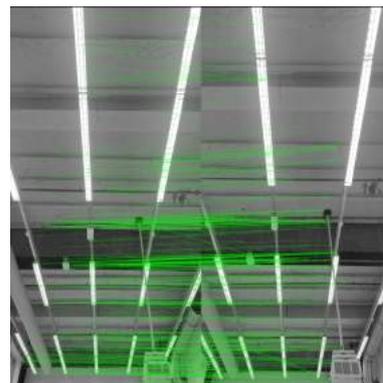
and they only differ in processing time as Figure 4.5 shows. Additionally, the two feature matching techniques have similar efficiency when they are combined with the two different detectors and descriptors, while brute-force performs slightly faster than FLANN. However, the latter can be more efficient than brute-force, when large datasets are involved. FLANN results into a higher number of matches between the user image and the reference image of the correct room in most cases. The same can be mentioned about SIFT, which results into more matches between the images compared to ORB, however the indoor localisation is calculated with worse time efficiency. In terms of quality, the suitability of SIFT, lies in the fact that it is scale and rotation invariant, whereas ORB is only rotation invariant and robust to noise. As a result, in case SIFT is used, the height and angle of the device do not affect the result. The time efficiency of SIFT, could be improved, by implementing the SURF detector and descriptor, an operation that did not take place, due to the fact that its use is patented from the OpenCV library. The ratio test that was applied in each experiment was strict, in order to avoid false correspondences, that were probable due to the common installations between the different rooms. The most clear results were observed concerning one test image of room 08.02.00.470, where approximately 400 matches were observed between the user and reference image, which are significantly more than the other reference images. This is an outcome of the similarity of the user and reference images, as they were acquired from a similar angle and cover approximately the same part of the ceiling. In other cases where the viewpoints of the user and reference images were different, the indoor localisation results were correct, as the user image had the most matches with its corresponding reference image, however the number of matches was significantly lower, between 50 and 100.

The wrong localisation results, were related to room 08.02.00.807, that cannot be entirely captured from a single image, due to its length. Therefore, in terms of size, it appears to be similar to different rooms of the case study. However, this result can be partially solved, in case the data acquisition is performed, by holding the sensor almost perpendicular to the ceiling, so that a bigger part of the ceiling is captured.

In this testing, there are no differences between the two different cameras regarding the quality of the results. However, certain illumination changes that create blurry areas, may significantly affect the intensity of each pixel of the tested images. In this situation, a high resolution camera could better capture the reality and avoid these blurry parts in the images. However, a drawback of using cameras with high resolution, is that they tend to produce bigger image files, that are not suitable for real-time applications, due to the necessity of a time efficient solution.



(a) Feature extraction with SIFT



(b) Feature matching with SIFT and FLANN

## 5. Results and analysis

Some wrong matches are highly affected by the ceiling lights that are on, during most part of the day in the Faculty of Architecture and the Built environment. These lights tend to create blurry areas around them, tampering with the real intensity values of the pixels. Additionally, the intensity values of these areas might appear similar to the windows, resulting into wrong matches between the windows and the lights, when two images are compared as it can be seen in the case of room 08.02.00.430 (Figure 5.17). Hence, during the acquisition, windows should be avoided as much as possible, due to their reflective ability.

Overall, indoor localisation based on the comparison of the features of an image seems really promising, however additional testing regarding lighting conditions and viewpoints, has to be implemented to produce safe conclusions about this method. Testing in a larger database is also a challenge, as well as the implementation of the SURF detector and descriptor, to check the suitability of this indoor localisation method based on images, for real-time applications.



Figure 5.17.: Feature detection with SIFT for room 430

### 5.2.2. Overlapping images

The acquisition of overlapping images targeted not only to 3D reconstruction of the ceilings, but also in order to check different combinations of feature detectors, descriptors and matching techniques for images of ceilings. This testing was performed in PhotoMatch and the results are presented in this section.

Table 5.5 focuses on the number of key-points, as well as their percentage that was used for matching between the overlapping images of the chosen subset, in order to test the suitability of each combination of detectors and descriptors in terms of key-point selection. For each combination of feature detectors, descriptors and matching techniques an upper threshold of 5000 points was set. It can be noticed that the combination of SURF as a detector and descriptor detects the maximum number of key-points with both feature matching algorithms. That is also observed when the SURF detector is combined with the SIFT descriptor. On the contrary, the lowest number of key-points is detected when SIFT and BRIEF are combined with almost 3500 thousand key-points. The latter happens due to the simplicity of the BRIEF descriptor which targets in fast description from simple intensity difference tests. Regarding the percentage of key-points that are used for matching, SURF detector with SIFT descriptor and FLANN matching take into advantage approximately 13% of the detected points, while the combination of SIFT detector, SURF descriptor and FLANN uses less than 1% of the detected key-points

## 5.2. Indoor localisation from images

| Detector | Descriptor | Matcher     | Average Number of keypoints | Percentage of keypoints used for matching (%) |
|----------|------------|-------------|-----------------------------|---|
| SIFT     | SIFT       | Brute Force | 3671                        | 8.96  |
|          |            | FLANN       | 3671                        | 9.02  |
| SURF     | SURF       | Brute Force | 5000                        | 10.60   |
|          |            | FLANN       | 5000                        | 9.82  |
| ORB      | ORB        | Brute Force | 4939                        | 3.34  |
|          |            | FLANN       | 4939                        | 6.09  |
| SIFT     | BRIEF      | Brute Force | 3495                        | 4.46  |
|          |            | FLANN       | 3495                        | 6.67  |
| SIFT     | BRISK      | Brute Force | 3585                        | 4.32  |
|          |            | FLANN       | 3585                        | 4.88  |
| SIFT     | SURF       | Brute Force | 2896                        | 1.04  |
|          |            | FLANN       | 3671                        | 0.68  |
| SURF     | SIFT       | Brute Force | 5000                        | 12.06   |
|          |            | FLANN       | 5000                        | 12.72   |
| SURF     | BRIEF      | Brute Force | 4928                        | 6.98  |
|          |            | FLANN       | 4928                        | 9.38  |
| SURF     | BRISK      | Brute Force | 4604                        | 10.36   |
|          |            | FLANN       | 4604                        | 12.34   |

Table 5.5.: Number of keypoints and their percentage used for matching per combination

for feature matching. This is a result of the size of the vectors of **SIFT** and **SURF** descriptors, which have a size of 128 and 64 elements, showing that **SIFT** entails more details concerning the description of the sub-region of the tested key-points. In most cases, **FLANN** uses a higher percentage of key-points for matching, compared to brute-force except when the **SIFT** detector and **SURF** descriptor are combined, however the difference is minor.

## 5. Results and analysis

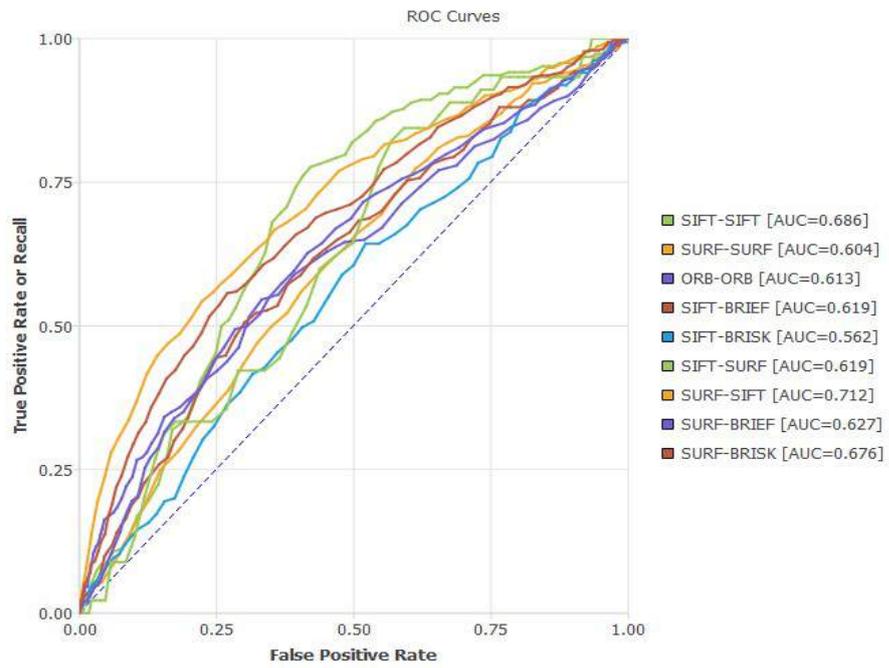


Figure 5.18.: ROC curves between 2 images with Brute-force matching

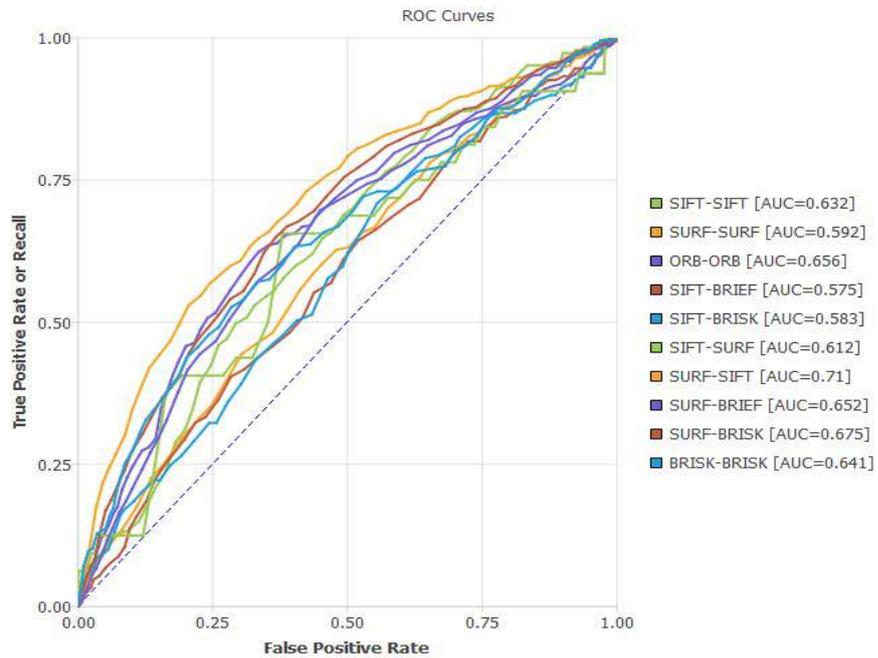


Figure 5.19.: ROC curves between 2 images with FLANN matching

Figure 5.18 and Figure 5.19 show the ROC curves for an image set when brute-force matching and FLANN are used respectively. Overall, both feature matching techniques reveal similar results with a recall of approximately 63%. This number shows how many matches were actually true and not mistakenly matched by the algorithms. If the information of these graphs is combined with Table 5.5, the finer results can be noticed when SURF detector and SIFT descriptors are combined, with approximately 71% of true to false positive ratio. On the other hand, the worst performance is observed when SIFT detector is combined with BRIEF and Binary Robust Invariant Scalable Keypoints (BRISK) descriptors.

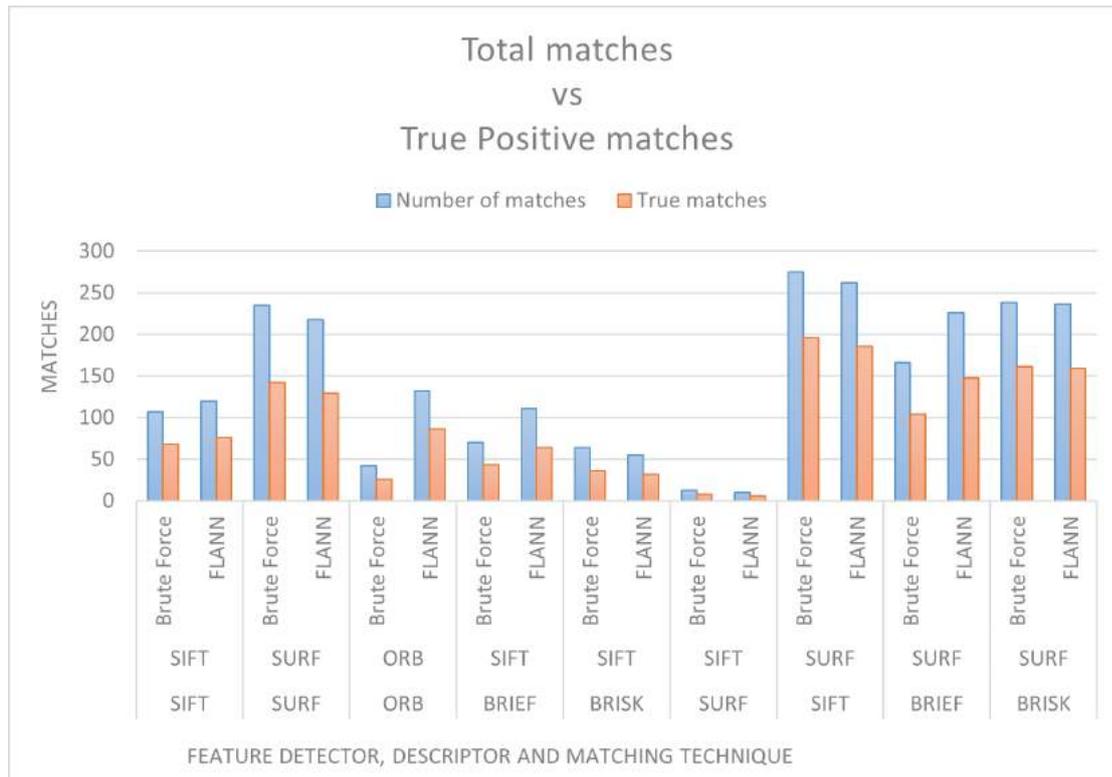
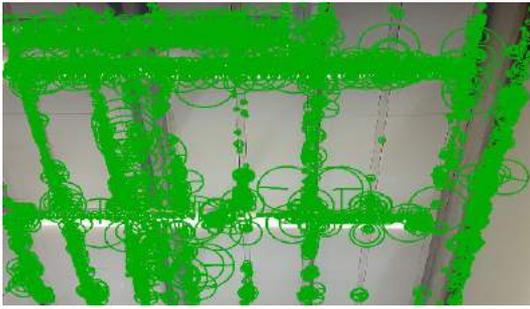


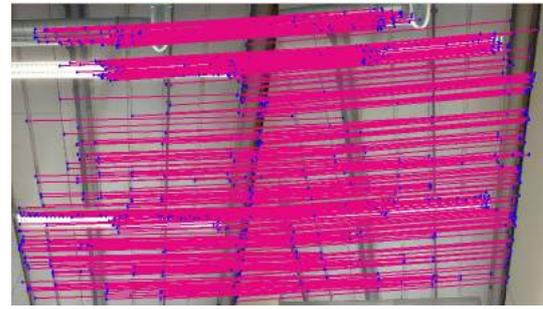
Figure 5.20.: Total number of matches and true positive matches

Figure 5.20 provides an additional visual insight towards the assessment of the different techniques. Specifically, the number of total matches and the true positive matches are shown. The best performance is observed when the SURF detector and SIFT descriptor are used with approximately 250 matches for both feature matching techniques. The most under-performing combination is the one of SIFT detector and SURF descriptor with a very low number of matches. The following figures show the feature extraction and matching for the best and worst combinations respectively, where the diameter of the circles, indicates the meaningful key-point neighborhood.

## 5. Results and analysis



(a) Feature extraction with SURF detector and SIFT descriptor



(b) Feature matching with SURF detector and SIFT descriptor



(a) Feature extraction with SIFT detector and SURF descriptor



(b) Feature matching with SIFT detector and SURF descriptor

### 5.3. Location tracking

The location tracking results are based on the different indoor locations of different users in different times of a day. Therefore, the quality of the followed paths is a direct outcome of the indoor localisation quality. The results are available in the ArcGIS online Server and can be seen in near real-time in a map, that is updated every 30 seconds.

To test the accuracy of the location tracking algorithm, a ground truth was set, based on the path that the user originally followed and was compared to the path, as it is visualised in the final product. This is shown in [Figure 5.23](#).

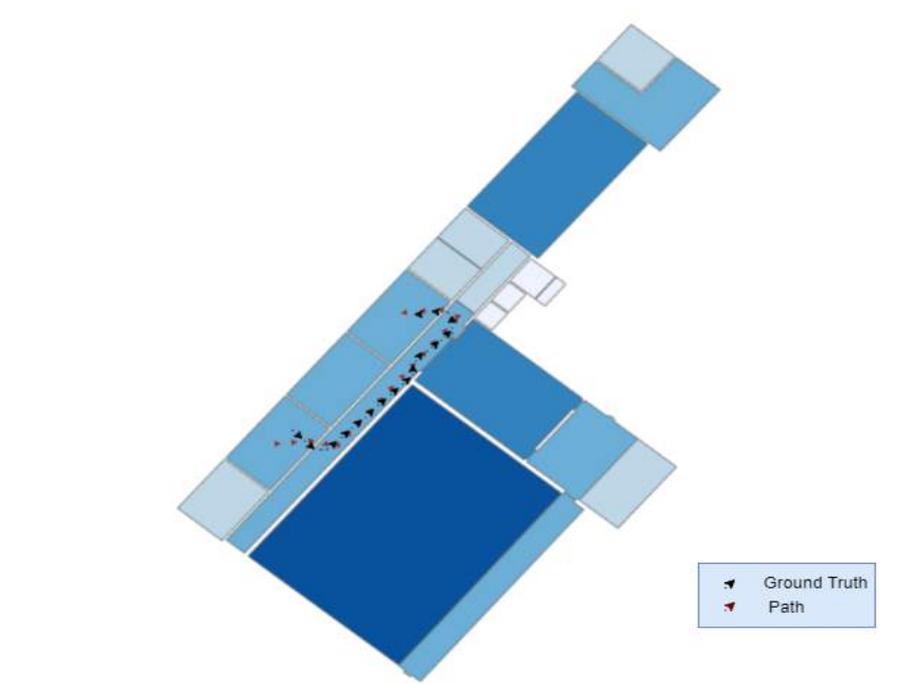


Figure 5.23.: Ground truth and visualised paths between rooms 08.02.00.430, 08.02.00.807 and 08.02.00.470

Figure 5.23 shows the path of a user that moved between rooms 08.02.00.430, 08.02.00.807 and 08.02.00.470. The indoor localisation was performed correctly for these three rooms, therefore the ground truth is similar to the path as it is visualised in ArcGIS Pro. Some differences exist due to the indoor network that is used to visualise the paths, as the center of each room is the representative node and the fact that the rooms are connected to each other with lines, therefore small deviations when the user is not moving completely straight cannot be detected.



## 6. Conclusion and discussion

This chapter presents the conclusions of this thesis which examined an alternative way of indoor localisation and in extent location tracking of different users, using LiDAR sensors of an iPad pro, as well as camera sensors of mobile devices to capture point clouds and images of the ceilings, in an effort to replace the most used methods that involve Wi-Fi fingerprinting and Bluetooth sensors. This is achieved by answering the research questions that were introduced in Chapter 1. Furthermore, a discussion including the contribution of this thesis to the scientific community, as well as the limitations of the current approach are presented. The last section includes the future work that is proposed based on new research questions that emerged from this thesis.

### 6.1. Research questions

In this section, the secondary research questions of this thesis will be first discussed, followed by the main research question:

**“To what extent can ceilings with characteristic details be used for indoor localisation purposes?”**

#### 6.1.1. Answer to the secondary questions

1. *“Which parameters (measuring angle, height, part of the room) should the user take into account while acquiring point clouds and images of ceilings?”*

Data acquisition concerning both point clouds from LiDAR sensors, as well as images from different devices is an important operation that can improve the indoor localisation results if it is implemented properly. Regarding point clouds, it is important that a user performs the acquisition while moving with a steady pace and without sudden changes in the measuring angle and height. The range of the LiDAR sensor is approximately 5 meters, so users should hold their device as high as possible, in case they cannot use an extensible accessory, such as a monopod or a tripod. This way, problems such as the one showed in Figure 5.17, where a sudden change in the position of the device resulted in a wrong distance calculation from the LiDAR beam, can be avoided. The iPad’s MEMS gyroscope, a microscopic version of a gyroscope, that combines mechanical parts in a small scale with electronic circuits, provides the user with the ability to easily capture point clouds, while rotating the device, so that a bigger surface is acquired. The same rules also apply to image acquisition. As it was shown in Chapter 5, where single images are used for indoor localisation based on feature matching, in case the acquisition includes large rooms that cannot be entirely captured, placing the sensor almost perpendicularly

## 6. Conclusion and discussion

to the ceiling can improve the indoor localisation results, as a larger part of a ceiling can be captured. For both point clouds and images, if the whole room cannot be captured, it is vital that the part of the room that is captured includes protruding installations, so that it can be correctly matched to the reference data that include the entire ceiling. Last but not least, walls are surfaces that might significantly affect the indoor localisation result. A small wall partition might not affect the localisation results, however if a significant part of a wall is captured, the wall might be selected as the main plane of the point cloud in the pre-processing step.

2. *"Which is the optimal point cloud registration algorithm to achieve indoor localisation from ceiling data?"*

Point cloud matching was performed based on three different datasets, including point clouds acquired by SiteScape, Pix4D Catch, and point clouds that were reconstructed from overlapping images. The different combinations of global and local registration algorithms resulted into indoor localisation with some differences for each dataset, however in terms of quality and time efficiency, the combination of **RANSAC** global registration with the Colored variation of the **ICP**, proved to be the optimal solution. In spite of **RANSAC**'s non-deterministic nature, the high number of iterations that was chosen, transforms it into a robust global registration choice, with a high possibility that a reasonable result is produced. In contrast to the other implemented local refinement algorithms, Colored **ICP** adds color information to the geometry as its name indicates, hence this additional information is the reason behind the suitability of the algorithm. The multi-registration scheme of Color **ICP** significantly improves the time efficiency of the algorithm, making it a concrete choice for real time applications that use point clouds of ceilings for indoor localisation.

3. *"Which is the optimal image matching algorithm to achieve indoor localisation from ceiling data?"*

Feature matching algorithms, such as brute-force and **FLANN** were combined with feature detection and description algorithms to investigate their reliability towards indoor localisation. These combinations were implemented for two different purposes, direct indoor localisation based on single images of the ceilings, as well as an intermediate step towards 3D reconstruction from overlapping images. For the first approach, **SIFT** detector and the homonymous descriptor provided the optimal indoor localisation results when combined with both brute-force and **FLANN**, however the time efficiency of this combination was worse than the one by **ORB**. The scale and rotation invariant character of **SIFT** makes it adaptable and robust to different types of distortion, illumination and noise. In terms of time efficiency, brute-force performs slightly faster than **FLANN**, due to the small size of the dataset that was used. **FLANN** resulted into a higher number of matches between the user and reference images, compared to brute-force, with a higher match difference between the correct and wrong reference images, ensuring the quality of indoor localisation. As a result, **ORB** performed more efficiently than **SIFT** as the indoor localisation is performed at approximately 10% of the time that **SIFT** takes, as shown in **Figure 4.5**. Thus, **ORB** is more suitable for real-time applications, however the higher result quality was noticed by **SIFT**. This uncertainty is solved during the second approach, that was implemented in Photomatch, a software that provided the chance to use a larger variety of feature detection and description algorithms. The number of matches, as well as the true to false positive ratio of **Figure 5.20** showed that the combination of **SURF** feature detector with the **SIFT** descriptor provides the most optimal results. **SURF** could be

considered as a faster version of SIFT, therefore it comprises a more time efficient way to present SIFT's quality.

4. *"Are LiDAR point clouds acquired by an iPhone device an accurate and accessible solution towards indoor localisation?"*

Apple's iPhone/iPad 12 and 13 Pro that include LiDAR sensors led to a new era of innovation and a path for less or non-technical users to use the capabilities of these sensors, in order to capture the third dimension of a scene. Currently, different user-friendly applications, such as the ones that were used during this thesis, facilitate the point cloud acquisition. Therefore, point clouds can be easily acquired by a holder of a newly released Apple device, as well as a few Android devices with increased capabilities and specifically ones that include a ToF camera and Android's ARCore. This new development indicates that more and more mobile devices will include LiDAR sensors in the future. The users of applications, such as SiteScape and Pix4D Catch have the ability of choosing the resolution and density of the created point clouds, which are captured with an accuracy of +/-1 inch, showing that the LiDAR sensors in Apple's devices can be even used for construction projects.

5. *"Can the proposed pipeline aid towards facilitating localisation in emergency situations?"*

In emergency situations, such as fires in indoor environments users should be able to find the name of the room they are located in, as well as a way to communicate that information with first-aid responders. In that way, users could capture a point cloud for a few seconds even in cases they are unable to move, by just rotating their device. In case a room is large and the user cannot capture the entire room, it is important that the part of the ceiling that is captured includes some characteristic details, so that indoor localisation is successful. This information is also important to first-aid responders, which are usually unaware of the number and names of the rooms in an indoor facility. The created web-app takes a few seconds to perform localisation and return the result to the user, depending on the size of the acquired point cloud. The point clouds with lower density, that were acquired by Pix4D Catch, are more suitable for a real time application, as indoor localisation is performed in less than 10 seconds, showing adequate results. Currently, the first localisation is implemented based on comparison of the user and all the reference point clouds, an operation that might significantly decrease the time efficiency of the algorithm, in buildings with a high number of rooms. Concerning images, they comprise a more straightforward way of capturing the ceiling, as that takes a click of a button, however the web-app does not currently support feature matching based on images. Additionally, protruding installations might cover some parts behind them, that cannot be shown in images. It has to be noted, that fumes that are produced during a fire emergency might affect the visibility of the ceiling and therefore do not optimally capture the surface. Indoor localisation results from point clouds concerning emergency situations are promising, however the time efficiency of the app, especially regarding the first localisation of a user has to be improved, whereas images have to be also incorporated.

6. *"How accurate is location tracking and does it respect user privacy?"*

Location tracking is based on different localisation results of users during different times of a day. As it was shown in Figure 5.23 the ground truth of the original path of the tested users, was slightly different than the visualised path in the heat-map. Since localisation could only be implemented in room-level, the center of each room was chosen as a representing node and the path was represented as a line connecting different nodes. The path

## 6. Conclusion and discussion

between two different rooms is based on the indoor network, that is created, based on the door openings of each room. Therefore the detailed movement of each user cannot be exactly interpreted, however information about the path that someone has followed can be shown. The indoor localisation algorithm is highly affected by the first localisation, as for the next one only the adjacent rooms are checked. Therefore, a mistake in the first room's localisation could lead to the misinterpretation of the path that a user followed. To reduce the possibility that the first localisation is wrong and also improve time efficiency, the current algorithm could be combined with Wi-Fi fingerprinting, in order to reduce the number of rooms that are checked in an indoor facility. Different random IDs were automatically created for every user, in order to associate them with the path information in the server. The random ID is a combination of the IP address of the user in a mixed order and some other random characters. Therefore, information about the paths that the same person followed is available, but it cannot be tracked back to the real identity of the user.

### 6.1.2. Answer to the main research question

Answering the secondary research questions of this thesis, enabled the easier answer of the main research question:

**"To what extent can ceilings with characteristic details be used for indoor localisation purposes?"**

This thesis aimed to investigate the reliability of ceilings for indoor localisation, based on [LiDAR](#) and image sensors, which are incorporated in up-to-date mobile devices. The ceiling data was acquired in the Faculty of Architecture and the Built Environment, in which ceilings have protruding installations, such as pipes and lights, a common characteristic in various indoor facilities in the Netherlands, mostly in public spaces and some private facilities. The indoor localisation from the different testing that was implemented showed promising results, both in terms of quality as well as time efficiency, as the scope of the thesis was to be able to perform real-time localisation of large indoor environments, focusing on ceilings with characteristic details. However, it has to be noted that 3D reconstruction from overlapping images is a time consuming process, therefore it cannot be implemented on the fly, for a real-time application. Based on the results, a point cloud acquisition of a few seconds is enough to indicate the room that users are, especially when the whole ceiling can be captured. In case a ceiling is partly acquired, the indoor localisation result depends on the uniqueness of the captured part. However, an image-based localisation method has to be incorporated in the final product and the time-efficiency of the point cloud algorithm concerning the first localisation has to be improved, so certain conclusions can be drawn for emergency situations. Additionally, the point cloud acquisition of ceilings led to promising localisation results while implemented dynamically, during continuous acquisition between different rooms. The range of the current [LiDAR](#) sensors is approximately 5 meters, therefore point clouds of ceilings in buildings with high ceilings cannot be captured, except the mobile device is mounted on an extensible monopod or tripod. However, this unavailability in acquisition, can be also translated into information that a person is in a room with a high ceiling. In parallel, the current implementation focuses on the protruding features of the ceilings, by detecting the flat plane of the ceiling and omitting it during point cloud registration. Therefore, it cannot be used for ceilings without characteristic details, without slightly adjusting the current pipeline. However, in this case the indoor localisation based on ceilings would be solely based on the dimensions of the ceiling, thus the result would be less accurate. Similar problems could be observed in ceilings that mostly consist of glass,

as its reflective character could result into problems both while using point clouds as well as images.

## 6.2. Discussion

This section includes a discussion concerning the thesis, including the applications where it could be utilised in, as well as the contribution of the thesis to the research community. Furthermore, the limitations of the existing pipeline are analysed.

### 6.2.1. Contribution

In this thesis, ceiling data was used as an alternative way of performing indoor localisation and in extent location tracking of users, in an effort to investigate if the implemented pipeline, can work efficiently and substitute the varied used localisation methods that mostly involve Wi-Fi fingerprinting and Bluetooth sensors. The implemented pipeline, incorporates both LiDAR sensors included in the recently released Apple mobile devices and a few Android phones, as well as camera sensors that are available in every phone. Therefore, indoor localisation becomes possible for a variety of users, without the need of additional equipment. The only requirement of this pipeline, is the existence of point clouds of ceilings that will act as reference for every room of the indoor facility. The implemented pipeline, that includes both indoor localisation from point clouds and images of the ceilings, could be applied in buildings with large rooms, such as airports, and train stations where people can easily lose their orientation. Therefore, localisation can be used as an affirmation that users are on the correct route towards their final destination. Point clouds provide a dynamic aspect to indoor localisation, especially with the use of ceilings, which were chosen due to the fact they usually are not altered during time. The static point cloud acquisition can help people that are unable to move during an emergency situation, understand their current location, an information that is vital if transmitted to first-aid responders. However, improvements are needed in the final product so it can be used for emergencies in real-time with the optimal efficiency, due to the importance of this application. Additionally, with the dynamic acquisition of point clouds, users can perform data acquisition while moving between different rooms, setting the basis for location tracking and additionally navigation. To reduce the acquisition problems stated in [Section 6.1.1](#), the LiDAR device could be mounted on an automated device, that is capable of moving in an indoor environment, by specifying a proper angle and height for the sensor. This continuous acquisition by an automated device could help towards the optimal mapping of indoor facilities based on point cloud acquisition. Regarding location tracking, it comprises an extension of the indoor localisation results, as it is implemented based on two or more localisation results. Information of most used paths is vital in an indoor facility, as its manager, can use daily, weekly or monthly statistics and optimise the distribution of people in an indoor space, based on the noticeable movement patterns, while also respecting user-privacy. The importance of this information is even higher during the COVID-19 era. Last but not least, point clouds of ceilings can be used as reference to CAD and BIM models, in order to help the modelling of the existing utilities and their components in an indoor facility.

## 6. Conclusion and discussion

### 6.2.2. Limitations

The implemented pipeline showed promising results, however there are some limitations that are analysed in this subsection and could further improve the current implementation.

- **Data acquisition:** The data that was tested in this thesis included both the use of an iPad, as well as a typical Android device. Concerning point clouds, the **LiDAR** sensors currently exist in the recently released Apple devices as well as a few Android users in phones with a **ToF** camera and connection to the ARCore. Additionally, another bottleneck is the range of the currently existing **LiDAR** sensors which is approximately 5 meters, hence some very high ceilings cannot be acquired, without an additional extensible device. The current implementation only manipulates **PLY** files, therefore other file types cannot be used without a slight adjustment in the algorithm.
- **Pre-processing:** The algorithm that was developed for point cloud pre-processing, includes many parameters that were tweaked in order to achieve an optimal result. Parameters, such as the threshold that was set for plane segmentation, so that the flat part of the ceilings is separated from the protruding installations, might have to change for different datasets, so that an optimal result is produced. The same is valid for the case of images, where the strict ratio test in order to reduce false matches, has to be adjusted for different image sets. Furthermore, walls are partially but not totally excluded during the point cloud cleanup, a fact that might affect the indoor localisation results.
- **Indoor localisation:** The current implementation of the web-app returns the location of a user a few seconds after the point cloud is loaded. However, the first time someone is using the app, the algorithm checks, every available reference point cloud to match it with a user point cloud. Therefore, in large buildings with many different rooms, the time efficiency of the algorithm would be significantly worse for the first localisation. Another drawback is that the current pipeline is based on the installations that protrude from the ceilings, therefore the algorithm at its current state cannot be applied to simple ceilings without details, as the algorithm would have to be altered. In cases where a room is big and a user cannot easily navigate around it, the algorithm works better with parts of ceilings that have identifiable installations, so that they can be differentiated from other rooms. Similarly, the reference images of large rooms cannot show the whole ceiling, therefore in some cases the whole room cannot be represented from a single image.
- **Web - app:** The current product is rather an initial version of a web-application, therefore it is simplistic and does not include many capabilities. Currently users are only able to localise themselves based on point clouds, as the feature matching algorithm based on images is not incorporated in the application.
- **Location tracking:** Location tracking is currently implemented based on two or more indoor localisation results, while its visualisation is based on the indoor model and its included door openings. Therefore, the limitations of this operation are based on the respective ones concerning indoor localisation. The current implementation is based on a network graph of a single floor of the indoor facility, therefore location tracking does not currently work between different floors. However, that can be easily implemented by adding more floors in the indoor model and subsequently the network graph. A drawback is that the current dashboard, including statistics on different used paths, needs slight adjustments each time that it is used by a facility manager, in order to visualise the required information.

## 6.3. Recommendations and future work

This discussion also involves some recommendations for similar research that could help improve the current methodology, as well some proposed future research, that emerged from the current implementation.

### 6.3.1. Recommendations

A complete final product should be created in a way that both point clouds and images are included, so that the approach addresses a higher number of users. The indoor localisation pipeline that is based on point clouds and images of ceilings, should be tested in an entire indoor facility, in order to understand how it behaves in large scale environments. Additionally, a complete pipeline should be created and tested on various ceiling types, with less protruding installations, that are used in the current thesis. Furthermore, point clouds and images should be combined with a Wi-Fi fingerprinting approach, in order to reduce the search area during the first localisation of a user. Large rooms should be divided accordingly into smaller parts based on their size, so that the performance of indoor localisation based on point clouds and images of the ceilings is tested in a sub-room level. Alternative point cloud registration algorithms could be used, such as the 3D equivalent of SIFT, so that the point cloud and image-based localisation techniques can be directly compared. Regarding feature matching based on single image comparison, the reference images of large rooms, could include an orthomosaic, created from a collection of single images, so that each room is ideally represented. Feature matching techniques should be further analysed for their robustness while affected from different illumination changes. Additional testing could be implemented in Android devices with a ToF camera and ARCore, as well as image acquisition during different times of a day, so that the influence of lighting conditions is investigated.

### 6.3.2. Future work

#### Machine learning

Based on the current research that involves indoor localisation and location tracking, alternative machine learning algorithms that focus on plane detection can be used, in order to automatically detect the large wall planes that negatively affect the indoor localisation results based on ceilings. Feature matching of images based on monocular depth estimation techniques could be tested, as an alternative way to achieve indoor localisation based on images. Various indoor environment datasets can be found such as the DIODE dataset (<https://arxiv.org/abs/1908.00463>) to train the model and achieve localisation for different types of ceilings as well as different surfaces.

#### Landmark-based indoor localisation

Additional focus could be given to the protruding installations of the ceilings that are captured from the LiDAR sensor. The use of an AR platform, such as Google's ARCore, that would recognise the different utilities, by detecting key-points and flat surfaces, could be compared with

## 6. Conclusion and discussion

some respective reference data for each room. Based on these utility inventories for each room, a landmark-based approach could be developed for localisation and in extent navigation.

### **Indoor navigation**

Additional research could be incorporated regarding navigation, based on the indoor localisation results, focusing on different aspects, such as the establishment of the optimal navigational instructions in both natural and programming language, so that humans and robots can navigate in indoor environments. Furthermore, navigation could be implemented based on shortest path algorithms, so that the fastest routes are followed, but also by taking into account information, such as the least used paths, that could help users navigate between less occupied spaces. Aside from **LiDAR** and camera sensors, different sensors that are included in mobile phones, regarding temperature or humidity can be taken into account in a navigation algorithm, to optimise user experience in indoor facilities. For the same purpose, sensors included in a smart facility could also be exploited and combined with the aforementioned sensors of a mobile phone.

### **User groups**

A special target group could be people with partial or severe blindness. The incorporation of a real-time application for indoor localisation and navigation, that includes instructions in braille would be an interesting case study. Last but not least, an additional research could involve, indoor localisation and navigation applications that focus people with movement disorders, such as paraplegic or quadriplegic users, who have to follow specific paths as they navigate to a specific destination.

# A. Reproducibility and self-assessment

## A.1. Marks for each of the criteria

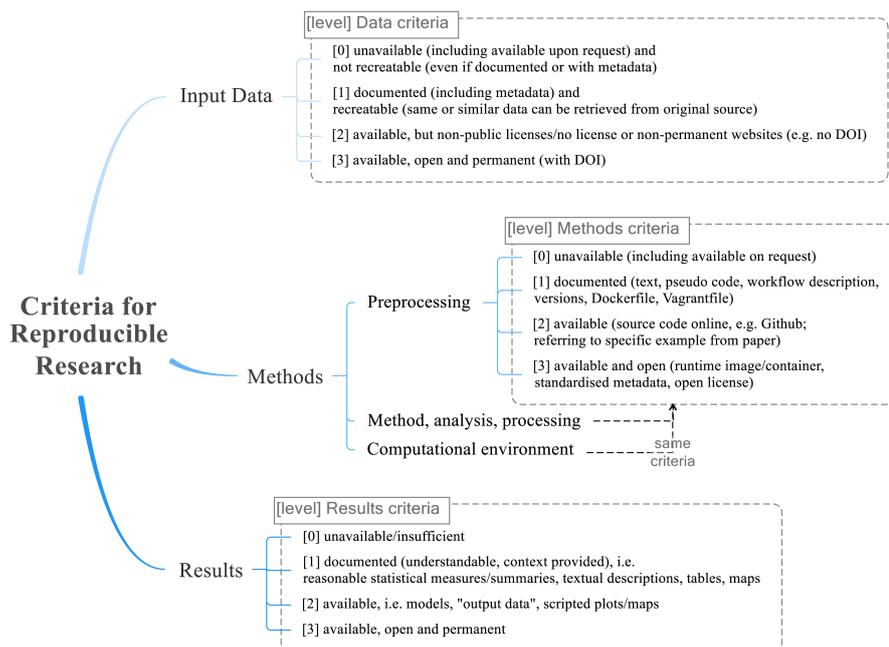


Figure A.1.: Reproducibility criteria to be assessed.

Grade/evaluate yourself for the 5 criteria (giving 0/1/2/3 for each):

Table A.1.: Evaluation of reproducibility criteria

| Criterion                 | Score |
|---------------------------|-------|
| Input data                | 2     |
| Pre-processing            | 3     |
| Methods                   | 2     |
| Computational environment | 2     |
| Results                   | 2     |

## A. *Reproducibility and self-assessment*

### A.2. Self-reflection

The self-assessment of this thesis is based on the criteria, that are explained in [Section A.1](#). Concerning input data, point clouds and images were acquired by the author and are available as open data in [https://github.com/jdardave/MSc\\_Geomatics\\_Thesis\\_Dardavesis.git](https://github.com/jdardave/MSc_Geomatics_Thesis_Dardavesis.git). Aside from that, additional data, such as the CAD files of the floorplans of the Faculty of Architecture and the Built Environment were specifically requested from TU Delft Real Estate, therefore are not open to the public.

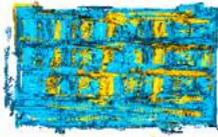
The pre-processing steps of this thesis were explained conceptually in [Section 3.3](#) and technically in [Section 4.3](#). The code including the implementation of these steps was written in Python and is openly available in [https://github.com/jdardave/MSc\\_Geomatics\\_Thesis\\_Dardavesis.git](https://github.com/jdardave/MSc_Geomatics_Thesis_Dardavesis.git). Therefore, these steps can be directly reproduced by different users.

For this thesis different programming languages (Python, [HTML](#) and JavaScript), as well as open source tools were used. Additionally most of the software that were used were open-source (CloudCompare, PhotoMatch, COLMAP), but the creation of the indoor model and network graph of the case study were implemented in ArcGIS pro, which is commercial and requires special license. Moreover, the current implementation of the web application works on a local level, as it does not run through a server.

Different results, that are presented and analysed in [Chapter 5](#) in the form of plots, tables, maps and statistics. Only a chosen part of the results is included in [Chapter 5](#), while the rest are included in [Appendix B](#). Due to the number and size of the results, such as plots of different point registration results, they are not available online.

## B. Additional results

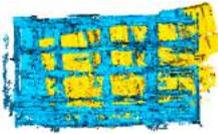
### B.1. Point clouds from SiteScape



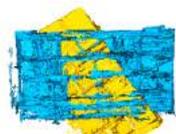
(a) Point-to-Point ICP (fitness=0.997, RMSE=0.06)



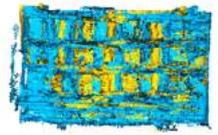
(b) Point-to-Point ICP (fitness=0.897, RMSE=0.21)



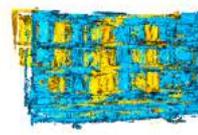
(a) Point-to-Plane ICP (fitness=0.89, RMSE=0.11)



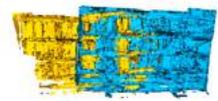
(b) Point-to-Plane ICP (fitness=0.808, RMSE=0.16)



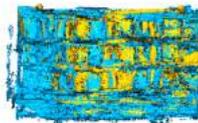
(a) Colored ICP (fitness=0.933, RMSE=0.11)



(b) Colored ICP (fitness=0.861, RMSE=0.11)



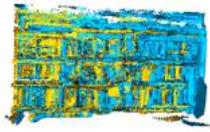
(a) Generalised ICP (fitness=0.59, RMSE=0.11)



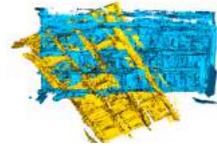
(b) Generalised ICP (fitness=0.995, RMSE=0.03)

Figure B.4.: RANSAC global registration (left) and fast global registration (right) of room 430, from point clouds acquired by SiteScape

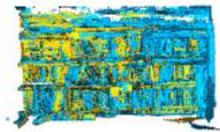
B. Additional results



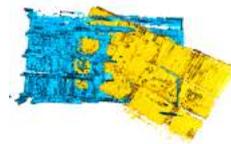
(a) Point-to-Point ICP (fitness=0.997, RMSE=0.03)



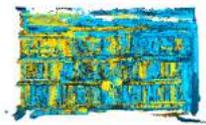
(b) Point-to-Point ICP (fitness=0.699, RMSE=0.14)



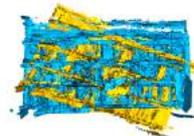
(a) Point-to-Plane ICP (fitness=0.996, RMSE=0.03)



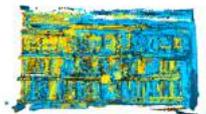
(b) Point-to-Point ICP (fitness=0.597, RMSE=0.21)



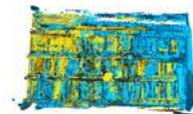
(a) Colored ICP (fitness=0.956, RMSE=0.07)



(b) Colored ICP (fitness=0.683, RMSE=0.108)



(a) Generalised ICP (fitness=0.998, RMSE=0.02)



(b) Generalised ICP (fitness=0.997, RMSE=0.02)

Figure B.8.: RANSAC global registration (left) and fast global registration (right) of room 470, from point clouds acquired by SiteScape

B.1. Point clouds from SiteScape



(a) Point-to-Point ICP (fitness=0.88, RMSE=0.11)



(b) Point-to-Point ICP (fitness=0.878, RMSE=0.104)



(a) Point-to-Plane ICP (fitness=0.612, RMSE=0.21)



(b) Point-to-Point ICP (fitness=0.995, RMSE=0.03)



(a) Colored ICP (fitness=0.845, RMSE=0.07)



(b) Colored ICP (fitness=0.827, RMSE=0.08)



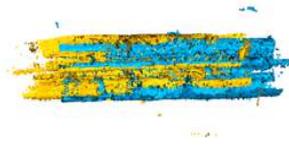
(a) Generalised ICP (fitness=0.931, RMSE=0.11)



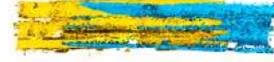
(b) Generalised ICP (fitness=0.987, RMSE=0.08)

Figure B.12.: RANSAC global registration (left) and fast global registration (right) of room 807, from point clouds acquired by SiteScape

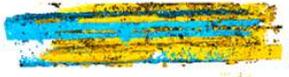
## B. Additional results



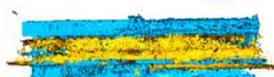
(a) Point-to-Point ICP (fitness=0.983, RMSE=0.11)



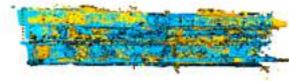
(b) Point-to-Point ICP (fitness=0.803, RMSE=0.19)



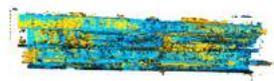
(a) Point-to-Plane ICP (fitness=0.932, RMSE=0.18)



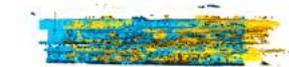
(b) Point-to-Point ICP (fitness=0.974, RMSE=0.17)



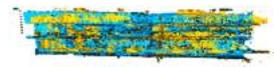
(a) Colored ICP (fitness=0.957, RMSE=0.07)



(b) Colored ICP (fitness=0.908, RMSE=0.08)



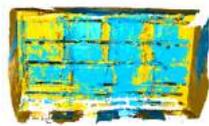
(a) Generalised ICP (fitness=0.987, RMSE=0.04)



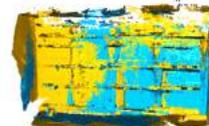
(b) Generalised ICP (fitness=0.99, RMSE=0.03)

Figure B.16.: RANSAC global registration (left) and fast global registration (right) of room 808, from point clouds acquired by SiteScape

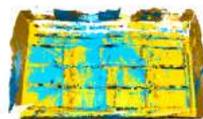
## B.2. Point clouds from Pix4D Catch



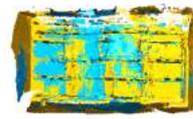
(a) Point-to-Plane ICP (fitness=0.761, RMSE=0.16)



(b) Point-to-Point ICP (fitness=0.732, RMSE=0.17)

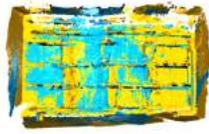


(a) Point-to-Plane ICP (fitness=0.743, RMSE=0.16)

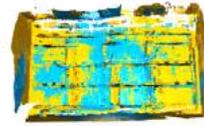


(b) Point-to-Point ICP (fitness=0.743, RMSE=0.16)

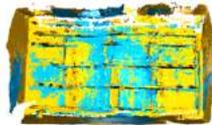
B.2. Point clouds from Pix4D Catch



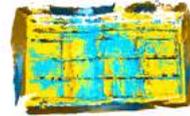
(a) Colored ICP (fitness=0.726, RMSE=0.06)



(b) Colored ICP (fitness=0.727, RMSE=0.07)

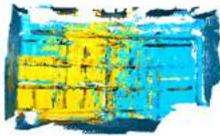


(a) Generalised ICP (fitness=0.943, RMSE=0.11)

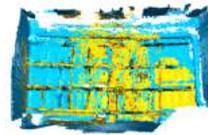


(b) Generalised ICP (fitness=0.93, RMSE=0.11)

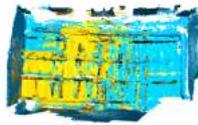
Figure B.20.: RANSAC global registration (left) and fast global registration (right) of room 430, from point clouds acquired by Pix4D Catch



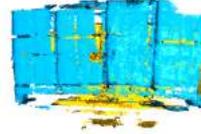
(a) Point-to-Point ICP (fitness=0.981, RMSE=0.11)



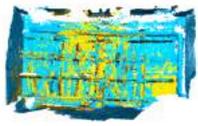
(b) Point-to-Point ICP (fitness=0.976, RMSE=0.12)



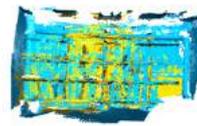
(a) Point-to-Plane ICP (fitness=0.979, RMSE=0.11)



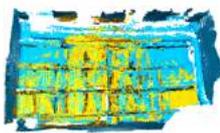
(b) Point-to-Plane ICP (fitness=0.693, RMSE=0.19)



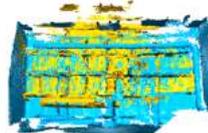
(a) Colored ICP (fitness=0.944, RMSE=0.09)



(b) Colored ICP (fitness=0.941, RMSE=0.09)



(a) Generalised ICP (fitness=0.998, RMSE=0.05)



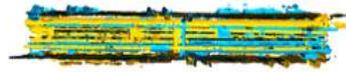
(b) Generalised ICP (fitness=0.902, RMSE=0.11)

Figure B.24.: RANSAC global registration (left) and fast global registration (right) of room 470, from point clouds acquired by Pix4D Catch

B. Additional results



(a) Point-to-Point ICP (fitness=0.663, RMSE=0.17)



(b) Point-to-Point ICP (fitness=0.773, RMSE=0.12)



(a) Point-to-Plane ICP (fitness=0.994, RMSE=0.12)



(b) Point-to-Point ICP (fitness=0.773, RMSE=0.21)



(a) Colored ICP (fitness=0.932, RMSE=0.08)



(b) Colored ICP (fitness=0.912, RMSE=0.13)



(a) Generalised ICP (fitness=0.992, RMSE=0.05)

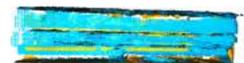


(b) Generalised ICP (fitness=0.56, RMSE=0.24)

Figure B.28.: RANSAC global registration (left) and fast global registration (right) of room 807, from point clouds acquired by Pix4D Catch



(a) Point-to-Point ICP (fitness=0.963, RMSE=0.11)

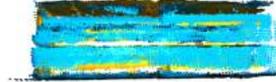


(b) Point-to-Point ICP (fitness=0.821, RMSE=0.19)

B.2. Point clouds from Pix4D Catch



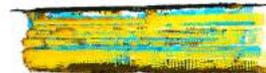
(a) Point-to-Plane ICP (fitness=0.971, RMSE=0.11)



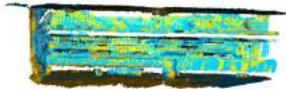
(b) Point-to-Point ICP (fitness=0.813, RMSE=0.19)



(a) Colored ICP (fitness=0.942, RMSE=0.08)



(b) Colored ICP (fitness=0.527, RMSE=0.09)



(a) Generalised ICP (fitness=0.943, RMSE=0.15)

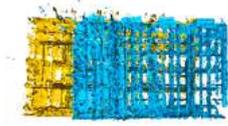


(b) Generalised ICP (fitness=0.812, RMSE=0.18)

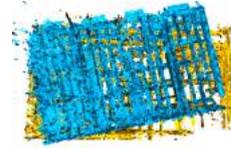
Figure B.32.: RANSAC global registration (left) and fast global registration (right) of room 808, from point clouds acquired by Pix4D Catch

B. Additional results

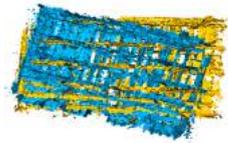
**B.3. Point clouds that are reconstructed from overlapping images**



(a) Point-to-Point ICP (fitness=0.642, RMSE=0.21)



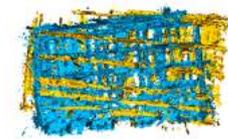
(b) Point-to-Point ICP (fitness=0.701, RMSE=0.22)



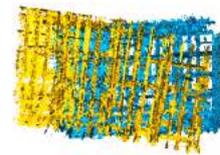
(a) Point-to-Plane ICP (fitness=0.672, RMSE=0.24)



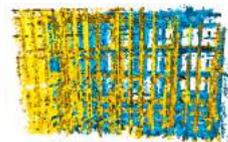
(b) Point-to-Point ICP (fitness=0.821, RMSE=0.18)



(a) Colored ICP (fitness=0.694, RMSE=0.19)



(b) Colored ICP (fitness=0.454, RMSE=0.13)

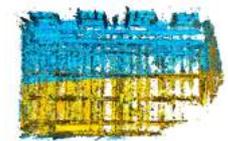


(a) Generalised ICP (fitness=0.571, RMSE=0.22)



(b) Generalised ICP (fitness=0.882, RMSE=0.18)

Figure B.36.: RANSAC global registration (left) and fast global registration (right) of room 430 from reconstructed images of ceilings



(a) Point-to-Point ICP (fitness=0.985, RMSE=0.07)



(b) Point-to-Point ICP (fitness=0.991, RMSE=0.05)

B.3. Point clouds that are reconstructed from overlapping images



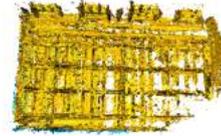
(a) Point-to-Plane ICP (fitness=0.993, RMSE=0.05)



(b) Point-to-Point ICP (fitness=0.991, RMSE=0.07)



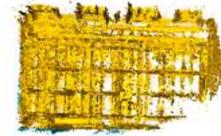
(a) Colored ICP (fitness=0.994, RMSE=0.05)



(b) Colored ICP (fitness=0.993, RMSE=0.07)

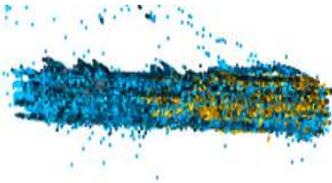


(a) Generalised ICP (fitness=0.993, RMSE=0.05)

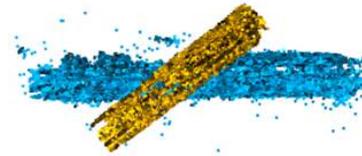


(b) Generalised ICP (fitness=0.991, RMSE=0.08)

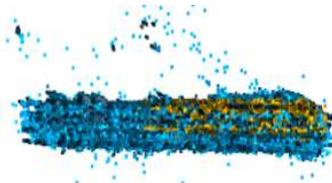
Figure B.40.: RANSAC global registration (left) and fast global registration (right) of room 470 from reconstructed images of ceilings



(a) Point-to-Plane ICP (fitness=0.991, RMSE=0.12)



(b) Point-to-Point ICP (fitness=0.662, RMSE=0.24)

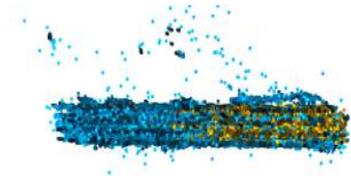


(a) Point-to-Plane ICP (fitness=0.981, RMSE=0.14)



(b) Point-to-Point ICP (fitness=0.982, RMSE=0.16)

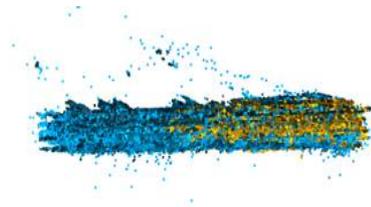
B. Additional results



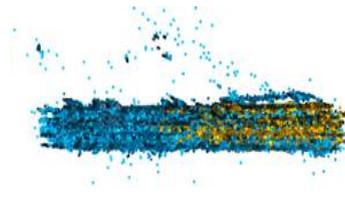
(a) Colored ICP (fitness=0.934, RMSE=0.09)



(b) Colored ICP (fitness=0.532, RMSE=0.12)

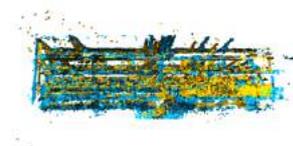


(a) Generalised ICP (fitness=0.991, RMSE=0.02)



(b) Generalised ICP (fitness=0.993, RMSE=0.03)

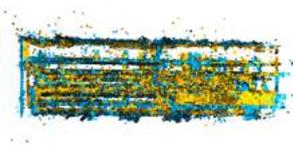
Figure B.44.: RANSAC global registration (left) and fast global registration (right) of room 807 from reconstructed images of ceilings



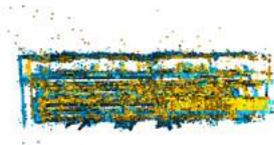
(a) Point-to-Point ICP (fitness=0.961, RMSE=0.17)



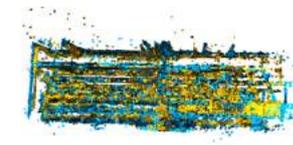
(b) Point-to-Point ICP (fitness=0.972, RMSE=0.16)



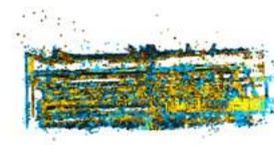
(a) Point-to-Plane ICP (fitness=0.972, RMSE=0.16)



(b) Point-to-Plane ICP (fitness=0.951, RMSE=0.16)



(a) Colored ICP (fitness=0.793, RMSE=0.11)



(b) Colored ICP (fitness=0.812, RMSE=0.12)

#### B.4. Accuracy/precision metrics

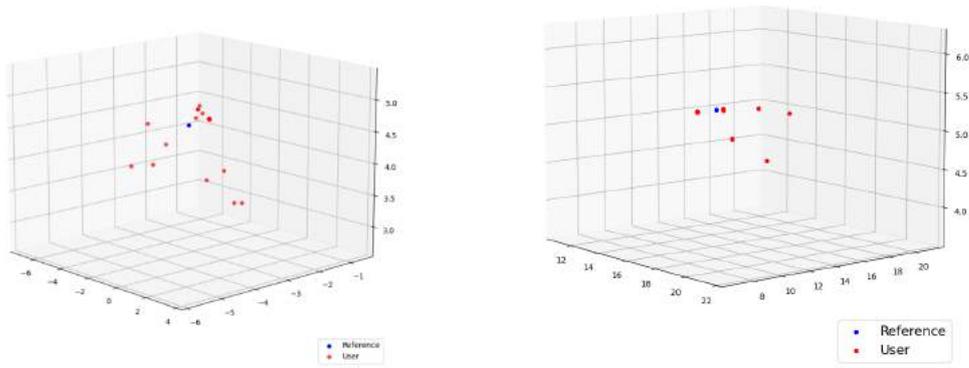


(a) Generalised ICP (fitness=0.993, RMSE=0.09)

(b) Generalised ICP (fitness=0.992, RMSE=0.09)

Figure B.48.: RANSAC global registration (left) and fast global registration (right) of room 808 from reconstructed images of ceilings

#### B.4. Accuracy/precision metrics



(a) Scatter plot with centers of reference and user point clouds after point cloud matching in room 08.02.00.430

(b) Scatter plot with centers of reference and user point clouds after point cloud matching in room 08.02.00.470

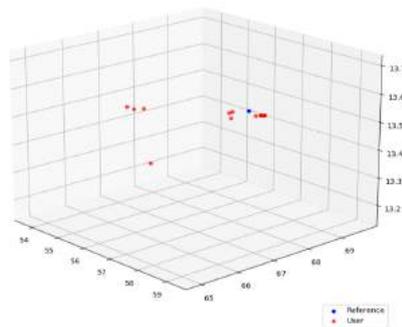
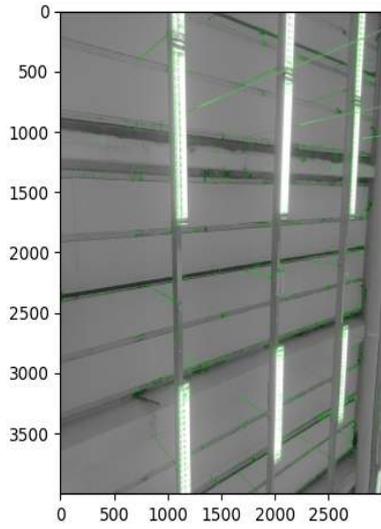


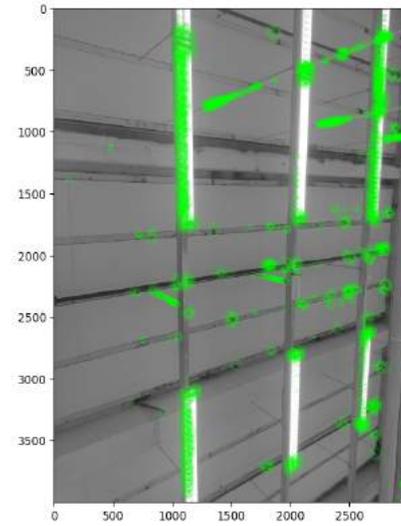
Figure B.50.: Scatter plot with centers of reference and user point clouds after point cloud matching in room 08.02.00.560

B. Additional results

**B.5. Feature detection in images**

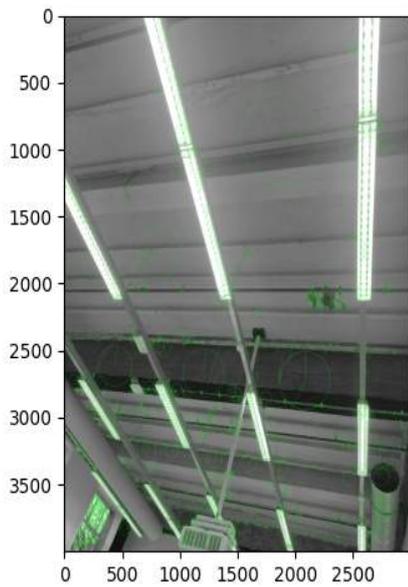


(a) Feature detection with SIFT

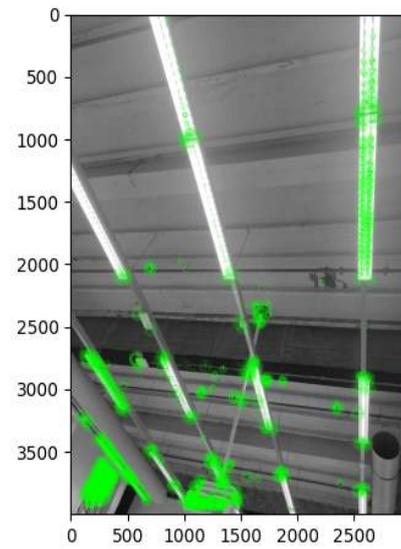


(b) Feature detection with ORB

Figure B.51.: Feature detection in room 08.02.00.430



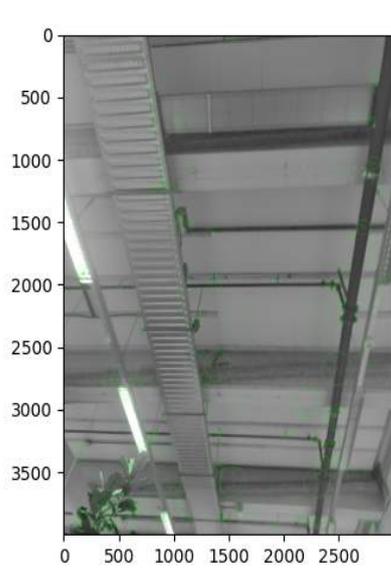
(a) Feature detection with SIFT



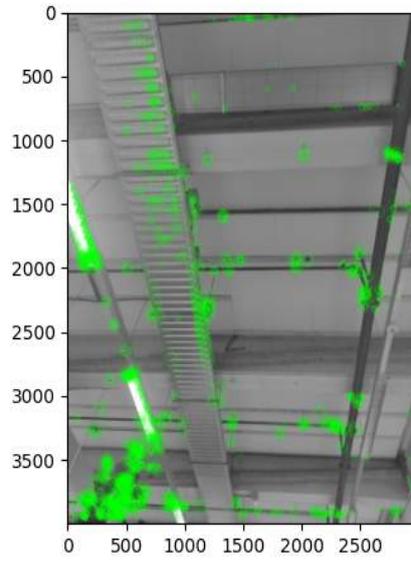
(b) Feature detection with ORB

Figure B.52.: Feature detection in room 08.02.00.470

B.5. Feature detection in images

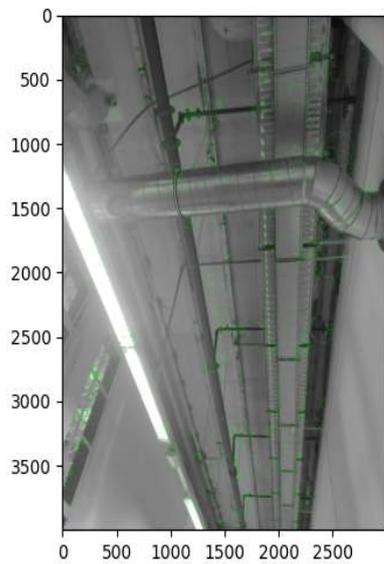


(a) Feature detection with SIFT

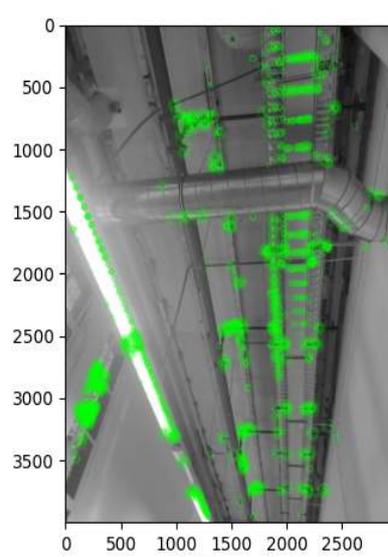


(b) Feature detection with ORB

Figure B.53.: Feature detection in room 08.02.00.560



(a) Feature detection with SIFT



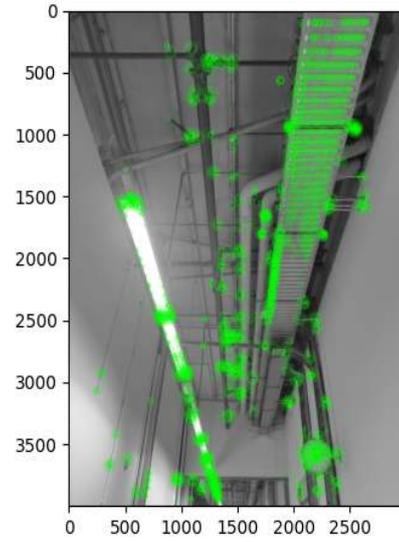
(b) Feature detection with ORB

Figure B.54.: Feature detection in room 08.02.00.807

*B. Additional results*



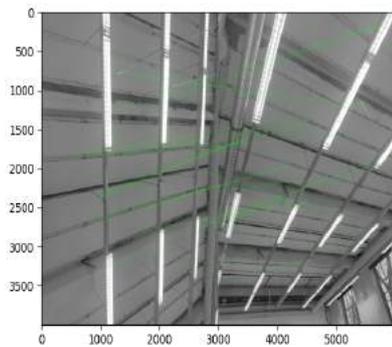
(a) Feature detection with **SIFT**



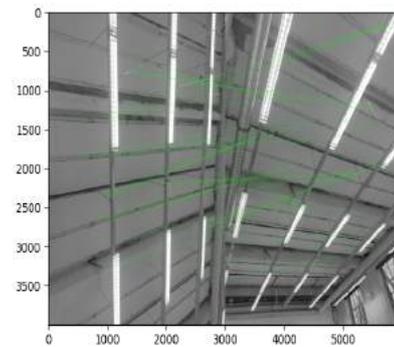
(b) Feature detection with **ORB**

Figure B.55.: Feature detection in room 08.02.00.808

**B.6. Feature matching in images**

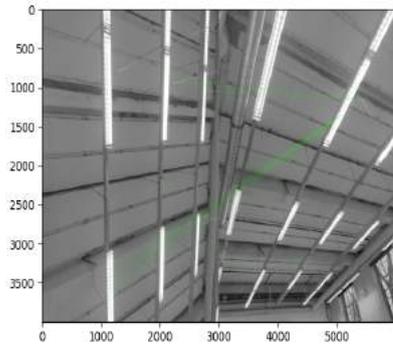


(a) Feature detection with **SIFT** and feature matching with brute force

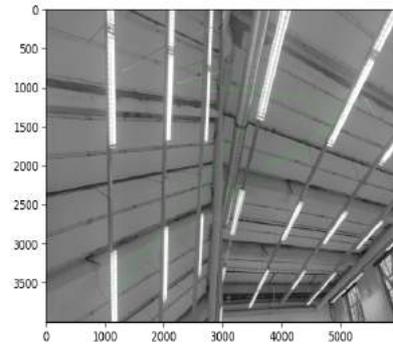


(b) Feature detection with **SIFT** and feature matching with **FLANN**

B.6. Feature matching in images

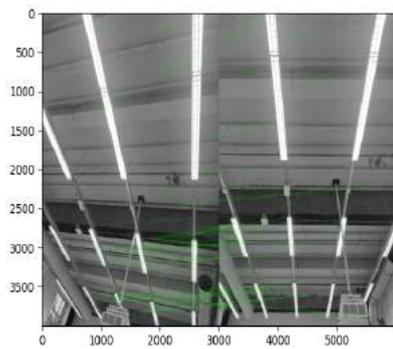


(a) Feature detection with **ORB** and feature matching with brute force

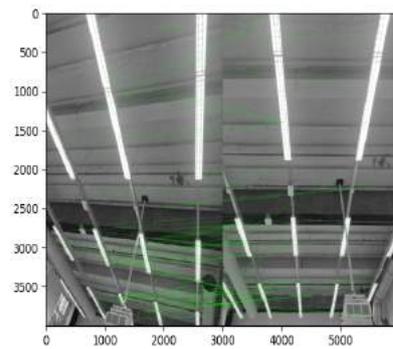


(b) Feature detection with **ORB** and feature matching with **FLANN**

Figure B.57.: Feature matching in room 08.02.00.430

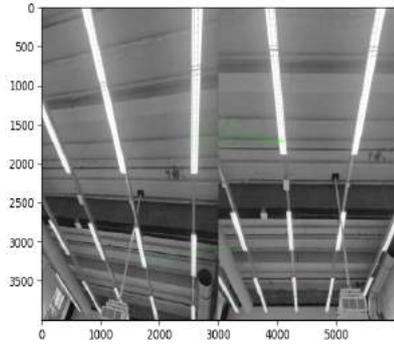


(a) Feature detection with **SIFT** and feature matching with brute force

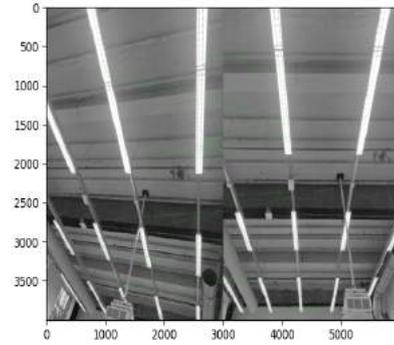


(b) Feature detection with **SIFT** and feature matching with **FLANN**

B. Additional results

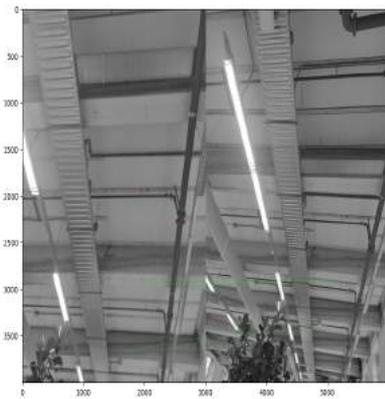


(a) Feature detection with **ORB** and feature matching with brute force

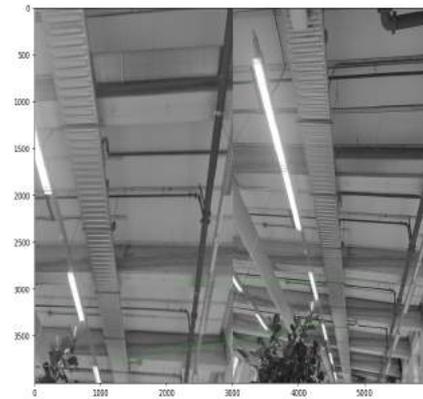


(b) Feature detection with **ORB** and feature matching with **FLANN**

Figure B.59.: Feature matching in room 08.02.00.470

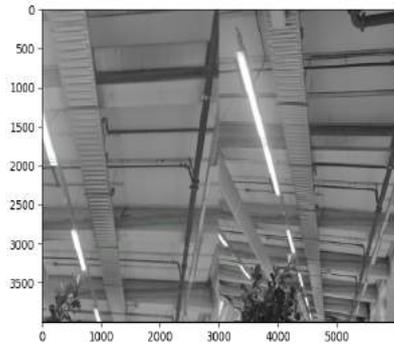


(a) Feature detection with **SIFT** and feature matching with brute force

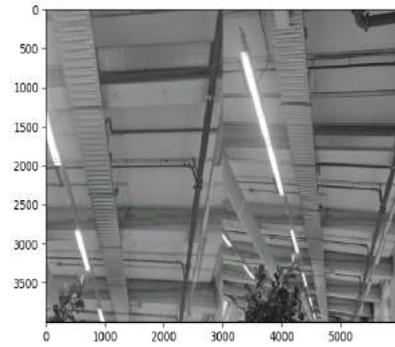


(b) Feature detection with **SIFT** and feature matching with **FLANN**

B.6. Feature matching in images

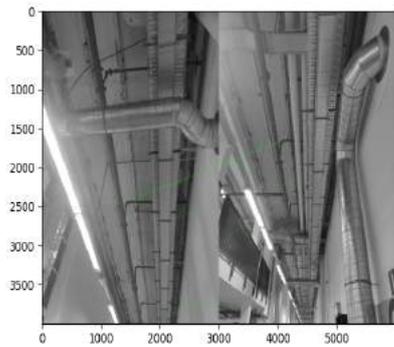


(a) Feature detection with **ORB** and feature matching with brute force

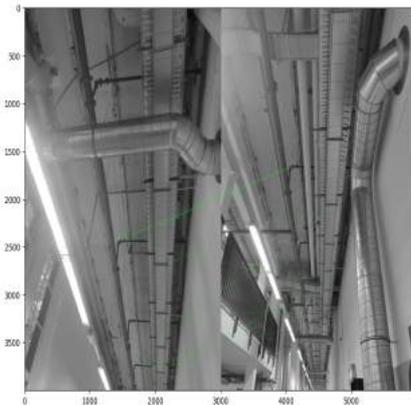


(b) Feature detection with **ORB** and feature matching with **FLANN**

Figure B.61.: Feature matching in room 08.02.00.560

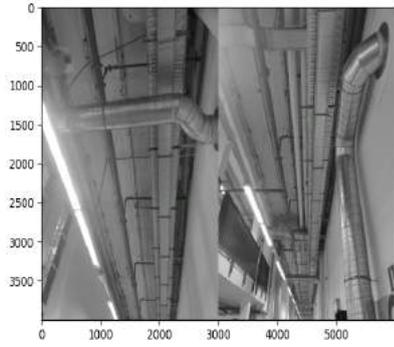


(a) Feature detection with **SIFT** and feature matching with brute force

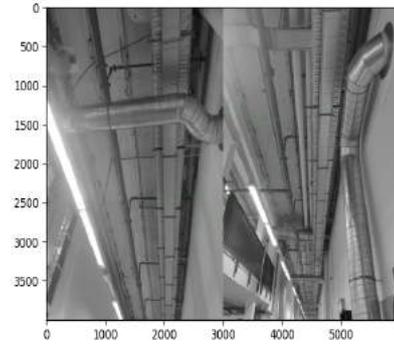


(b) Feature detection with **SIFT** and feature matching with **FLANN**

B. Additional results

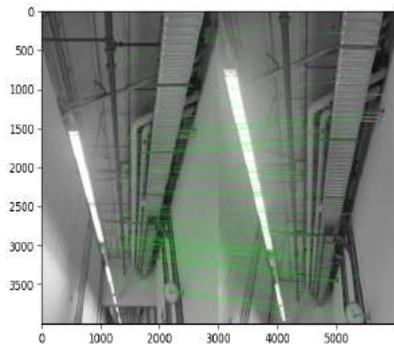


(a) Feature detection with **ORB** and feature matching with brute force

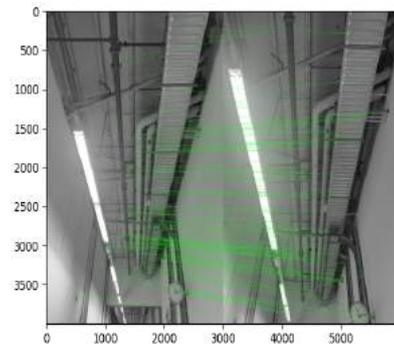


(b) Feature detection with **ORB** and feature matching with **FLANN**

Figure B.63.: Feature matching in room 08.02.00.807

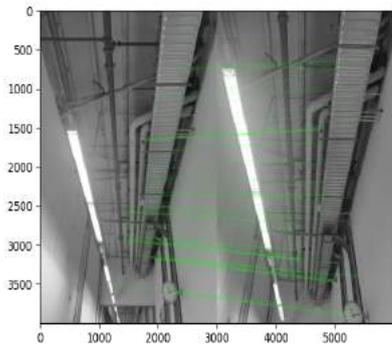


(a) Feature detection with **SIFT** and feature matching with brute force

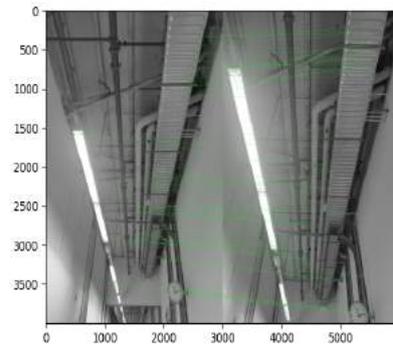


(b) Feature detection with **SIFT** and feature matching with **FLANN**

B.6. Feature matching in images



(a) Feature detection with **ORB** and feature matching with brute force



(b) Feature detection with **ORB** and feature matching with **FLANN**

Figure B.65.: Feature matching in room 08.02.00.808



# Bibliography

- H. Bay, T. Tuytelaars, and L. V. Gool. Surf: Speeded up robust features. In *European conference on computer vision*, pages 404–417. Springer, 2006. URL [https://link.springer.com/chapter/10.1007/11744023\\_32](https://link.springer.com/chapter/10.1007/11744023_32).
- P. Besl and N. D. McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992. doi: 10.1109/34.121791. URL <https://ieeexplore.ieee.org/document/121791>.
- A. Bose and C. H. Foh. A practical path loss model for indoor wifi positioning enhancement. In *2007 6th International Conference on Information, Communications & Signal Processing*, pages 1–5. IEEE, 2007. URL <https://ieeexplore.ieee.org/document/4449717?arnumber=4449717>.
- Y. Chen and G. G. Medioni. Object modelling by registration of multiple range images. *Image Vis. Comput.*, 10:145–155, 1992. URL <https://www.sciencedirect.com/science/article/abs/pii/026288569290066C>.
- C.-Y. Chow, M. F. Mokbel, and X. Liu. A peer-to-peer spatial cloaking algorithm for anonymous location-based service. In *Proceedings of the 14th annual ACM international symposium on Advances in geographic information systems*, pages 171–178, 2006. URL [https://www.researchgate.net/publication/221589528\\_A\\_Peer-to-Peer\\_Spatial\\_Cloaking\\_Algorithm\\_for\\_Anonymous\\_Location-based\\_Service](https://www.researchgate.net/publication/221589528_A_Peer-to-Peer_Spatial_Cloaking_Algorithm_for_Anonymous_Location-based_Service).
- L. Díaz Vilariño, H. Tran, E. Frías, J. Balado Frias, and K. Khoshelham. 3d mapping of indoor and outdoor environments using apple smart devices. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLIII-B4-2022:303–308, 06 2022. doi: 10.5194/isprs-archives-XLIII-B4-2022-303-2022. URL [https://www.researchgate.net/publication/361066256\\_3D\\_MAPPING\\_OF\\_INDOOR\\_AND\\_OUTDOOR\\_ENVIRONMENTS\\_USING\\_APPLE\\_SMART\\_DEVICES](https://www.researchgate.net/publication/361066256_3D_MAPPING_OF_INDOOR_AND_OUTDOOR_ENVIRONMENTS_USING_APPLE_SMART_DEVICES).
- D. Forsyth and J. Ponce. *Computer vision: A modern approach*. Prentice hall, 2011.
- C. Fratzeskou, C. Garg, K. Staring, M. Deng, C. Jansen, E. Verbree, and M. Meijers. Indoor localisation based on point clouds of the ceiling: Syntheses project 2019. 2019. URL <https://repository.tudelft.nl/islandora/object/uuid%3A1ae057a7-8974-4b60-88b5-b00d164619c4>.
- S. Garcia-Villalonga and A. Perez-Navarro. Influence of human absorption of wi-fi signal in indoor positioning with wi-fi fingerprinting. In *2015 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–10, 2015. doi: 10.1109/IPIN.2015.7346778. URL <https://ieeexplore.ieee.org/document/7346778>.
- D. González-Aguilera, E. Oña, L. López Fernández, E. Farella, E. Stathopoulou, I. Toschi, F. Remondino, P. Rodríguez-Gonzálvez, D. Hernandez, A. Fusiello, and F. Nex. Photomatch: An open-source multi-view and multi-modal feature matching tool for

## Bibliography

- photogrammetric applications. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLIII-B5-2020:213–219, 08 2020. doi: 10.5194/isprs-archives-XLIII-B5-2020-213-2020. URL [https://www.researchgate.net/publication/343854813\\_PHOTOMATCH\\_AN\\_OPEN-SOURCE\\_MULTI-VIEW\\_AND\\_MODAL-FEATURE\\_MATCHING\\_TOOL\\_FOR\\_PHOTOGAMMETRIC\\_APPLICATIONS](https://www.researchgate.net/publication/343854813_PHOTOMATCH_AN_OPEN-SOURCE_MULTI-VIEW_AND_MODAL-FEATURE_MATCHING_TOOL_FOR_PHOTOGAMMETRIC_APPLICATIONS).
- P. Groves. *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems, Second Edition*. 03 2013.
- G. Gupta, N. Kejriwal, P. Pallav, E. Hassan, S. Kumar, and R. Hebbalaguppe. Indoor localisation and navigation on augmented reality devices. In *2016 IEEE International Symposium on Mixed and Augmented Reality (ISMAR-Adjunct)*, pages 107–112, 2016. doi: 10.1109/ISMAR-Adjunct.2016.0052. URL [https://www.researchgate.net/profile/Ramya-Hebbalaguppe/publication/311439001\\_Indoor\\_Localisation\\_and\\_Navigation\\_on\\_Augmented\\_Reality\\_Devices/links/584673f708aeda69681cc7b9/Indoor-Localisation-and-Navigation-on-Augmented-Reality-Devices.pdf](https://www.researchgate.net/profile/Ramya-Hebbalaguppe/publication/311439001_Indoor_Localisation_and_Navigation_on_Augmented_Reality_Devices/links/584673f708aeda69681cc7b9/Indoor-Localisation-and-Navigation-on-Augmented-Reality-Devices.pdf).
- X.-F. Han, J. Jin, M.-J. Wang, W. Jiang, L. Gao, and L. Xiao. A review of algorithms for filtering the 3d point cloud. *Signal Processing: Image Communication*, 57, 05 2017. doi: 10.1016/j.image.2017.05.009. URL <https://www.sciencedirect.com/science/article/abs/pii/S0923596517300930>.
- X.-F. Han, J. Jin, J. Xie, M.-J. Wang, and W. Jiang. A comprehensive review of 3d point cloud descriptors. 02 2018. URL [https://www.researchgate.net/publication/323003836\\_A\\_comprehensive\\_review\\_of\\_3D\\_point\\_cloud\\_descriptors](https://www.researchgate.net/publication/323003836_A_comprehensive_review_of_3D_point_cloud_descriptors).
- R. I. Hartley and J. L. Mundy. Relationship between photogrammetry and computer vision. *Integrating photogrammetric techniques with scene analysis and machine vision*, 1944:92–105, 1993. URL <https://www.spiedigitallibrary.org/conference-proceedings-of-spie/1944/0000/Relationship-between-photogrammetry-and-computer-vision/10.1117/12.155818.short?SS0=1>.
- R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.
- K. Hata and S. Savarese. Course notes of stanford, course cs231a, 2022. URL [https://web.stanford.edu/class/cs231a/course\\_notes.html](https://web.stanford.edu/class/cs231a/course_notes.html).
- H. Hile and G. Borriello. Positioning and orientation in indoor environments using camera phones. *IEEE Computer Graphics and Applications*, 28:32–39, 7 2008. ISSN 02721716. doi: 10.1109/MCG.2008.80. URL [https://www.researchgate.net/publication/3210676\\_Positioning\\_and\\_Orientation\\_in\\_Indoor\\_Environments\\_Using\\_Camera\\_Phones](https://www.researchgate.net/publication/3210676_Positioning_and_Orientation_in_Indoor_Environments_Using_Camera_Phones).
- V. Honkavirta, T. Perala, S. Ali-Loytty, and R. Piché. A comparative survey of wlan location fingerprinting methods. In *2009 6th workshop on positioning, navigation and communication*, pages 243–251. IEEE, 2009. URL <https://ieeexplore.ieee.org/document/4907834>.
- A. Jakubović and J. Velagić. Image feature matching and object detection using brute-force matchers. In *2018 International Symposium ELMAR*, pages 83–86, 2018. doi: 10.23919/ELMAR.2018.8534641. URL [https://www.researchgate.net/publication/328991586\\_Image\\_Feature\\_Matching\\_and\\_Object\\_Detection\\_Using\\_Brute-Force\\_Matchers](https://www.researchgate.net/publication/328991586_Image_Feature_Matching_and_Object_Detection_Using_Brute-Force_Matchers).
- M. Jiang, Y. Wu, T. Zhao, Z. Zhao, and C. Lu. Pointsift: A sift-like network module for 3d point cloud semantic segmentation, 2018. URL <https://arxiv.org/abs/1807.00652>.

- A. R. Jiménez, F. Seco, J. C. Prieto, and J. Guevara. Indoor pedestrian navigation using an ins/ekf framework for yaw drift reduction and a foot-mounted imu. In *2010 7th workshop on positioning, navigation and communication*, pages 135–143. IEEE, 2010. URL <https://ieeexplore.ieee.org/document/5649300>.
- K. Kaemarungsi and P. Krishnamurthy. Modeling of indoor positioning systems based on location fingerprinting. In *Ieee Infocom 2004*, volume 2, pages 1012–1022. IEEE, 2004. URL <http://d-scholarship.pitt.edu/6395/1/dissertation28Feb05.pdf>.
- P. Kaniouras, M. MOSCHOLAKI, J. van Liempt, K. Jarocki, L. Zhang, E. Verbee, and M. Meijers. Direct analysis on point clouds: Geomatics syntesis project 2019. 2019. URL <https://repository.tudelft.nl/islandora/object/uuid%3A44f610e66-01b5-409a-aaf9-9f03d96c7889>.
- E. Kaplan and C. Hegarty. *Understanding GPS/GNSS: Principles and Applications, Third Edition*. 2017.
- H. Kido, Y. Yanagisawa, and T. Satoh. An anonymous communication technique using dummies for location-based services. In *ICPS'05. Proceedings. International Conference on Pervasive Services, 2005.*, pages 88–97. IEEE, 2005. URL <https://ieeexplore.ieee.org/document/1506394>.
- S.-C. Kim, Y.-S. Jeong, and S.-O. Park. Rfid-based indoor location tracking to ensure the safety of the elderly in smart home environments. *Personal and ubiquitous computing*, 17(8):1699–1707, 2013. URL <https://link.springer.com/article/10.1007/s00779-012-0604-4>.
- S. K. Kim, S.-J. Kang, Y.-J. Choi, M.-H. Choi, and M. H. and. Augmented-reality survey: from concept to application. *KSII Transactions on Internet and Information Systems*, 11(2):982–1004, February 2017. doi: 10.3837/tiis.2017.02.019. URL [https://www.researchgate.net/publication/316646460\\_Augmented-Reality\\_Survey\\_from\\_Concept\\_to\\_Application](https://www.researchgate.net/publication/316646460_Augmented-Reality_Survey_from_Concept_to_Application).
- J. Koch, J. Wettach, E. Bloch, and K. Berns. Indoor localisation of humans, objects, and mobile robots with rfid infrastructure. In *7th International Conference on Hybrid Intelligent Systems (HIS 2007)*, pages 271–276, 2007. doi: 10.1109/HIS.2007.25. URL <https://ieeexplore.ieee.org/document/4344063>.
- A. Konstantinidis, G. Chatzimilioudis, D. Zeinalipour-Yazti, P. Mpeis, N. Pelekis, and Y. Theodoridis. Privacy-preserving indoor localization on smartphones. *IEEE Transactions on Knowledge and Data Engineering*, 27(11):3042–3055, 2015. URL <https://ieeexplore.ieee.org/document/7118199>.
- H. Koyuncu and S. H. Yang. A survey of indoor positioning and object locating systems. *IJCSNS International Journal of Computer Science and Network Security*, 10(5):121–128, 2010. URL [https://www.researchgate.net/profile/Hakan-Koyuncu-3/publication/267956851\\_A\\_Survey\\_of\\_Indoor\\_Positioning\\_and\\_Object\\_Locating\\_Systems/links/5e5b7beca6fdccbeba0f3b97/A-Survey-of-Indoor-Positioning-and-Object-Locating-Systems.pdf](https://www.researchgate.net/profile/Hakan-Koyuncu-3/publication/267956851_A_Survey_of_Indoor_Positioning_and_Object_Locating_Systems/links/5e5b7beca6fdccbeba0f3b97/A-Survey-of-Indoor-Positioning-and-Object-Locating-Systems.pdf).
- J. Li, Q. Hu, and M. Ai. Point cloud registration based on one-point ransac and scale-annealing biweight estimation. *IEEE Transactions on Geoscience and Remote Sensing*, PP:1–14, 01 2021. doi: 10.1109/TGRS.2020.3045456. URL <https://ieeexplore.ieee.org/document/9318535>.

## Bibliography

- H. Liu, H. Darabi, P. Banerjee, and J. Liu. Survey of wireless indoor positioning techniques and systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 37(6):1067–1080, 2007. URL <https://ieeexplore.ieee.org/document/4343996>.
- H. C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135, 1981. URL <https://www.sciencedirect.com/science/article/pii/B978008051581650012X>.
- D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004. URL <https://link.springer.com/article/10.1023/B:VISI.0000029664.99615.94>.
- D. Lymberopoulos, J. Liu, X. Yang, R. R. Choudhury, V. Handziski, and S. Sen. A realistic evaluation and comparison of indoor location technologies: Experiences and lessons learned. In *Proceedings of the 14th International Conference on Information Processing in Sensor Networks, IPSN '15*, page 178–189, New York, NY, USA, 2015. Association for Computing Machinery. ISBN 9781450334754. doi: 10.1145/2737095.2737726. URL <https://doi.org/10.1145/2737095.2737726>.
- R. Mautz. Indoor positioning technologies. 2012. URL <https://www.research-collection.ethz.ch/bitstream/handle/20.500.11850/54888/eth-5659-01.pdf>.
- P.-E. Michon and M. Denis. When and why are visual landmarks used in giving directions? In D. R. Montello, editor, *Spatial Information Theory*, pages 292–305, Berlin, Heidelberg, 2001. Springer Berlin Heidelberg. ISBN 978-3-540-45424-3. URL [https://link.springer.com/chapter/10.1007/3-540-45424-1\\_20](https://link.springer.com/chapter/10.1007/3-540-45424-1_20).
- M. Miknis, J. Ware, R. Davies, and P. Plassmann. Efficient point cloud pre-processing using the point cloud library. *International Journal of Image Processing*, 10(2):63–72, June 2016. ISSN 1985-2304. URL <https://www.cscjournals.org/manuscript/Journals/IJIP/Volume10/Issue2/IJIP-1063.pdf>.
- S. A. Mitilineos, D. M. Kyriazanos, O. E. Segou, J. N. Goufas, and S. C. A. Thomopoulos. Indoor localization with wireless sensor networks. 2010. URL <https://www.semanticscholar.org/paper/INDOOR-LOCALIZATION-WITH-WIRELESS-SENSOR-NETWORKS-Mitilineos-Kyriazanos/c0c3165d6094a66e637603e3b274026e8413b5dd>.
- R. Mohr, L. Quan, and F. Veillon. Relative 3d reconstruction using multiple uncalibrated images. *The International Journal of Robotics Research*, 14(6):619–632, 1995. URL [https://www.researchgate.net/publication/220122743\\_Relative\\_3D\\_Reconstruction\\_Using\\_Multiple\\_Uncalibrated\\_Images](https://www.researchgate.net/publication/220122743_Relative_3D_Reconstruction_Using_Multiple_Uncalibrated_Images).
- M. Muja and D. G. Lowe. Flann, fast library for approximate nearest neighbors. In *International Conference on Computer Vision Theory and Applications (VISAPP'09)*, volume 3. INSTICC Press, 2009. URL <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.192.5378&rep=rep1&type=pdf>.
- L. Oostwegel. Indoor positioning using augmented reality. 2020. URL <https://repository.tudelft.nl/islandora/object/uuid%3Ab50e234d-3fe0-44a4-9ec0-f5664ca79844>.
- J. Park, Q.-Y. Zhou, and V. Koltun. Colored point cloud registration revisited. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 143–152, 2017. URL <https://ieeexplore.ieee.org/document/8237287>.

- G. Ping, M. A. Esfahani, and H. Wang. Visual enhanced 3d point cloud reconstruction from a single image. *arXiv preprint arXiv:2108.07685*, 2021. URL <https://arxiv.org/abs/2108.07685>.
- A. Pérez-Navarro, J. Torres-Sospedra, R. Montoliu, J. Conesa, R. Berkvens, G. Caso, C. Costa, N. Dorigatti, N. Hernández, S. Knauth, E. S. Lohan, J. Machaj, A. Moreira, and P. Wilk. 1 - challenges of fingerprinting in indoor positioning and navigation. In J. Conesa, A. Pérez-Navarro, J. Torres-Sospedra, and R. Montoliu, editors, *Geographical and Fingerprinting Data to Create Systems for Indoor Positioning and Indoor/Outdoor Navigation*, Intelligent Data-Centric Systems, pages 1–20. Academic Press, 2019. ISBN 978-0-12-813189-3. doi: <https://doi.org/10.1016/B978-0-12-813189-3.00001-0>. URL <https://www.sciencedirect.com/science/article/pii/B9780128131893000010>.
- E. Rosten and T. Drummond. Machine learning for high-speed corner detection. In *European conference on computer vision*, pages 430–443. Springer, 2006. URL [https://link.springer.com/chapter/10.1007/11744023\\_34](https://link.springer.com/chapter/10.1007/11744023_34).
- S. Royo and M. Ballesta-Garcia. An overview of lidar imaging systems for autonomous vehicles. *Applied Sciences*, 9(19), 2019. ISSN 2076-3417. doi: 10.3390/app9194093. URL <https://www.mdpi.com/2076-3417/9/19/4093>.
- E. Rublee, V. Rabaud, K. Konolige, and G. Bradski. Orb: An efficient alternative to sift or surf. In *2011 International Conference on Computer Vision*, pages 2564–2571, 2011. doi: 10.1109/ICCV.2011.6126544. URL <https://ieeexplore.ieee.org/document/6126544>.
- R. B. Rusu. *Semantic 3D Object Maps for Everyday Manipulation in Human Living Environments*. PhD thesis, Computer Science department, Technische Universitaet Muenchen, Germany, October 2009. URL <https://link.springer.com/article/10.1007/s13218-010-0059-6>.
- R. B. Rusu, N. Blodow, and M. Beetz. Fast point feature histograms (fpfh) for 3d registration. In *2009 IEEE International Conference on Robotics and Automation*, pages 3212–3217, 2009. doi: 10.1109/ROBOT.2009.5152473. URL <https://ieeexplore.ieee.org/document/5152473>.
- D. Schmidt, T. Luksch, J. Wettach, and K. Berns. Autonomous behavior-based exploration of office environments. In *ICINCO-RA*, 2006. URL <https://www.scitepress.org/papers/2006/12040/12040.pdf>.
- J. L. Schönberger and J.-M. Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. URL [https://www.researchgate.net/publication/301197096\\_Structure-from-Motion\\_Revisited](https://www.researchgate.net/publication/301197096_Structure-from-Motion_Revisited).
- J. L. Schönberger, E. Zheng, M. Pollefeys, and J.-M. Frahm. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision (ECCV)*, 2016. URL [https://link.springer.com/chapter/10.1007/978-3-319-46487-9\\_31](https://link.springer.com/chapter/10.1007/978-3-319-46487-9_31).
- A. Segal, D. Hähnel, and S. Thrun. Generalized-icp. 06 2009. doi: 10.15607/RSS.2009.V.021. URL [https://www.researchgate.net/publication/221344436\\_Generalized-ICP](https://www.researchgate.net/publication/221344436_Generalized-ICP).
- G. Sithole and S. Zlatanova. Position, location, place and area: An indoor perspective. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, III-4:89–96, 2016. doi: 10.5194/isprs-annals-III-4-89-2016. URL <https://www.isprs-ann-photogramm-remote-sens-spatial-inf-sci.net/III-4/89/2016/>.

## Bibliography

- D. Sobel. A brief history of early navigation. *JOHNS HOPKINS APL TECHNICAL DIGEST*, 19, 1998. URL <https://www.jhuapl.edu/Content/techdigest/pdf/V19-N01/19-01-Sobel.pdf>.
- G. Spinoza Andreo, I. Dardavesis, M. de Jong, P. Kumar, M. Prihanggo, G. Triantafyllou, N. van der Vaart, and E. Verbree. Building rhythms: Reopening the workspace with indoor localisation. pages 106–116, 2021. URL <https://repository.tudelft.nl/islandora/object/uuid%3A060d104f-bce9-4608-9aa9-a73132317254?collection=education>.
- E. K. Stathopoulou and F. Remondino. Open-source image-based 3d reconstruction pipelines: Review, comparison and evaluation. In *6th international workshop lowcost 3d-sensors, algorithms, applications*, pages 331–338, 2019. URL [https://www.researchgate.net/publication/337644843\\_OPEN-SOURCE\\_IMAGE-BASED\\_3D\\_RECONSTRUCTION\\_PIPELINES\\_REVIEW\\_COMPARISON\\_AND\\_EVALUATION](https://www.researchgate.net/publication/337644843_OPEN-SOURCE_IMAGE-BASED_3D_RECONSTRUCTION_PIPELINES_REVIEW_COMPARISON_AND_EVALUATION).
- L. Sweeney. k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(05):557–570, 2002. URL <https://www.win.tue.nl/~jhartog/CourseVerif/Papers/10.1.1.90.4099.pdf>.
- L. Tong and Y. Xiang. 3d point cloud initial registration using surface curvature and surf matching. *3D Research*, 9, 09 2018. doi: 10.1007/s13319-018-0193-8. URL [https://www.researchgate.net/publication/327164739\\_3D\\_Point\\_Cloud\\_Initial\\_Registration\\_Using\\_Surface\\_Curvature\\_and\\_SURF\\_Matching](https://www.researchgate.net/publication/327164739_3D_Point_Cloud_Initial_Registration_Using_Surface_Curvature_and_SURF_Matching).
- R. A. Wadden and P. A. Scheff. Indoor air pollution. 1983. URL <https://www.bookdepository.com/Indoor-Air-Pollution-Peter-Scheff/9780471876731>.
- Z. Wang and M. Menenti. Challenges and opportunities in lidar remote sensing. *Frontiers in Remote Sensing*, 2, 2021. ISSN 2673-6187. doi: 10.3389/frsen.2021.641723. URL <https://www.frontiersin.org/article/10.3389/frsen.2021.641723>.
- O. Willems. Exploring a pure landmark-based approach for indoor localisation. Master’s thesis, 11 2017. URL <https://repository.tudelft.nl/islandora/object/uuid%3A6332755b-72ea-4a74-a781-5cb3f951de67?collection=education>.
- C. Wong, R. Klukas, and G. G. Messier. Using wlan infrastructure for angle-of-arrival indoor user location. In *2008 IEEE 68th Vehicular Technology Conference*, pages 1–5. IEEE, 2008. URL <https://ieeexplore.ieee.org/document/4656978>.
- L. Yang and M. Worboys. Similarities and differences between outdoor and indoor space from the perspective of navigation. 01 2011. URL [https://www.researchgate.net/publication/228843732\\_Similarities\\_and\\_differences\\_between\\_outdoor\\_and\\_indoor\\_space\\_from\\_the\\_perspective\\_of\\_navigation](https://www.researchgate.net/publication/228843732_Similarities_and_differences_between_outdoor_and_indoor_space_from_the_perspective_of_navigation).
- M. L. Yiu, C. S. Jensen, X. Huang, and H. Lu. Spacetwist: Managing the trade-offs among location privacy, query performance, and query accuracy in mobile services. In *2008 IEEE 24th International Conference on Data Engineering*, pages 366–375. IEEE, 2008. URL <https://ieeexplore.ieee.org/document/4497445>.
- M. L. Yiu, G. Ghinita, C. S. Jensen, and P. Kalnis. Outsourcing search services on private spatial data. In *2009 IEEE 25th International Conference on Data Engineering*, pages 1140–1143. IEEE, 2009. URL <https://ieeexplore.ieee.org/document/4812485>.

## Bibliography

- G. V. Zàruba, M. Huber, F. Kamangar, and I. Chlamtac. Indoor location tracking using rssi readings from a single wi-fi access point. *Wireless networks*, 13(2):221–235, 2007. URL [https://www.researchgate.net/publication/225608279\\_Indoor\\_location\\_tracking\\_using\\_RSSI\\_readings\\_from\\_a\\_single\\_Wi-Fi\\_access\\_point](https://www.researchgate.net/publication/225608279_Indoor_location_tracking_using_RSSI_readings_from_a_single_Wi-Fi_access_point).
- Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, 2000. doi: 10.1109/34.888718. URL <https://ieeexplore.ieee.org/document/888718>.
- Q.-Y. Zhou, J. Park, and V. Koltun. Fast global registration. volume 9906, 10 2016. ISBN 978-3-319-46474-9. doi: 10.1007/978-3-319-46475-6.47. URL [https://www.researchgate.net/publication/305983982\\_Fast\\_Global\\_Registration](https://www.researchgate.net/publication/305983982_Fast_Global_Registration).

## **Colophon**

This document was typeset using L<sup>A</sup>T<sub>E</sub>X, using the KOMA-Script class scrbook. The main font is Palatino.

