SLAM at the Katwijk beach

Marlon B. de Jong

Layout: typeset by the author using LaTeX.
Cover illustration: ESA's Automation and Robotics group

SLAM at the Katwijk beach

Marlon B. de Jong
11857323

Bachelor thesis
Credits: 18 EC

Bachelor *Kunstmatige Intelligentie*

University of Amsterdam
Faculty of Science
Science Park 904
1098 XH Amsterdam

Supervisor
Dr. A. Visser

Informatics Institute
Faculty of Science
University of Amsterdam
Science Park 904
1098 XH Amsterdam

July 28th, 2020

Acknowledgements

I would like to thank Robert A Hewitt for kindly answering my emails. Without his dataset this thesis would not have been possible. I'd also like to thank my supervisor Dr. Arnoud Visser for our weekly Skype sessions where he gave advice, suggestions, and feedback. Furthermore, I would like to extend my thanks to Mathworks for providing the Matlab licence which was essential for this thesis.

Additionally, I thank Demi Jansen van Wigmont for allowing me to send her my thesis and provide a much needed grammar check.

Abstract

On November 26th, 2015 a planetary rover drove across the Katwijk beach. Man-made rocks were arranged on the beach to imitate a Martian landscape. While driving the rover recorded several datasets. These can be used for the application of different SLAM techniques. In this thesis the focus is on a single sensor in the datasets: the stereo camera. Three SLAM techniques are applied to the stereo camera footage, to get a good estimate of the position of the rover and a reliable map. These techniques are: visual SLAM using points clouds, Structure from Motion, and a combination of both techniques.

Using the point clouds from visual SLAM and the location estimations from Structure from Motion, this thesis was able to create a visually more comprehensible map with a more accurate location estimation of the rover. The results indicate that by combining the techniques a better performance on both mapping and localisation can be achieved.

Contents

Abstract **1**

1 Introduction **4**
1.1 Research Background 5
1.2 Research Question 6

2 The Katwijk Beach Planetary Rover Dataset **7**
2.1 Introduction 7
2.2 Description of Recordings 8
2.2.1 Location of the Rocks 8
2.2.2 GPS-Latlong 9
2.2.3 GPS-UTM31 9
2.2.4 IMU 9
2.2.5 Odometry 9
2.2.6 LocCam 10
2.2.7 PanCam and Ptu 10
2.2.8 ToF 10
2.2.9 Velodyne 10

3 Simultaneous Localization and Mapping **12**

4 Ground Truth **13**

5 Visual SLAM Using Point Clouds **14**
5.1 Point Clouds 14
5.2 Merging Point Clouds 19

6 Structure from Motion **23**
6.1 Key Points 24
6.2 Essential Matrix 26
6.3 Bundle Adjustment 28

7 Method **30**

8 Related work **32**

9 Results **34**
9.1 Ground Truth 34
9.2 Visual SLAM Using Point Clouds 36
9.2.1 Part 1 36
9.2.2 Part 2 38
9.2.3 Part 3 39
9.2.4 Structure from Motion 41

9.2.5 Combined Method . . . 47
9.2.6 Part 1 . . . 47
9.2.7 Part 5 . . . 49
9.2.8 Part 6 . . . 51

10 Discussion **53**
10.1 Ground Truth . . . 53
10.2 Visual SLAM . . . 53
10.3 Structure from Motion . . . 53
10.4 Combined Method . . . 54

11 Conclusion **55**
11.1 Future Work . . . 55

References **57**

A Ground Truth **60**

B Visual SLAM Using Point Clouds **62**
B.0.1 Part 1 . . . 62
B.0.2 part 2 . . . 64
B.0.3 Part 3 . . . 66
B.0.4 part 4 . . . 68
B.0.5 part 5 . . . 70

C Structure from Motion **72**

D Combined method **73**
D.1 Part 2 . . . 75
D.2 Part 3 . . . 77
D.3 Part 4 . . . 79
D.4 Part 5 . . . 81
D.5 Part 6 . . . 83
D.6 Part 7 . . . 84
D.7 Part 8 . . . 86
D.8 Part 9 . . . 87
D.9 Part 10 . . . 89

1 Introduction

Where have you been, where are you, and where can you go? These are three important questions in robotics [10]. In order to answer these a map of the environment is needed. However, not every map is equally capable in answering these questions correctly for an autonomous vehicle. When using pre-existing maps to represent the environment there is a risk that the maps are no longer representative. Moreover, an autonomous vehicle must not only be able to avoid crashes with stationary objects, but also avoid coming into contact with dynamic objects. A pre-existing map is thus not sufficient for an autonomous vehicle to move without incident.

An autonomous vehicle bases its route on a map of its current environment. This map depicts objects and obstacles, such as rocks or buildings, as well as the autonomous vehicle's current position within the map. When the location of the autonomous vehicle itself is incorrect, the planned route will likely lead to a crash. The route assumes the location of the autonomous vehicle and plans from there, meaning that a wrong location will likely send the rover into the obstacles it was meant to avoid. An incorrect map would also lead to a crash, for similar reasons.

These factors illustrate why it is important that the autonomous vehicle is in possession of both an accurate environment representation and an accurate estimation of its own location within this environment.

Commonly GPS is used to provide this. However, GPS systems have a varying degree of accuracy, which can cause an inaccurate location estimate. Furthermore, there are locations where GPS is not available, such as underground mines, urban canyons, or, most notably, different planets. This is where SLAM comes into play. SLAM stands for Simultaneous Localization And Mapping and is the collective name for techniques which use information provided by sensors to map the environment of these sensors and locate them within the environment.

SLAM techniques do not require pre-existing knowledge about the environment, nor does it require a GPS system to determine the location of the vehicle. This allows the autonomous vehicle to explore unknown territories and tailor its reaction to dynamic changes. SLAM techniques have wide application potential and as of now it is already used in self-driving cars, soccer robots, and autonomous grass mowers. The planet Mars does not have the satellites needed to create an accurate location description using GPS, nor can one rely on a hypothetical pre-existing map of Mars because its landscape is dynamic, due to the notorious sandstorms [20]. SLAM techniques are therefore a suitable approach for Mars rovers.

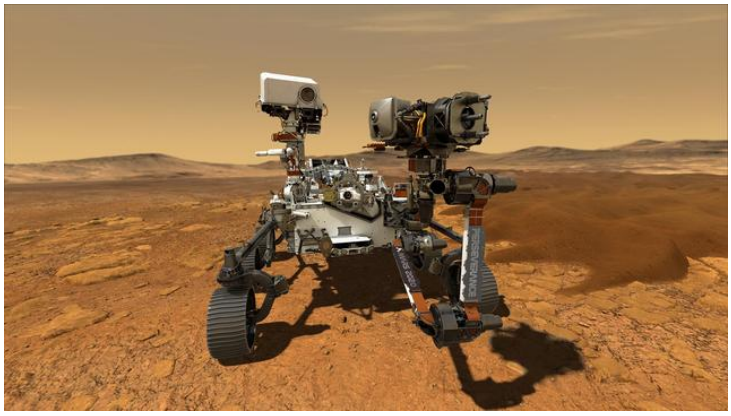


Figure 1: Mars rover Perseverance, developed for the 2020 NASA Mars mission[1].

1.1 Research Background

On the 26th of November 2015 man-made rocks were arranged on Katwijk beach to simulate a Martian landscape. A research group led by Robert A Hewitt then released a Heavy-Duty Planetary Rover (HDPR) equipped with nine sensors to gather a dataset that was exceptionally large [4]. The article that followed this experiment made this dataset public with the goal of enabling users to test and apply different SLAM techniques [11].

One of the SLAM techniques explored in this thesis is visual SLAM. This uses the stereo camera footage of the Katwijk beach dataset to create a 3D point cloud. Depth, the third dimension, can be determined by overlaying the left and right images from the stereo camera, this concept is similar to human eyesight. By overlaying consecutive point clouds, it is possible to determine the displacement between recordings. This gives an indication of how much the vehicle has moved since the previous recording. By continuously combining consecutive overlays a 3D RGB map of the environment is created.

Another SLAM technique is Structure from Motion. In contrast to visual SLAM this approach does not require stereo camera images, the footage of a single camera and its accompanying camera parameters suffice. Key points from consecutive images are extracted and matched. The camera parameters describe how a camera captures and distorts an image, this combined with the distance between each key point match makes it possible to calculate the relative position of the camera. The key points are stored as a 3D point cloud, which has no meaningful color. Over time the key point cloud extends and becomes a 3D representation of the environment.

Both SLAM approaches have aspects which could be improved. The camera position estimate of visual SLAM is a prediction which is not always correct. This can lead to an inaccurate map. The camera position estimate of the Structure from Motion technique generally performs relatively well, however the map representation, which consists of colourless points, is difficult to comprehend. A possible solution to this could be to

[1] *Perseverance on Mars*, NASA, March 2020, for more details see `https://mars.nasa.gov/mars2020/`

combine the strengths of each technique.

1.2 Research Question

A possible strong combination is to use the camera position estimations from Structure from Motion as a basis for the merging of point clouds generated by visual SLAM. *This thesis studies the potential outcomes of mapping the route travelled by the HDPR when calculated through Structure from Motion, visual SLAM, and a combination of both SLAM techniques.* The goal is to illustrate how these results differ, what their accuracy is compared to ground truth, and expound which factors could be of influence on their performance. The desired outcome is a completely accurate 3D map with an accurate location description. The assumption is that the combination of both techniques will lead to a result with a closer resemblance to the ground truth. The dataset, the techniques and their results are expounded in this thesis.

2 The Katwijk Beach Planetary Rover Dataset

2.1 Introduction

The Heavy-Duty Planetary Rover (HDPR) is equipped with sensors which in total collect nine sensor-datasets per route [11] [3]. The HDPR completed three routes with varying lengths and speed, their length and average speed can be found in Table 1. Route 1 is divided in eight parts and covers roughly the same terrain as route 2 which is divided in six parts. The difference is that route 1 travels northbound while route 2 travels southbound. This has an effect on the image quality, the southbound route suffered from sunlight in the camera lens causing blooming on most of the images. Blooming refers to white patches of pixels caused by overexposure. Route 3 is a relatively short route and is divided in five parts. The amount of time between data collection points, referred to as timestamps, is always equal, which is why there is a difference between route 1 and 2, and route 3 when it comes to data collection. The slower speed of route 3 means that the distance travelled between two data collection points is less, indicating that images are more similar which allows for better detail. Whereas route 1 and 2 have images that are further apart. Further differences stem from the direction of the routes, route 1 and 2 are both travelling in a continuous direction, whereas route 3 makes a loop. Route 3 has therefore more corners and images both with blooming and without. The timestamps of the nine sensors are not synchronised, meaning that each has a different interval for data collection. The routes can be seen in Figure 2.

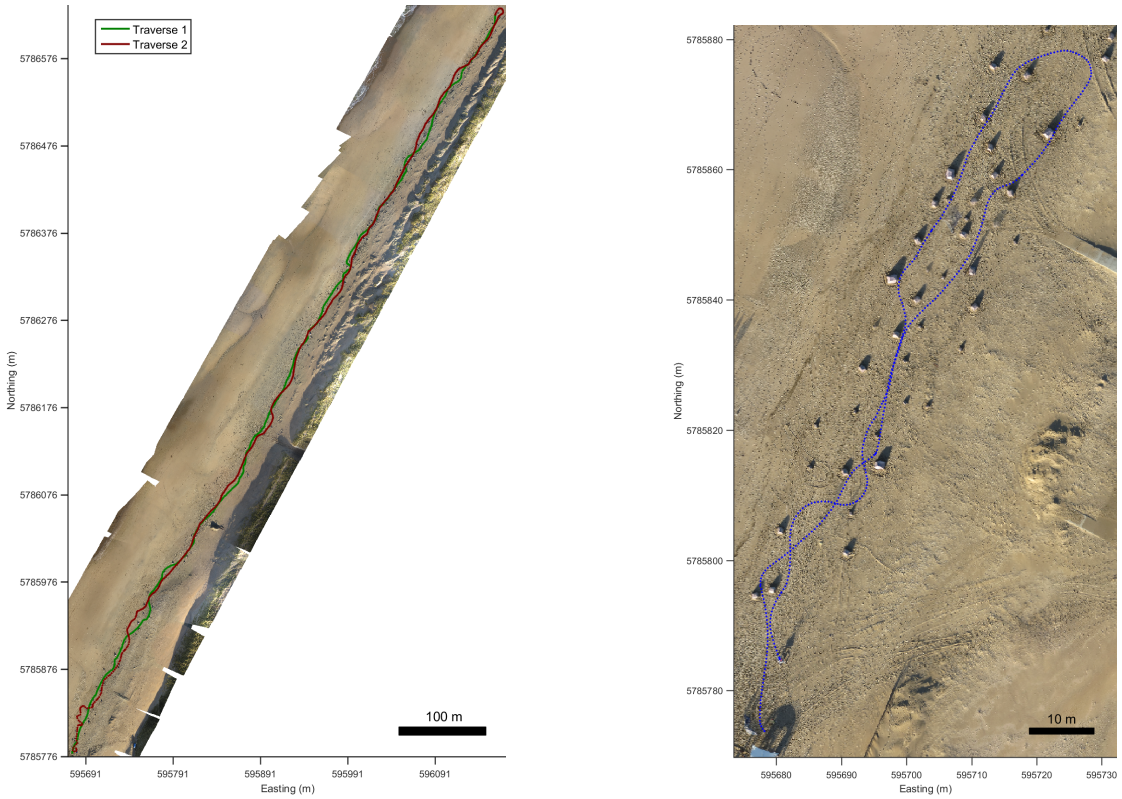
	Length in Kilometers	Average Speed in Meters per Second
Route 1	1.026	0.5077
Route 2	0.797	0.5057
Route 3	0.221	0.1813

Table 1: Table displays distance and average speed of the routes.

When applying SLAM, the route and the map can greatly improve when loop-closure occurs. This means that the sensors detect objects which are known from previous sections of the route and recognises them. The algorithm can use that recognition to get a higher degree of accuracy of the position estimate of the HDPR. Furthermore, these known objects can be used as landmarks to reduce the accumulation error. Route 3 has multiple occasions in which loop-closure occurs, while route 1 and 2 do not.

In this thesis the focus will be on route 3. This route has the most divers route, including both straight sections and sections with multiple curves, as can be seen in Figure 2 (b). Route 1 and 2 are between four and five times as large as route 3. Due to limited computing power is route 3 divided into five and at most ten parts. The size of route 1 and 2 implies that the amount of sub routes with this approach would be abundant. When more computing power is available the subsections can be merged, which would make the results of route 1 and 2 more comprehensible. For these reasons are route 1 and 2 not used in this thesis.

All data can be downloaded from Estec's website[2].



(a) Satellite image of route 1 and 2.

(b) Satellite image of route 3.

Figure 2: Satellite images of the terrain and all routes. Images can be found on the Estec website[3]

2.2 Description of Recordings

2.2.1 Location of the Rocks

The rocks are made of cardboard, all have the same shape and come in three different sizes: small, medium, and large. The location of the rocks is recorded in northing and easting in the UTM31N coordinate system. However, the altitude of the rocks is not recorded, which means that it is only possible to make a 2D map of the ground truth. The data points describing the location of the rocks can be found in the files:
`large-rocks-traverse12.txt`
`medium-rocks-traverse12.txt`
`small-rocks-traverse12.txt`

[2] `https://robotics.estec.esa.int/datasets/katwijk-beach-11-2015/`
[3] `https://robotics.estec.esa.int/datasets/katwijk-beach-11-2015/`

`large-rocks-traverse3.txt`
`medium-rocks-traverse3.txt`
and `small-rocks-traverse3.txt`

2.2.2 GPS-Latlong

The location of the HDPR per timestamp is recorded in RTK (Real Time Kinematics) GPS, which is an approach with a relatively high level of accuracy [3]. This dataset describes the location of the HDPR per timestamp. The timestamps have an interval of roughly three seconds. The measurements are expressed on the WGS84 ellipsoid, which is the standard coordinate system for GPS, and are recorded in latitude, longitude and altitude. The dataset also includes the standard deviation in meters per timestamp. The inaccuracy of the GPS, represented by the standard deviation, appears to be negligible, since it is mostly between 4 and 9 millimeters.
The data points describing the location of the HDPR can be found in the file:
`gps-latlong.txt`

2.2.3 GPS-UTM31

This file should contain the location of the HDPR per timestamp recorded in northing and easting in the UTM31N coordinate system. However this file seems to contain the same data as the GPS-latlong file. The authors of the article have been contacted and this should be rectified in the near future.

2.2.4 IMU

The information collected by an Internal Measurement Unit (IMU) includes a timestamp, acceleration in the x, y, and z direction recorded by an accelerometer. The angular velocity measured in the x, y, and z direction recorded by a gyroscope. It also includes another acceleration measurement in the x, y, and z direction recorded by an inclinometer.
The data collected through the IMU can be found in the file:
`imu.txt`

2.2.5 Odometry

Odometry refers to the information which keeps track of the displacement of each wheel between timestamps. This file contains both the angular displacement and the steering angular displacement of each wheel per timestamp. Also included is the orientation of the rocker, and the left and right bogie. However this file contains more columns than was specified in the article [11]. This because each wheel and steering joint has two extra columns that follow after the displacement value, these correspond to angular velocity and the analogue value reported by the encoder. According to the author where these accidentally left in the dataset.
The odometry data can be found in the file:
`odometry.txt`

2.2.6 LocCam

The images taken by the stereo cameras, a PointGrey Bumblebee2, are stored in the zip file `LocCam`. Stereo means that two cameras take the same image at the same timestamp. The images differ in the fact that the cameras are situated 12 centimeters apart. Using two cameras with a known distance between them enables the user to reconstruct a 3D image of the scene. By combining and overlaying images depth can be extracted.

2.2.7 PanCam and Ptu

During the route panorama images were taken by a PointGrey GrassHopper2. This is a stereo camera of which the cameras are placed fifty centimeters apart and rotate while the HDPR is traversing. The information about the orientation of the pan-tilt frame on which the sensor is mounted is stored in the file `ptu.txt`. It contains a timestamp, and the pan and tilt displacement between timestamps. The images taken by a panorama camera are stored in the zip file `PanCam`.

2.2.8 ToF

Depth and intensity images were recorded with a SwissRanger. This is a sensor based on the Time of Flight (ToF) principle. The sensor sends out a light pulse and an object reflects this light back into the sensor. The time it took to travel from the sensors to the object and back is the time of flight. The speed of light is known and via a simple calculation the distance between the sensor and the object can be determined. Through this the sensor is able record both the intensity of the image as well as depth. The zip file named `ToF` contains the data recorded by the SwissRanger.

2.2.9 Velodyne

The zip file named Velodyne contains data which is collected through a 3D LiDAR sensor. LiDAR stands for Light Detection And Ranging, this is a technique which sends out laser pulses and measures the time between sending out the pulse and the pulse returning. This sensor provides dense point-clouds. The recorded data can be found in the zip file `Velodyne`.

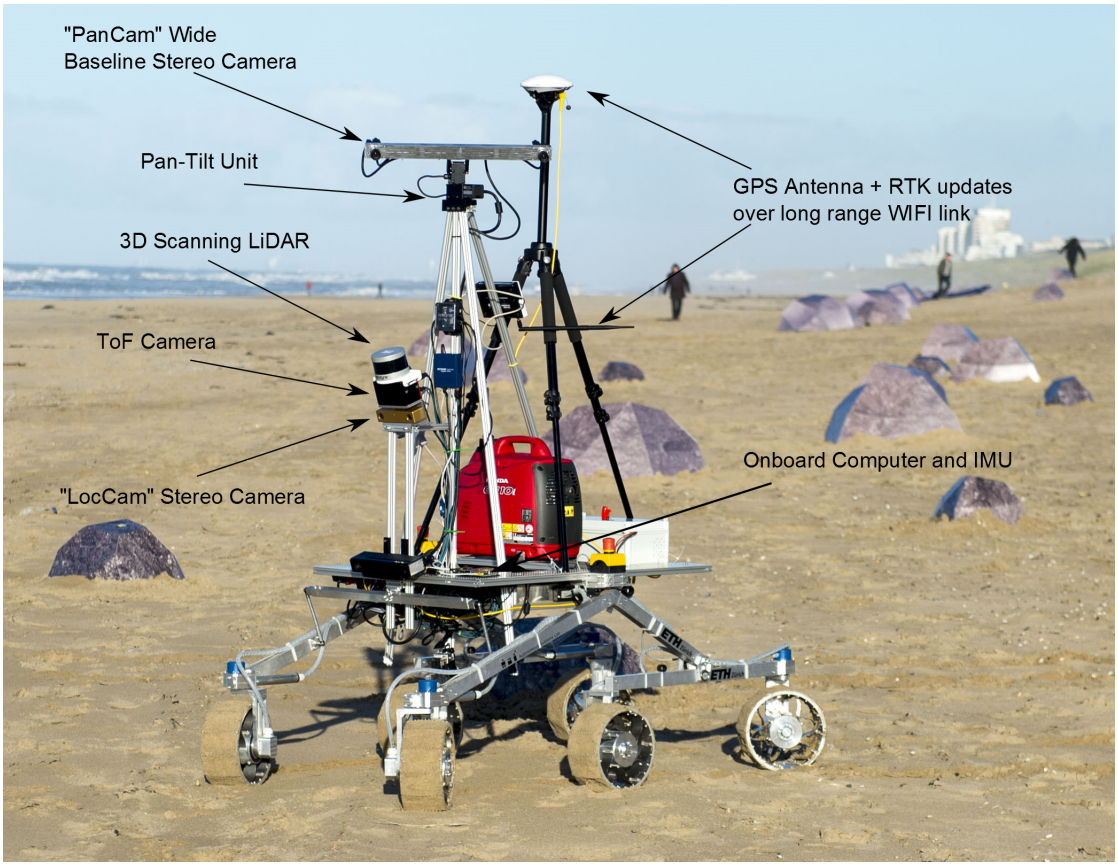


Figure 3: The HDPR with indicator of where the sensors are. In the background the artificial rocks of the parkour can be seen. Image can be found on the Estec website[4].

[4] https://robotics.estec.esa.int/datasets/katwijk-beach-11-2015/

3 Simultaneous Localization and Mapping

In the following chapters 4, 5 and 6 the theory which is the foundation of this thesis will be discussed. There are many different SLAM approaches, each with its own benefits and disadvantages [27]. The approaches tested in this thesis are: visual SLAM via point clouds, Structure from Motion, and a combination of both. In order to test the accuracy of these methods a comparison needs to be made against an independent medium which has a relatively high accuracy. This medium would establish a ground truth that describes the route and environment that SLAM is compared against. The amount in which the route and map calculated through the SLAM approaches differs from the ground truth represent their performance.

For this thesis `MATLABR2020a` was used with the Computer Vision toolbox. Note that this toolbox is not standard and does require an additional licence provided by Mathworks as main sponsor of the RoboCup.

4 Ground Truth

The ground truth describes the location of the rocks and the movements that the Heavy-Duty Planetary Rover (HDPR) has made. How much the calculated route and map differs from the ground truth, determines the quality of the produced route. The accuracy of the SLAM approach is relatively high when there is little to no difference between its map and route and the ground truth.

The location of the HDPR is based on the data from the GPS-latlong dataset. The latitude, longitude, and altitude describe the position of the HDPR per timestamp. However, GPS is known for the tendency to be inaccurate. The standard deviation of the GPS is included in the dataset and shows the deviation to be between 4 and 9 millimeters, which is negligible.

The exact locations of the rocks is ambiguous. The files do not include where on the rocks the measurements were taken, nor is the standard deviation of these measurements given. The assumption that the measurements are based on the center of the rocks does not hold because the rocks do not have a clear center point, as illustrated in Figure 4. The altitude of the rocks is not recorded, therefore it is not possible to produce a 3D map of the environment.

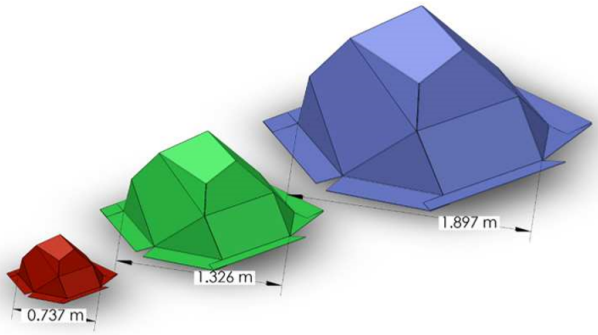


Figure 4: 3D models of the different rock sizes, illustrates that there is not a clear center point on the rocks[5]

[5]This image is Figure 3 from Hewitt *et al* [11]

5 Visual SLAM Using Point Clouds

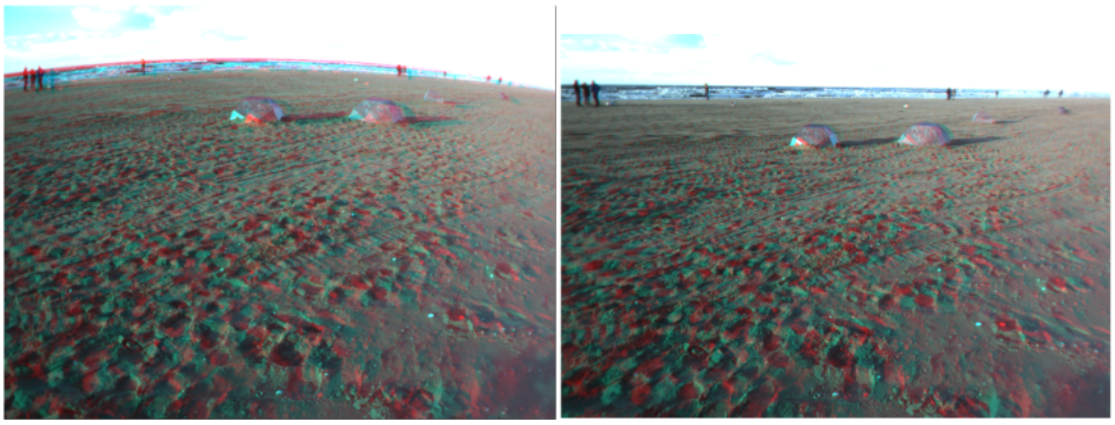
5.1 Point Clouds

SLAM based on imagery is called visual SLAM. The imagery used in this thesis is captured by a stereo camera, a PointGrey Bumblebee2. Stereo cameras are two cameras that capture the same scene, but which are located slightly apart. The distance between the cameras is known, which makes it possible to overlay the images and create depth. An example of the stereo images can be seen in Figure 5.

(a) Left image. (b) Right image.

Figure 5: Left and right stereo camera image from route 3.

In order to overlay the 2D images and create a 3D scene, it would be convenient if the corresponding point between the two images have the same row coordinates. The images captured by the stereo camera do not possess this quality. There is distortion caused by the camera lens and other factors inherent to the cameras. The influence of the camera on the image needs to be removed in order to properly overlay the images. In this dataset the camera parameters, which include the data needed to correct the images, were provided by the makers of the dataset [11]. The process in which the original images are corrected for distortion caused by the camera is called rectification [9]. This is illustrated by Figure 6 which displays the influence of distortion before and after rectification.



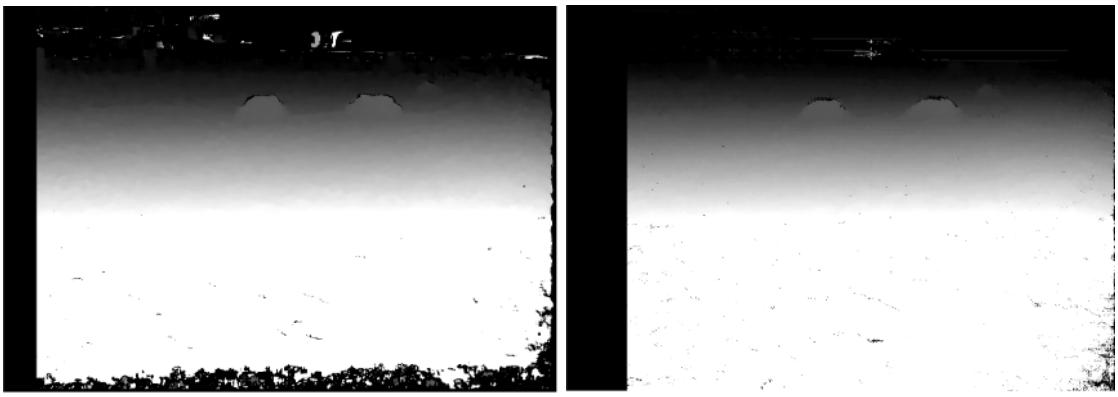
(a) Overlay before rectification.

(b) Overlay after rectification.

Figure 6: (a) is the overlay of the stereo images before rectification. (b) is the overlay of the stereo images after rectification

Image (b) of Figure 6 illustrates the degree of shift between the scene caught by the left and the right stereo camera. Shift is referred to as disparity. Objects nearby display a relatively large amount of disparity, while objects far away have little to none. The severity of disparity is therefore a representation of depth. When comparing the amount of disparity it is possible to translate the overlaying 2D images to a 3D representation. This is the disparity map. This map displays the difference in pixel positioning for every feature in the images and this can be used to extract depth from the rectified images.

There are multiple approaches for calculating the disparity map, the approaches explored in this thesis are block matching and semi-global matching. Block matching is a local method which looks for matching pixels within a region, which is a small number of pixels, surrounding the pixel of interest [15]. This is in contrast with the semi-global matching where the program looks for the matching pixel in directions instead of in regions [12].



(a) Block matching. (b) Semi-global matching.

Figure 7: Disparity map results.

The differences between the result of semi-global matching versus block matching can be seen in Figure 7. The techniques do not have a vast difference, but the semi-global matching appears to be more complete, this is why this approach has the preference in this application.

By combining the rectified images, the camera parameters, and the depth information from the disparity map, it is possible to create a 3D point cloud of the scene captured by the stereo cameras. This point cloud has an additional axis which represents the depth of the scene. This additional axis allows the user to optimise the point cloud and reduce the level of unnecessary objects in the background. Figure 8 exemplifies what adjusting this axis does. Reducing objects in the background could be convenient, since the sea is visible in most images and there are occasions where there are people walking in the far distance. As of now there are no people on Mars, nor is there an ocean. Therefore they are considered a disturbance which could be removed by applying a depth limit.

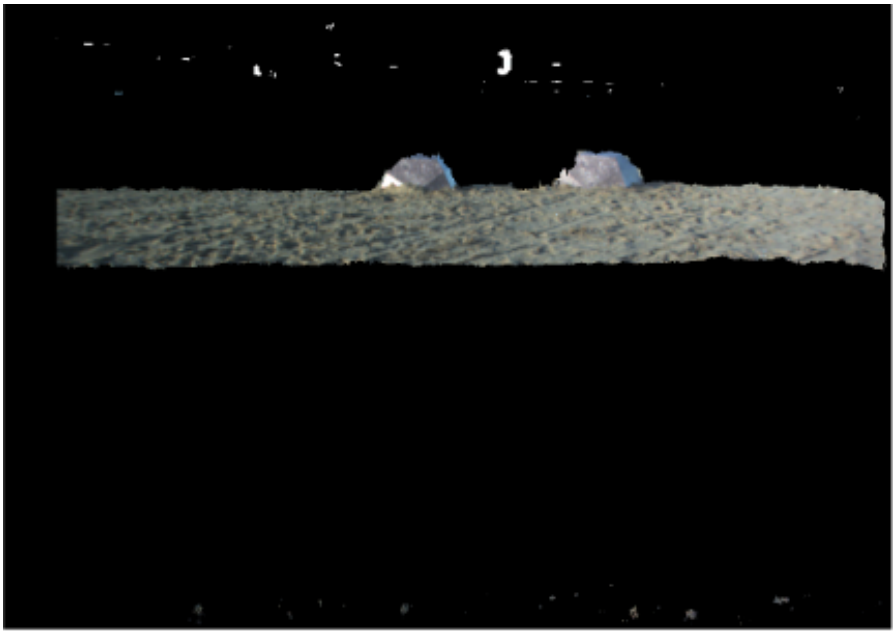
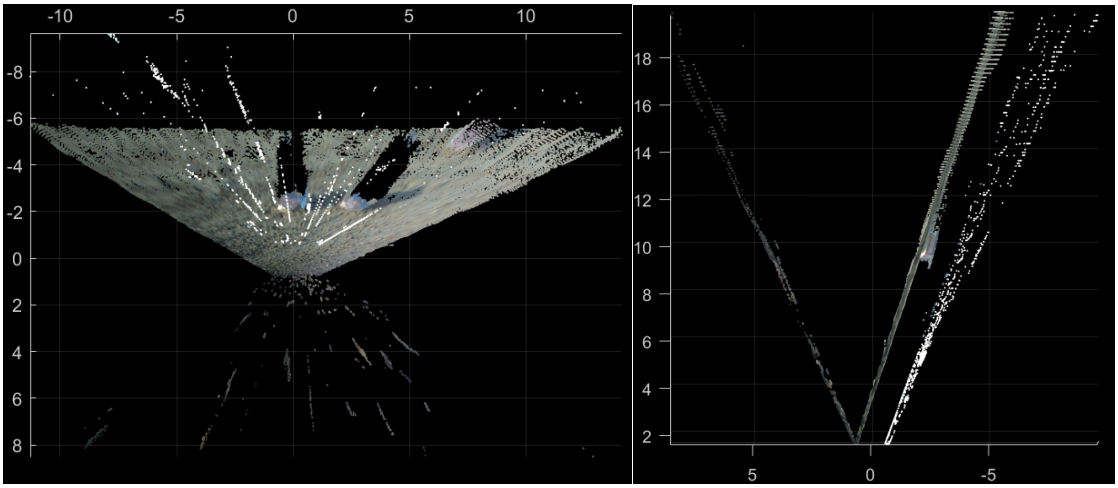


Figure 8: Image shows what is between 5 and 10 meter distance of the camera.

Other noise is visible in Figure 9. Some of this visual noise was already removed by a Matlab denoise function, however, as the image shows it was not capable of removing all noise. The leftover noise is underneath the ground level and in a white cloud above the scene. This noise can be removed by putting a mask on the point clouds which excludes all data which is outside a certain height.



(a) Frontal view of point cloud. (b) Side view of point cloud.

Figure 9: Frontal and side view of the first point cloud after application of Matlab denoise function and before additional noise removal.

The stereo camera is positioned with a small downward angle on the HDPR. This causes the point clouds to be angled as well, as can be seen in Figure 9 (b). This can be altered by rotating the scene with $\frac{1}{10}\pi$. The final point cloud can be seen in Figure 10, 11, and 12.

Figure 10: Side view of de-noised point cloud.

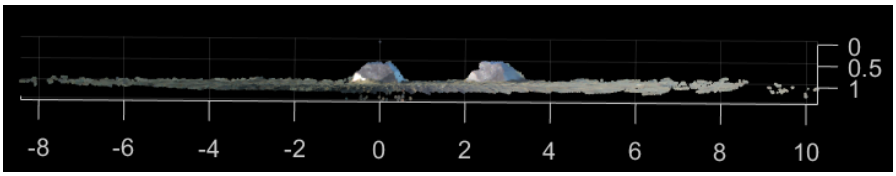


Figure 11: Front view of de-noised point cloud.

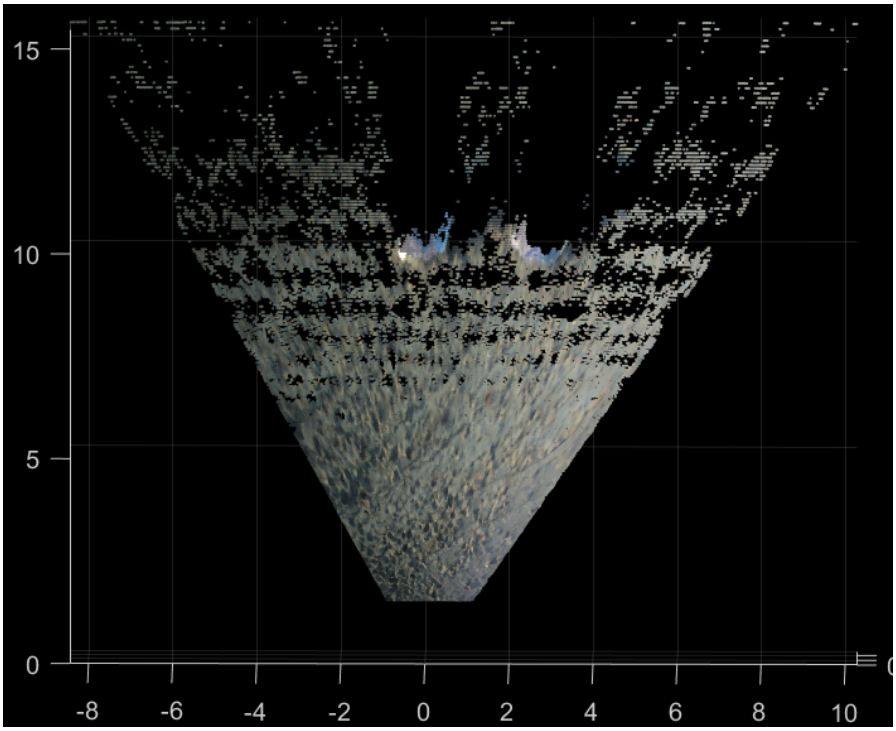
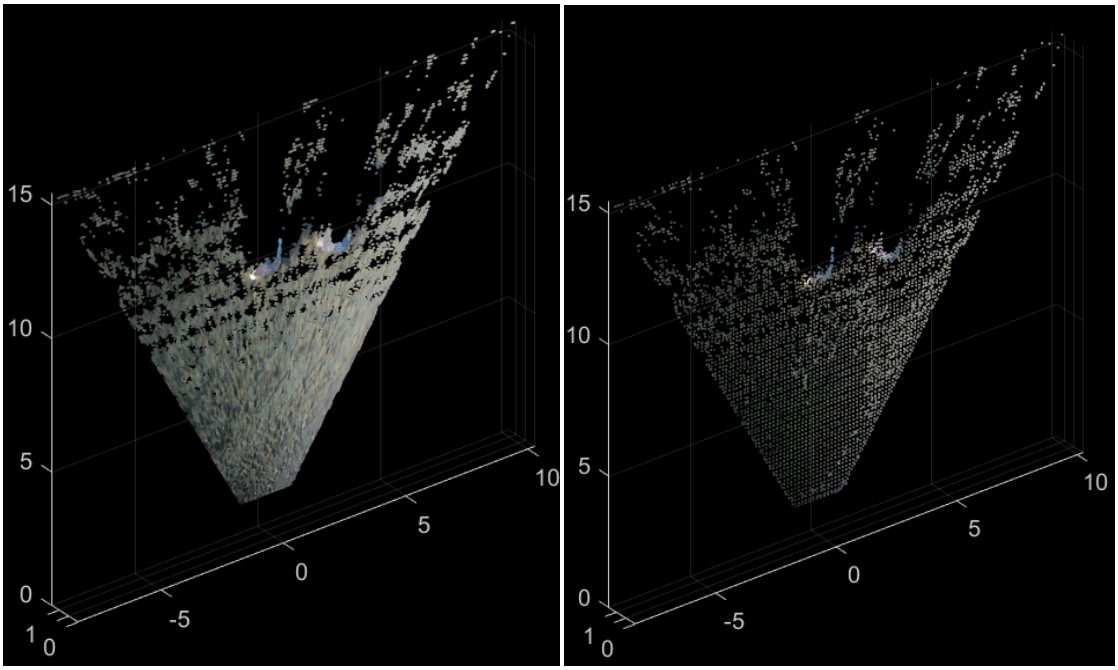


Figure 12: Top view of de-noised point cloud.

5.2 Merging Point Clouds

The point clouds of each timestamp visualise what the stereo cameras observe per timestamp. However, this observation is quite limited. The data captured by the stereo cameras can be optimised by merging the point clouds, hereby creating a 3D representation of the surrounding area. In order to merge two point clouds, they must be pre-processed. The first step is to downsample them. The downsampling is done to limit the influence of noise on the final output. The downsampled point clouds are used to determine the displacement between them. A box grid is applied to the point clouds, which divides them into cubes. The points which are located within this cube are combined to produce the average colour value of all points within this cube. The size of the cubes is determined by the grid size. If the grid size is relatively small, the grid will produce a precise representation of the original point cloud. This precise representation could however be prone to noise, hereby overruling the function of downsampling. Additionally, a small grid size uses a lot of memory and needs a lot of processing time. If the grid size is relatively large then processing will be quick, but imprecise. An example of downsampling can be seen in Figure 13.



(a) The original point cloud. (b) Downsized point cloud.

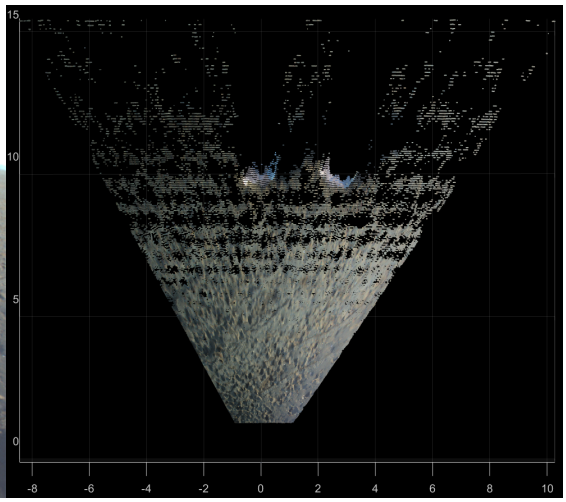
Figure 13: Illustration of effect of downsampling with grid size of 10 centimeter

The two point clouds of different timestamps are given the names 'fixed' and 'moving'. The fixed point cloud is the first point cloud produced by the stereo cameras. All following point clouds are transformed relative to the original first fixed point cloud. Between the two images the HDPR has moved and changed position. Both point clouds are self centred, meaning that each new image puts the stereo camera at the origin. The position of the moving point cloud must be altered in order to merge with the fixed point cloud. The transformation between the point clouds is a rigid one and can be calculated through the Iterative Closest Point (ICP) algorithm. ICP is an approach to estimate which transformation must be applied to align the moving point cloud with the fixed point cloud [5].

The two down sampled point clouds are overlain, and the closest point within the fixed point cloud is calculated for each individual point in the moving point cloud. This is the data association step. Subsequently, the moving point cloud is repeatedly transformed and aligned with the fixed point cloud to lessen the error. The error is defined as the least square error, this is the squared sum of the distance between the points of the moving point cloud and their closest point in the fixed point cloud. The data association and alignment steps are then repeated until the two point clouds converge. After applying the rigid transformation calculated via ICP, the two point clouds largely cover the same data, meaning that some data points are displayed twice. In order to discard the double data, the separate point clouds can be merged to form a new point cloud. The overlapping

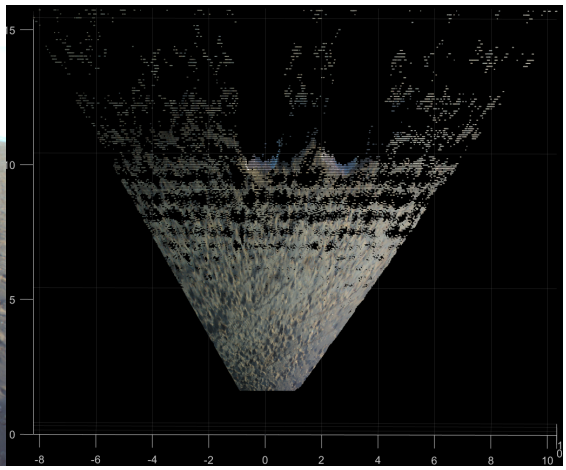
point clouds are merged using a box filter. The size of the box filter determines the memory which is needed and the resolution of the scene. One can increase the merge size to reduce the storage requirement of the resulting merged point cloud, and decrease the merge size to increase the scene resolution. An example of merging point clouds can be seen in Figure 14.

(a) First image from left camera.



(b) Accompanying point cloud.

(c) Second image from left camera.



(d) Accompanying point cloud.

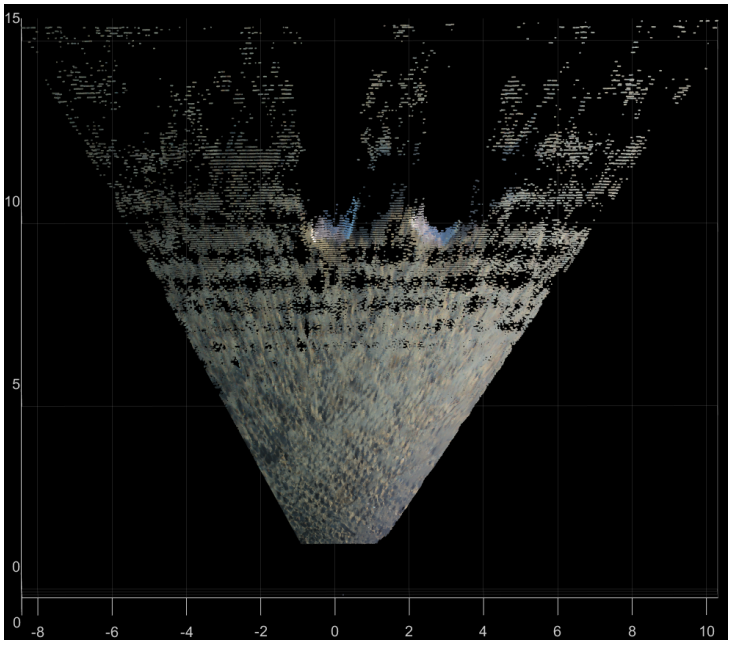


Figure 14: The two point clouds are merged to form a new larger point cloud.

This process must be slightly altered in order to align and merge all the following timestamp point clouds accordingly. All moving point clouds must not only be transformed and aligned in accordance to the coordinate system of their direct predecessor. They must also be transformed in accordance to the fixed point cloud. The aligned point cloud will merge with all preceding point clouds. How much the point cloud has moved with respect to the previous timestamp is the displacement of the HDPR and is recorded in the rigid transformation produced by the ICP algorithm. By storing all intermediate rigid transformations it is possible to reconstruct the route of the HDPR. This technique thus enables the HDPR to map the environment and locate itself within the map hereby making it simultaneously locate itself and map the environment.

6 Structure from Motion

Structure from Motion is a technique that uses the most unique features in a series of images to produce a 3D point cloud. These features are key points and are matched between multiple images to decipher the distance between them. This distance allows for the making of a 3D space where the key points and camera locations are displayed. Before the key points can be extracted the images must first undergo two pre-processing steps. The first step of this approach is to turn the original images into grayscale images, seen in Figure 15.

Figure 15: The original six images are turned to grayscale.

The second step is similar to the visual SLAM approach, namely removing the camera distortion from the image. In this dataset the camera parameters, which include the data needed to correct the images, was provided [13]. The original image and its undistorted version can be seen in Figure 16.

Figure 16: Example of original grayscale image and the undistorted version.

6.1 Key Points

Turning the images to grayscale and rectifying them is preparation for the key point extraction. Key point extraction can be done in a variety of ways, but the Speeded-Up Robust Features (SURF) algorithm was chosen in this thesis because it is scale invariant as well as invariant to geometric and photometric variations [1]. This is very fortunate when applying SLAM, because its very use implies the occurrence of these variations. The groundwork of SURF is the Harris–Stephens algorithm [8], also referred to as Harris Corner Detector. The Harris–Stephens algorithm detects corners in an image, which are in essence key points. The drawback is that this algorithm is not scale-invariant [2]. SURF makes it scale invariant by incorporating techniques such as Laplacian of Gaussian [22] and Scale Invariant Feature Transform (SIFT). SIFT makes use of the difference of Gaussian function [18]. An example of the key points from an image can be seen in Figure 17.

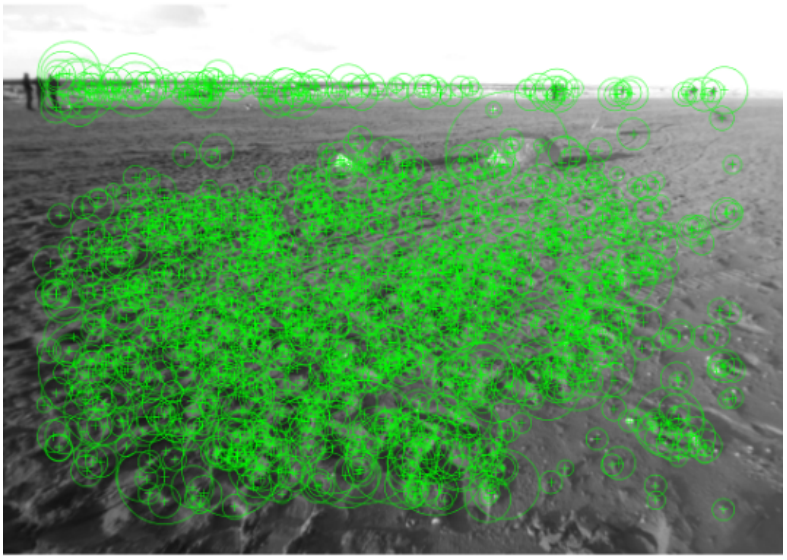


Figure 17: The key points are calculated by the SURF algorithm and are circled in green.

To prevent the registering of unnecessary key points SURF can be set with a region of interest. A border is put on an image which is then excluded when the key points are determined. Key points are given an identity in order to match them across frames. This identity must be as unique as possible in order to prevent false matches. A possible approach for making it more specific per key point is to attach features which are directly above the key point to the identity of the key point. However, this is only possible if there is little to no in-plane rotation. The Katwijk beach dataset is suited for this because all images are taken with approximately the same image orientation, which is only changed slightly by the terrain across which the HDPR travels. Since the terrain is fairly level, this should not pose a problem.

The distance between matched key points indicates what the distance is between the recording locations. It is imperative that key points are matched with just their unique counterparts and not with another that might be close. Meaning that a key point can not be matched with multiple other key points in a different image. An example of what key point matches look like can be seen in Figure 18.

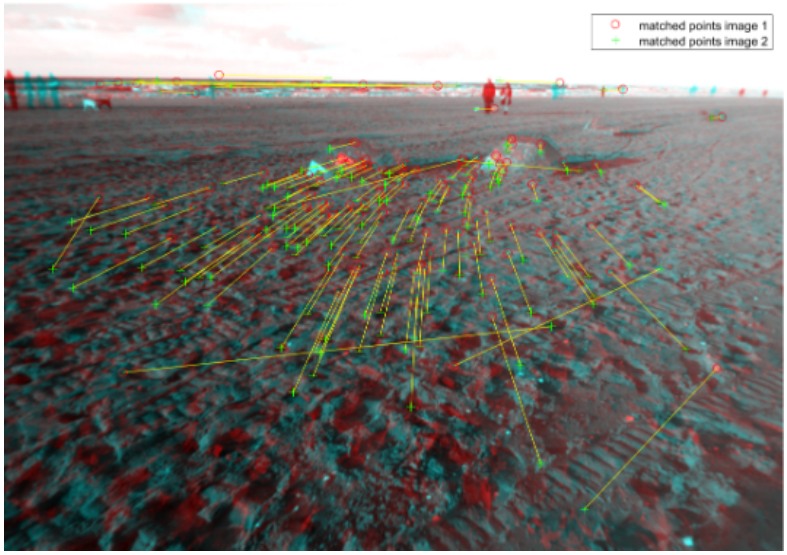


Figure 18: Visualisation of the matching key points across two images. Note that there are some outliers.

6.2 Essential Matrix

The position of the camera can be estimated by using the essential matrix E. The distance between key point matches and the camera parameters are used to describe the geometric relation between images. This matrix contains information which describes the relative orientation from corresponding points. E is determined by the coplanarity constraint of the matching key points. The constraint implies that there is a 3D plane which contains the locations x' and x'', which are the same key point across frames, and the 3D representation of this key point. [17].

Mathematically this looks like:

$$\begin{aligned} x' &= \text{key point in image 1} \\ x'' &= \text{same key point in image 2} \\ x'^T E\ x'' &= 0 \end{aligned}$$

The HDPR travels an inconsistent distance between timestamps. This means that in some instances there is too little difference between the images from consecutive timestamps to make an accurate essential matrix E. This is because the influence of noise is large when there is little difference between two images. An example of a case where there is too little difference between images can be seen in Figure 19.

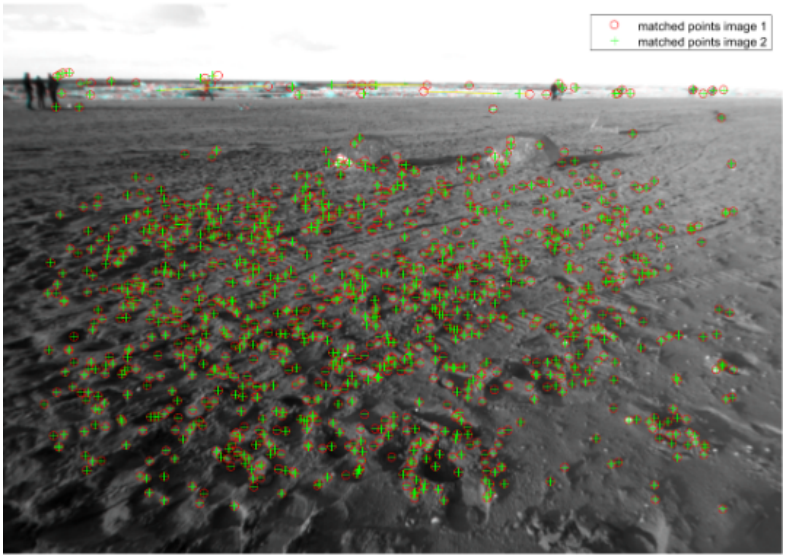


Figure 19: There is too little difference between image 1 and 2 to calculate E.

The accuracy of E can be determined by the fraction of key points for which the equation $x'^T E\ x'' = 0$ holds. When this fraction is too low E must be recalculated. If, after recalculation, the fraction is still too low then the accompanying image is discarded. When there is a satisfactory amount of inliers, the position of the cameras, their rotation, and the location of all inlier key points can be computed.

E enables the user to find more matching points between images by using epipolar lines. Given a point in the previous image, E can be used to span a plane which cuts a line, the epipolar line, through the next image. Somewhere along this line the image point is located. Using epipolar lines drastically limits the search space of an image for locating matching key points. By using these lines the locations of key points in 3D space are determined and a point cloud containing all key points is created.

In the approach used in this thesis the essential matrix E was determined by the M-estimator Sample And Consensus algorithm (MSAC). MSAC is an extension to RANSAC in the fact that it also takes the likelihood of matching point pairs into account [24]. This is important because, even though the key points should have a unique identity, it is still possible that key points are wrongfully matched. The key point matches which are deemed likely to be true are used to determine E.

By using the essential matrix, the camera parameters, and the inlier key points, the orientation and the baseline vector of the camera can be produced. The orientation describes how the next image has rotated with respect to the previous image. The baseline vector describes the location of the next camera with respect to the previous camera, but not what the actual distance is between the cameras. Meaning that the direction of the camera is known but not the distance between cameras. Due to this, the coordinate system of Structure from Motion is not in meters, but uses pixels of the camera coordinate system as the distance measure.

6.3 Bundle Adjustment

The estimate of the camera location and the locations of the key points in 3D space, can be further optimised by applying bundle adjustment. This works with the notion that there is a set of 3D key points: X_j. These key points are observed by cameras which each have their own projection matrix: P_i. This matrix describes where in the image a key point would project. Thus, the location of a specific key point is described by $x_{ij} = P_i X_j$, where x_{ij} are the image coordinates of the j-th key point in the i-th image [25].

Bundle adjustment is an approach which limits the summed square reprojection error by adjusting the projection matrix(P_i) and the world coordinates(X_j). The reprojection error can be seen as a quality marker which calculates the 3D coordinates of a key point based on two or more images. These coordinates are based on the internal and external parameters of the camera, as well as the position of the point in the images. Once the 3D coordinates of the point are computed, the 3D point is re-projected on all images where the point is present. The distance between the actual location of the point and the re-projected location is the reprojection error. The smaller the error the better the location of the camera is recorded.

Mathematically this looks like:

$$\min_{P_i, X_j} \sum_{ij} d(P_i X_j, x_{ij})^2$$

Where $d(x, y)$ is the Euclidean distance between image points x and y [25].

Bundle adjustment is a non-linear minimization problem which can be solved using iterative non-linear least squares methods such as the Levenberg-Marquardt algorithm [21][16].

The bundle adjustment re-determines the locations of all previous cameras, and all 3D key points, based on the information which is provided by a new image [25]. Hereby lessening the influence of an accumulating error. By repeating these steps for all consecutive images it is possible to create a 3D world scene consisting of the key points, which also displays the position of the cameras. An example of this can be seen in Figure 20.

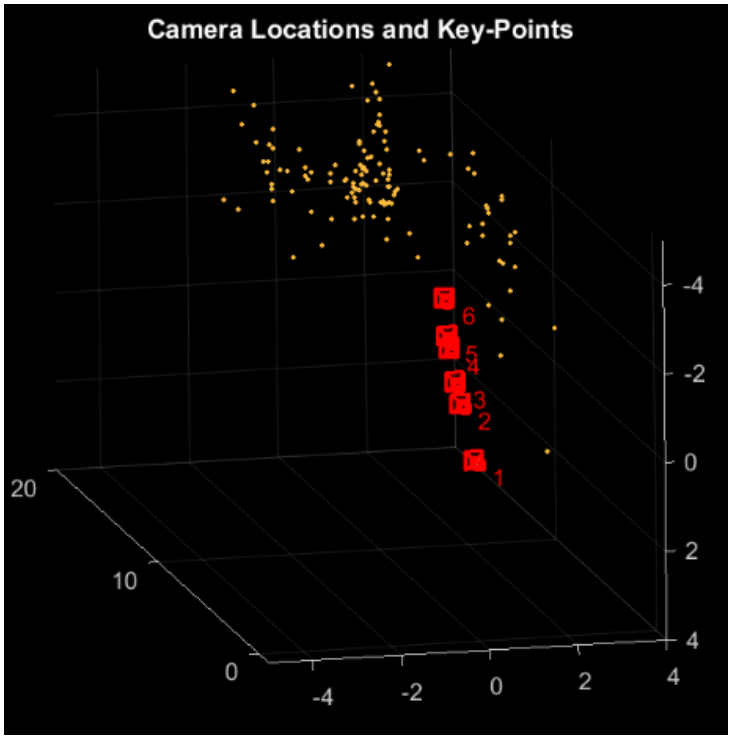


Figure 20: 3D representation of the scene, based on 6 images.

7 Method

In the approach of this thesis the map calculated through Structure from Motion has little texture and is relatively difficult to comprehend, compared to a map produced via point clouds. The latter one produces a more visually pleasing map representation, however, Structure from Motion delivers a more accurate location prediction. For these reasons the proposed SLAM technique of this thesis is a combination of both techniques. The goal is to apply the rigid transformation calculated through Structure from Motion to the point cloud, which is called 'moving'. After the rigid transformation, the 'moving' point cloud should be located in the correct position with respect to the 'fixed' point cloud. Subsequently, the point clouds can be merged in the same manner as visual SLAM.

The point clouds from visual SLAM are in a meter coordinate system, whereas Structure from Motion uses the camera coordinate system. This illustrates the necessity to transform the location descriptions from Structure from Motion to a system that also uses meters. A possible approach to transform the camera coordinate system into a meter coordinate system is to use the known distance between the left and the right camera to extrapolate the distance in meters between the coordinates. The distance between the left and right stereo camera is known to be 12 centimeters. This distance can be seen as a vector with the same length. However, in Structure from Motion the distance between the two cameras is constantly adjusted by the bundle adjustment. Therefore an average of this vector length has to be taken between multiple image pairs in order to approximate the right value. Following, there are occasions where the distance between individual camera pairs is far greater or smaller than the average distance. These image pairs are outliers and therefore must be removed. The mean distance between the left and right camera is recalculated. While it is known that the distance between cameras is 12 centimeters, the current system uses an arbitrary value to represent this distance. The following step is to find a factor by which to multiply this arbitrary value so it will equal 12 centimeters, thus transforming the system in a meter coordinate system. This factor is produced in the following manner:

$$
\begin{aligned}
baseline &= \text{mean(distances between left and right camera)} \\
scale &= baseline/0.12m \\
factor &= 1/scale
\end{aligned}
$$

If the computation is done right then the first camera position, the left camera of the first timestamps, is located at [0 0 0]. This means that the coordinates of the cameras are left camera centered. However, the point clouds base the location on the stereo camera, not just the left camera. In order to simulate this all camera locations must be translated 6 centimeters to the left.

The translation and rotations which describe the location of the stereo camera is an average of the left and right camera rotation and translation. The rotation of the left

and the right camera should be very similar because the cameras are mounted in the same frame and therefore have the same orientation.

Indexes are created with the images that comply with the accuracy demands, outliers are excluded. Visual SLAM produces point clouds of the images and merges them according to the camera locations from Structure from Motion. By applying the rigid transformation calculated through Structure from Motion to the point clouds used in visual SLAM it is possible to create a 3D map of the environment and locate the HDPR within this environment.

8 Related work

The work presented in this thesis builds on multiple previous studies. A study from 2014 modeled the terrain of a river by using aerial images to generate 3D point clouds and stitch these together using Structure from Motion [13]. Another research paper with a similar approach also used aerial images to map both the structure of a vineyard and where its vegetation was located [23]. Both of these studies used aerial images, whereas the Katwijk beach dataset consists of ground level imagery. Images taken at ground level have the added complexity of multiple perspectives. This is more similar to another study from 2014, where the same method was applied to ground level images. Specifically, this study was interested in making 3D RGB models of archaeological sites [7].

These three studies do not show an interest in location data, which is an essential element of SLAM. Nor do they show an interest in the production of a comprehensible RGB map of the terrain. A study which does have interest in location data and which subject is similar to this thesis, was performed by Paul Timothy Furgale [6]. In his thesis Furgale researched how a stereo camera and LiDAR can be used for SLAM purposes on a Martian dataset. Structure from Motion was used to determine the location of the rover, but no 3D representation of the terrain using point clouds was made. The accuracy of the route calculated by Furgale can be seen in Figure 21.

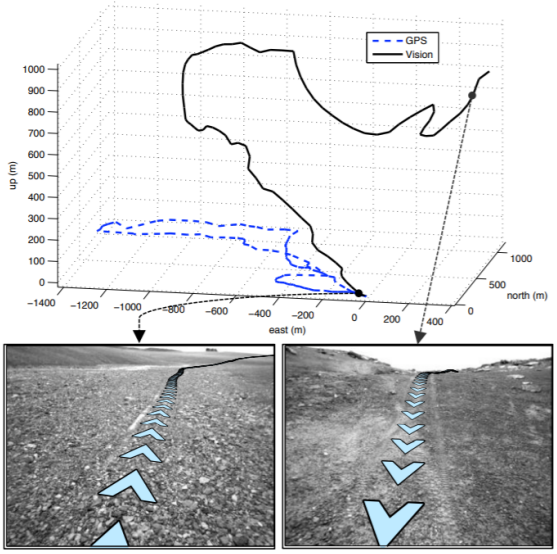


Figure 21: The route according to vision deviates greatly from the GPS route. However, locally the route is correct as can be seen in the two images taken at the same location.[6]

[6]This image is Figure 5.6 in Furgale's thesis [6].

The dataset used by Furgale was supplied by the University of Toronto Institute for Aerospace Studies. No previous research was found that applies point cloud and Structure from Motion techniques to the Katwijk beach dataset [11].

Where this thesis differs from the related work is in its use of visual SLAM to create point clouds. Visual SLAM also allows for the creation of a more visually comprehensive map, with more detail and colour than previous studies were interested in creating.

9 Results

The data of the route was provided in five parts, where possible this distribution was used. In this chapter only the most interesting results are displayed. The results of all parts can be found in the appendix. The results are displayed per section of the route, per technique.

9.1 Ground Truth

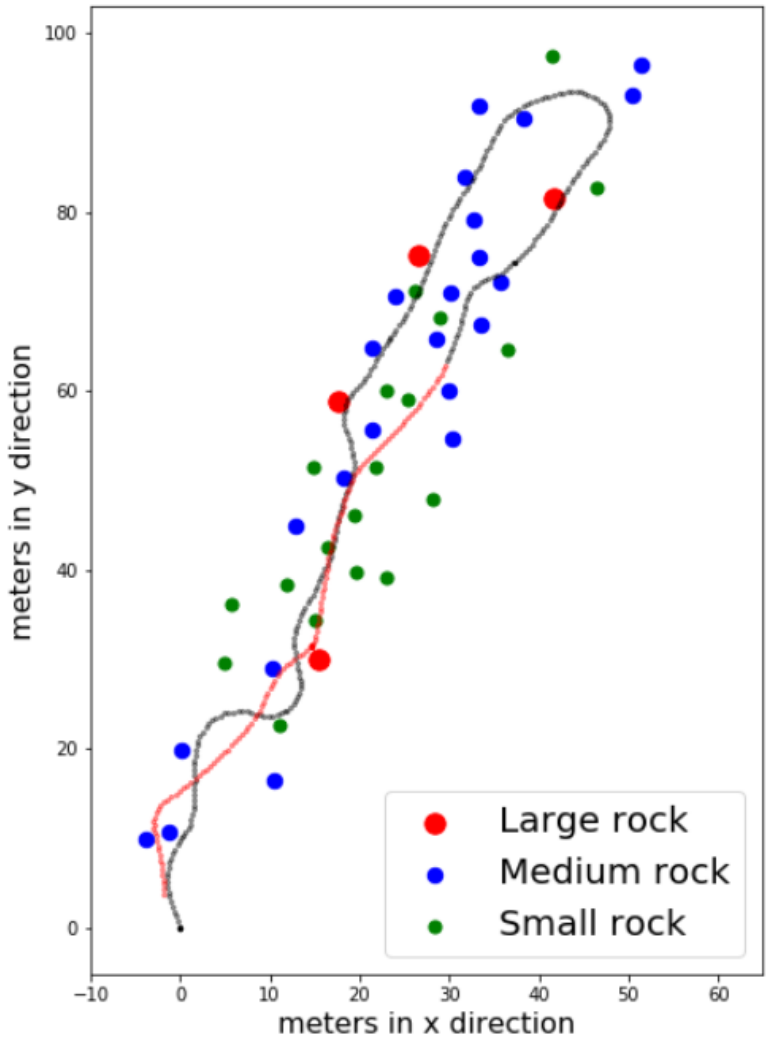
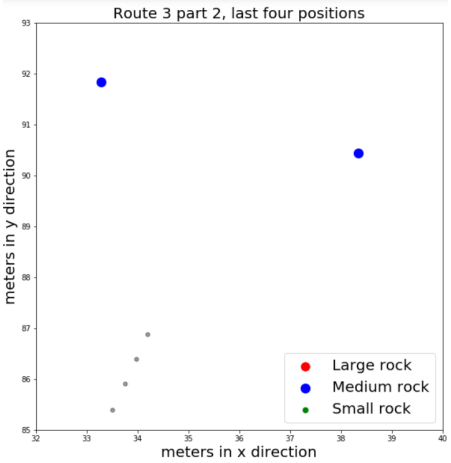


Figure 22: Complete ground truth of route.

Figure 22 displays the ground truth of the whole route. The route starts at the origin. It overlaps for the first and last quarter, for this reason some parts of the route are displayed in red instead of black.



(a) The last location of part 2 is the black dot closest to the rocks.

(b) The last image taken in part 2.

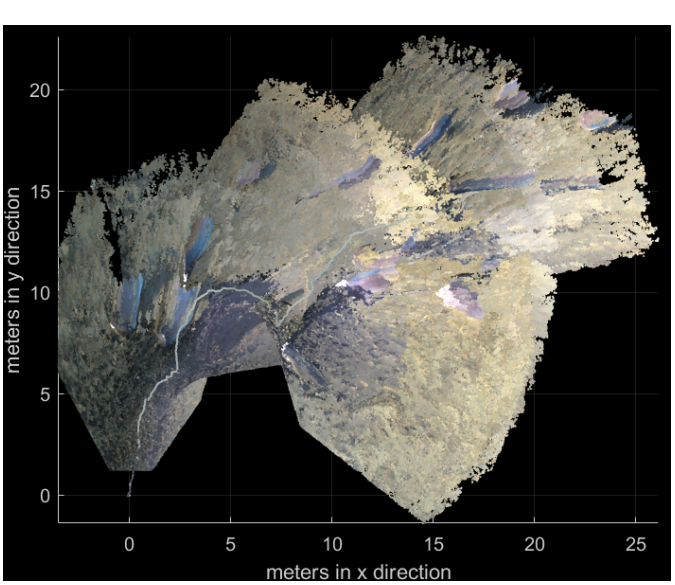
Figure 23: Last location of part 2 according to the GPS (a) and according to the camera footage (b).

The accuracy of the ground truth should be high. Millimeter accuracy is guaranteed for over 95% of the dataset's traverse, as it is based on RTK GPS. Yet, as illustrated by Figure 23, there is a discrepancy between the images and the GPS data. The last image of part 2 appears to be closer to the rock than it should be according to the GPS. Because the GPS is guaranteed to have high accuracy is the most likely explanation for this discrepancy a fault with the timestamps.

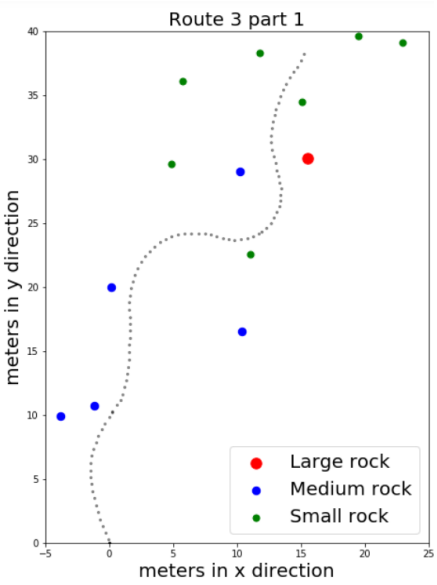
9.2 Visual SLAM Using Point Clouds

The results of visual SLAM are presented in the five parts in which the dataset was provided. Only the most notable results are displayed here, all others can be found in appendix B.

9.2.1 Part 1



(a) Map en route of part 1 according to visual SLAM. The route can be seen as a gray line.



(b) Map and route of part 1 in ground truth.

Figure 24: Part 1 according to visual SLAM(a) and ground truth (b).

The map and route of part 1 according to visual SLAM can be seen in Figure 24 (a). Figure 24 (b) is what the map and route should look like according to the ground truth. At the beginning of the route are two medium sized rocks. These rocks, though stretched, are still recognisable and in the correct position, this is not the case for rocks further along the route. The computed route diverges significantly from the ground truth, this means that computing the mean distance between ground truth and visual SLAM becomes obsolete. There seems to be an overlaying problem. A likely cause of this is an accumulating error, due to which both the map and the location of the HDPR become more inaccurate as the route continues.

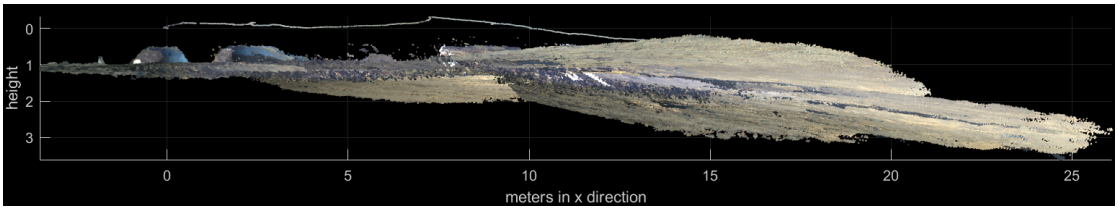
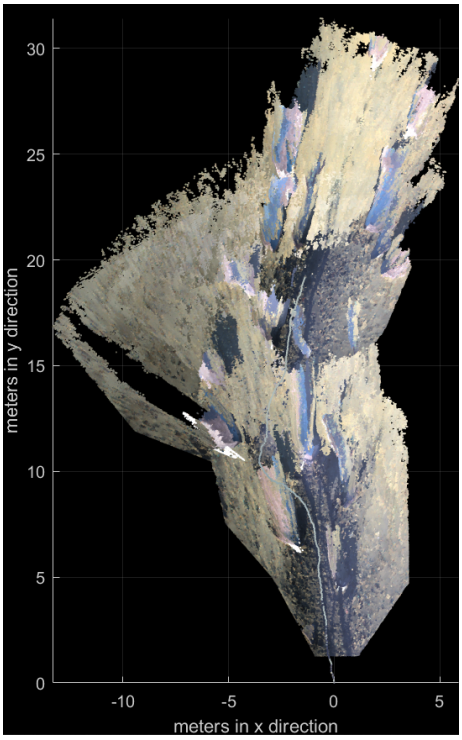


Figure 25: Frontal view, the route can be seen floating above the scene.

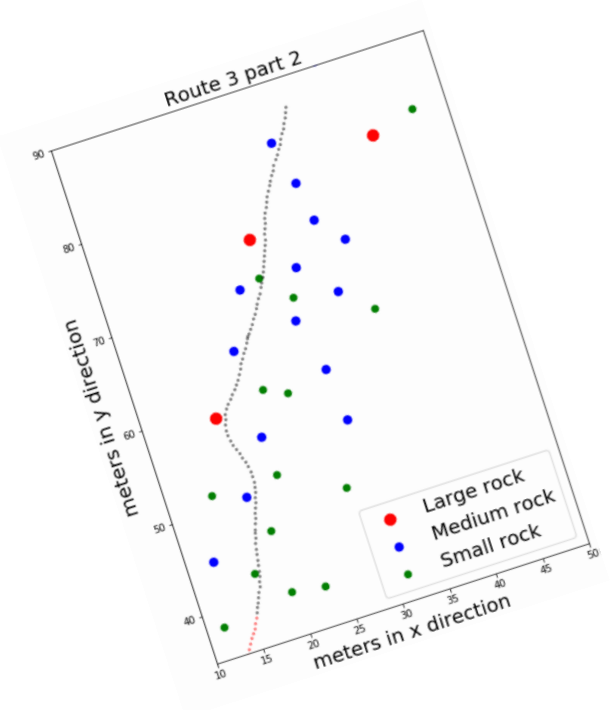
In the frontal view of part 1, as seen in Figure 25, the contours of the first two medium sized rocks can be seen between 0 and 4 meters in the x direction. The left side of the image represents the beginning of the route, as the route continues the contours of the rocks become unclear. Furthermore, there is a relatively large height difference on the right side of the image, which is another indicator of incorrect image overlaying. There should be hardly any height difference because this dataset was recorded on the Dutch seaside.

Side views of part 2 can be found in appendix B.

9.2.2 Part 2



(a) Map en route of part 2 according to visual SLAM. The route can be seen as a gray line.



(b) Map and route of part 2 according to ground truth.

Figure 26: Map and route of part 2 according to visual SLAM(a) and ground truth (b).

The map and route of part 2 according to visual SLAM can be seen in Figure 26 (a). Figure 26 (b) is what the map and route should look like according to the ground truth. When comparing image (a) to (b) it is not possible to identify individual rocks, because of their stretched appearance. This indicates another merging problem, which could be caused by a faulty location estimation. A wrong estimation of the camera position causes a mismatching problem when trying to overlay images. However, the route calculated by visual SLAM, (a), is very similar to the ground truth route, (b), likely due to this part being relatively straight. The assumption could be made that a better location estimation should result in less stretching. However, even with the better estimation in the straight part of the route the same problem remains. This indicates that the location estimation is not solely responsible for the merging problem.

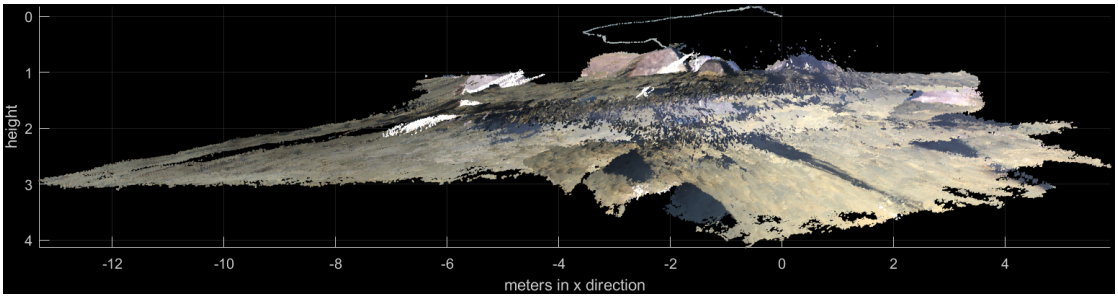
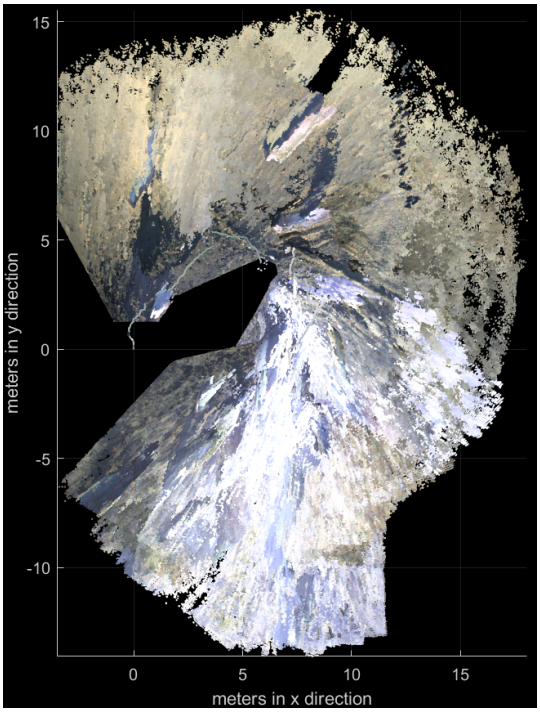


Figure 27: Frontal view, the route can be seen floating above the scene in gray.

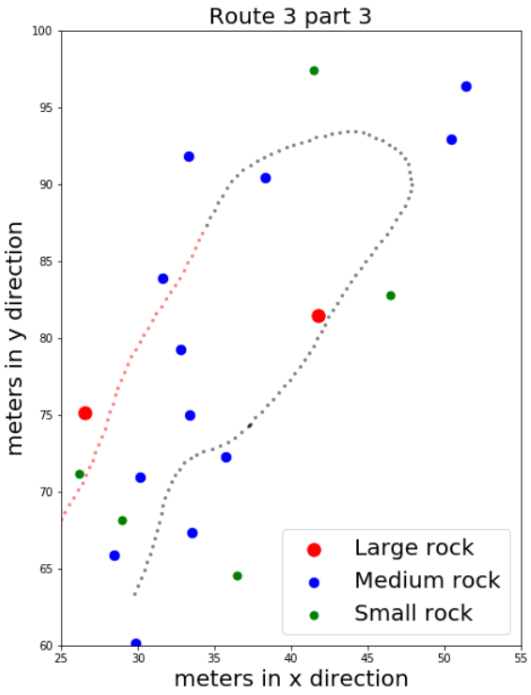
The frontal view of part 2, as seen in Figure 27, shows that there is a lot of height difference between the point clouds. This height difference illustrates the merging problem once more.

Side views of part 2 can be found in appendix B.

9.2.3 Part 3



(a) Map en route of part 3 according to visual SLAM. The route can be seen as a gray line.



(b) Map and route of part 3 according to ground truth.

Figure 28: Map and route of part 3 according to visual SLAM(a) and ground truth (b).

The map and route of part 3 according to visual SLAM can be seen in Figure 28 (a). Figure 28 (b) is what the map and route should look like according to the ground truth. Part 3 of the route has blooming problems. Figure 28 shows that these problems start halfway through the route, when the HDPR makes a u-turn to go southward. The blooming makes the map more chaotic, causing white sections throughout the map. Furthermore the route according to visual SLAM follows the outlines of the ground truth route until the HDPR makes the turn, after this the route is no longer visible.

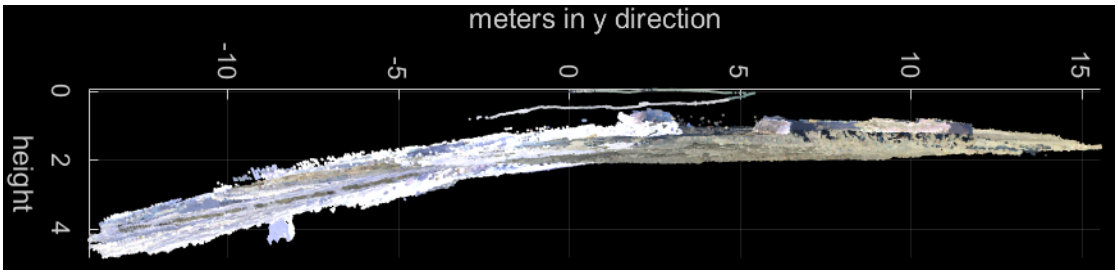


Figure 29: Left side view of part 3, the route can be seen floating above the scene.

Figure 29 displays the left side view of the scene depicted in Figure 28 (a). The blooming also appears to have affected the merging of point clouds, resulting in an inaccurate overlay and more height difference when blooming is present.

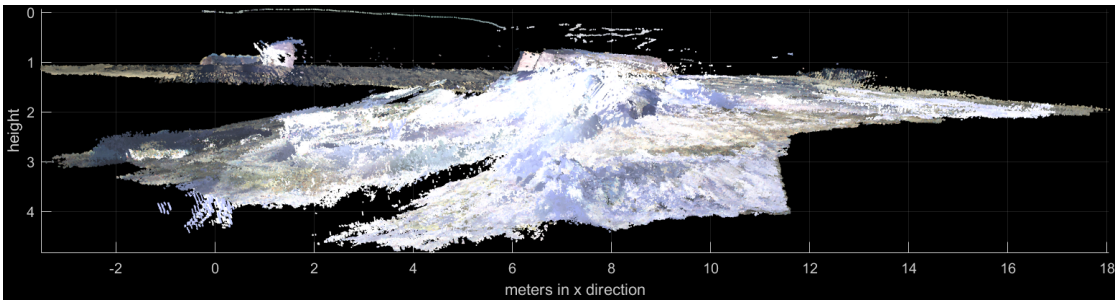
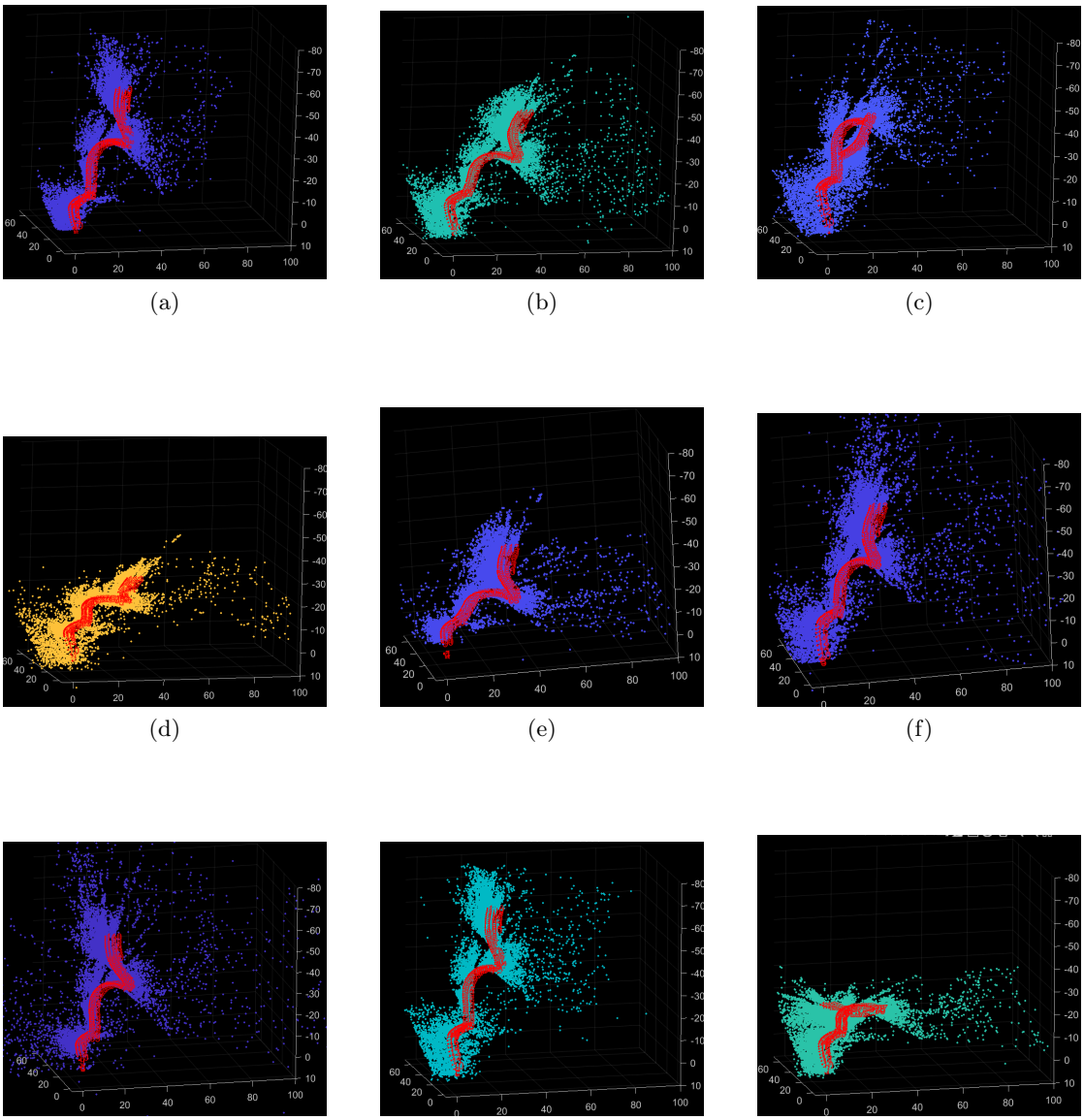


Figure 30: Frontal view of part 3, the route can be seen floating above the scene.

Figure 30 displays the frontal view of the scene depicted in Figure 27 (a). On the upper left side of the figure the contour of a rock is visible. In the middle of the route the image quality is lower due to blooming once more.

Part 4 and 5 consist mostly of images with blooming therefore their results are very similar to part 3 after the HDPR changes direction. All images of part 4, 5, and the right side view of part 3 can be found in appendix B.

9.2.4 Structure from Motion



(a) (b) (c)

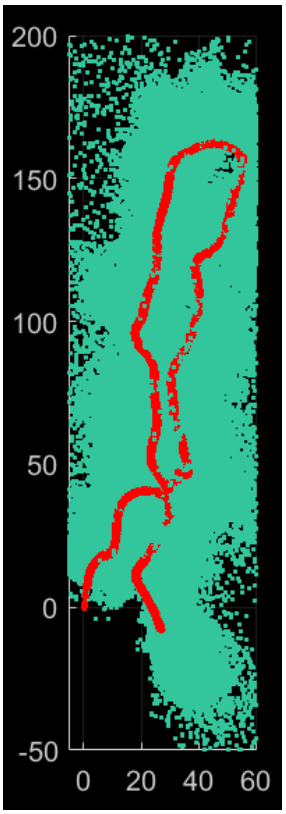
(d) (e) (f)

(g) (h) (i)

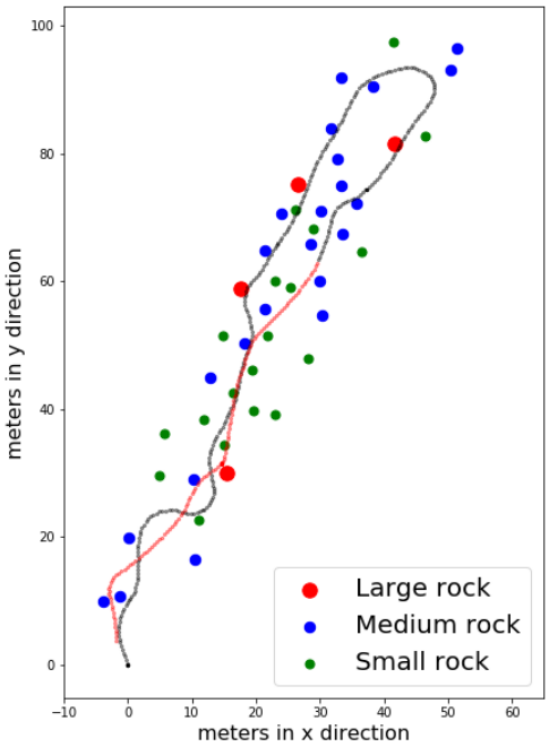
Figure 31: The route and map of part 1 are rendered nine different times.

Figure 31 displays the difference in results when computing the map and route via Structure from Motion. Each computation has a different result. (a), (c), and (i) display a route which gets lost, this is likely caused by the limited number of discernible objects in the images. In general deviation is more likely in the last section of the route. In

order to limit the possibility of the route going astray the total dataset is divided into ten equal parts instead of the five original parts.



(a) Map en route according to Structure from Motion line.

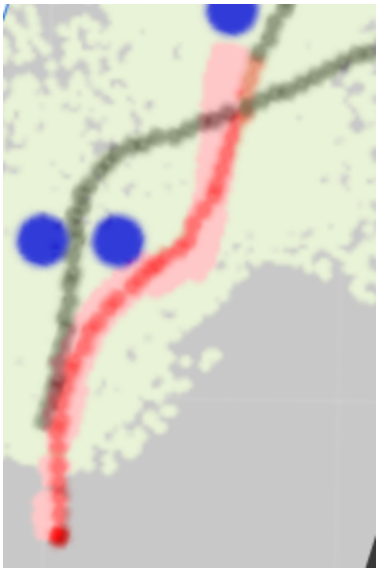


(b) Map and route in ground truth. Route can be seen in red.

Figure 32: Route according to Structure from Motion(a) and ground truth (b).

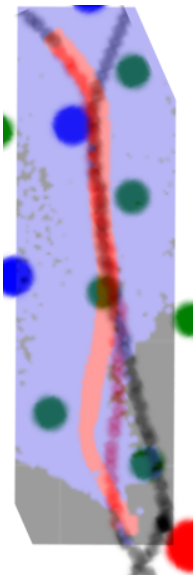
Figure 32 displays the route and map according to Structure from Motion (a) and according to ground truth (b). The route according to Structure from Motion is very similar to the ground truth. When observing Figure 32 (a) it appears that Structure from Motion handles straight sections better than parts where there are more corners such as the beginning of the route.

The routes and maps of all individual parts can be found in appendix C.

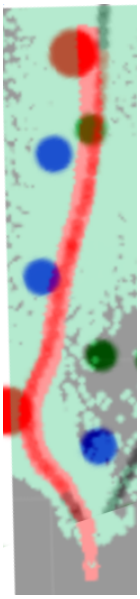


(a) Part 1.

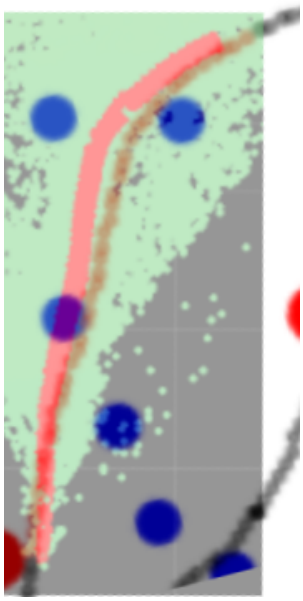
(b) Part 2.



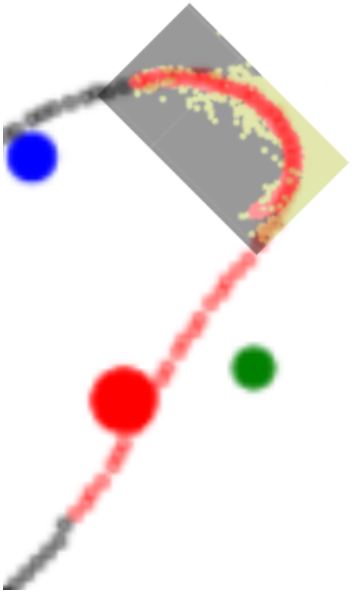
(c) Part 3.



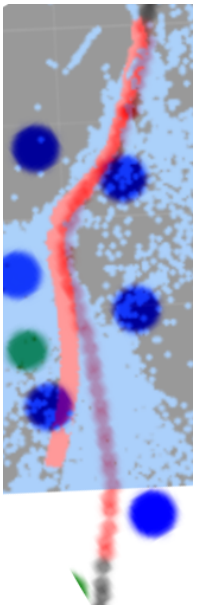
(d) Part 4.



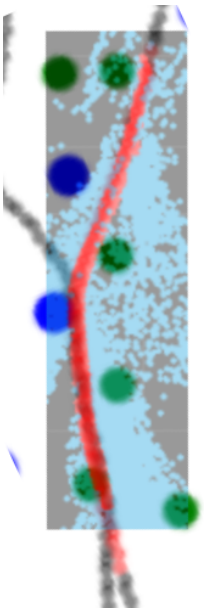
(e) Part 5.



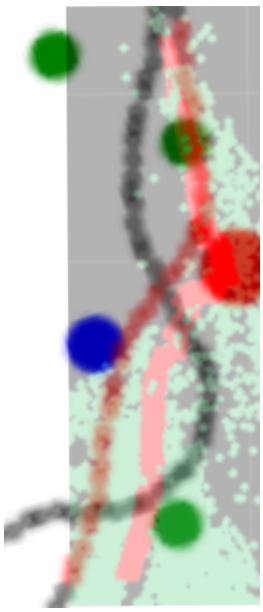
(f) Part 6.



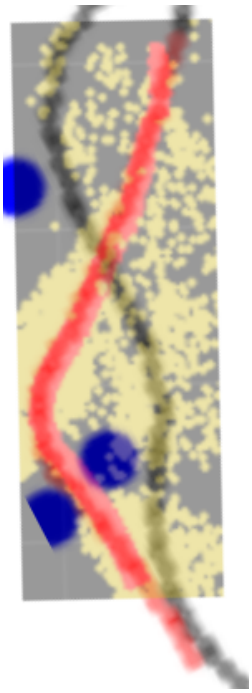
(g) Part 7.



(h) Part 8.



(i) Part 9.



(j) Part 10.

Figure 33: Overlay of the ground truth and the path according to Structure from Motion of each part.

The route created by Structure from Motion is compared to the ground truth by overlaying them with the ground truth, as can be seen in Figure 33. Some parts such as part 4, 5, 8, and 10 are very similar to the ground truth. However some parts deviate quite substantially, such as part 1, 2, 6, and 7. Because these are the sections with the most curves is the deviation likely caused by rough corner handling in the estimation matrix. In part 6 and 7 is it evident that the Structure from Motion routes are significantly shorter than the ground truth. These are parts with a lot of blooming, therefor is it probable that blooming has a negative influence on the location estimation.

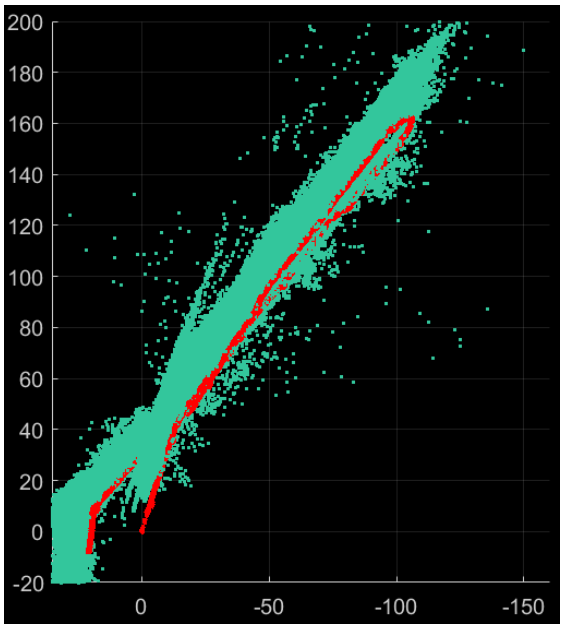
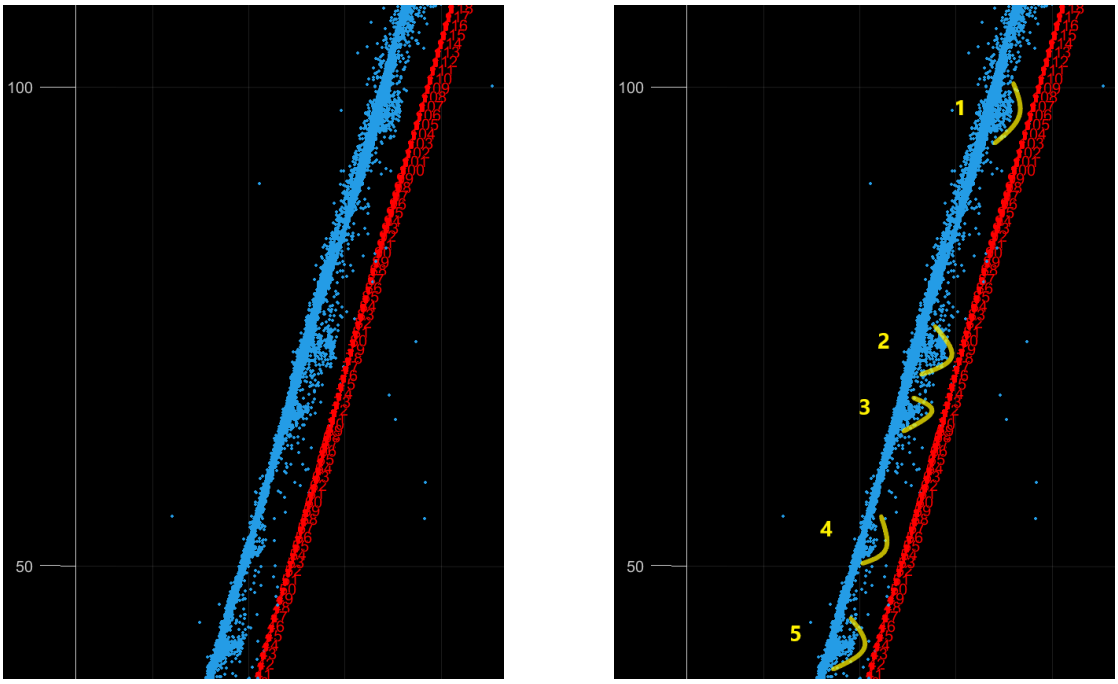


Figure 34: Side view of route

The assumption that the angle of the route would be $\frac{1}{10}\pi$ is wrong, when looking at Figure 34 it is visible that the angle is not consistent throughout. This is against the expectations that were based on the camera angle.



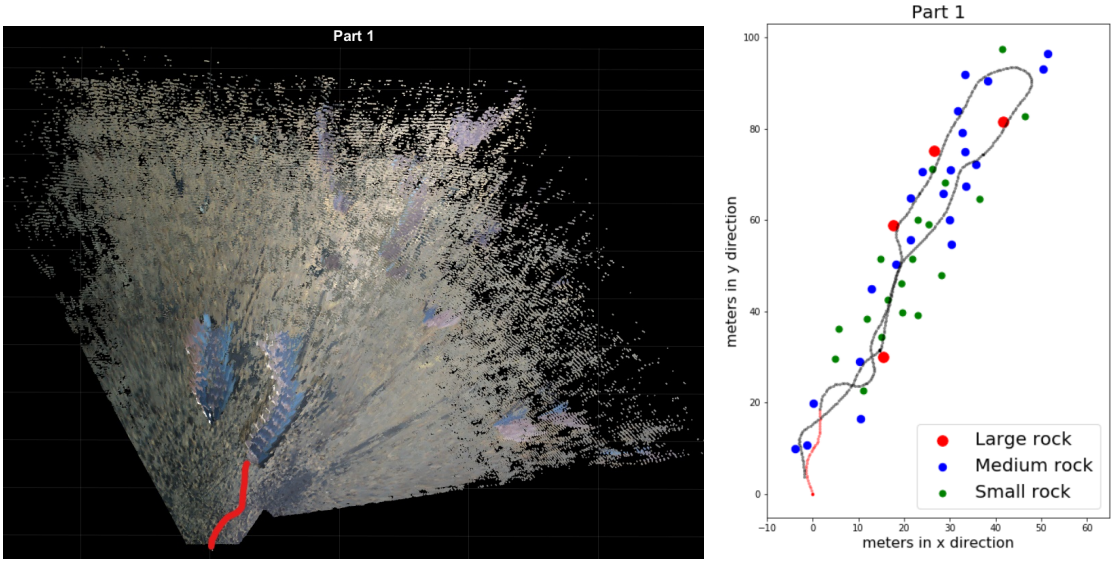
(a) (b)

Figure 35: Example of what rocks look like from the side in Structure from Motion.

The map of Structure from Motion consists of dots without a meaningful colour. This makes it difficult to discern the rocks. An example of what rocks look like in the Structure from Motion map can be seen in Figure 35. Their rough outlines are visible, but it is difficult to identify individual rocks.

9.2.5 Combined Method

9.2.6 Part 1



(a) Map and route of part 1 according to the combined method. Originally the route is a white dotted line, but for visualisation purposes it is accentuated in red. The original image can be found in the appendix.

(b) The ground truth route of part 1 is displayed in red.

Figure 36: Map and route of part 1 according to the combined method (a) and ground truth (b).

Figure 36 (a) shows what the map and route of part 1 is according to the combined method. The rocks appear stretched, which is an indication that there is a problem with the merging of point clouds. The route is identical to the Structure from Motion route, meaning the quality is also the same.

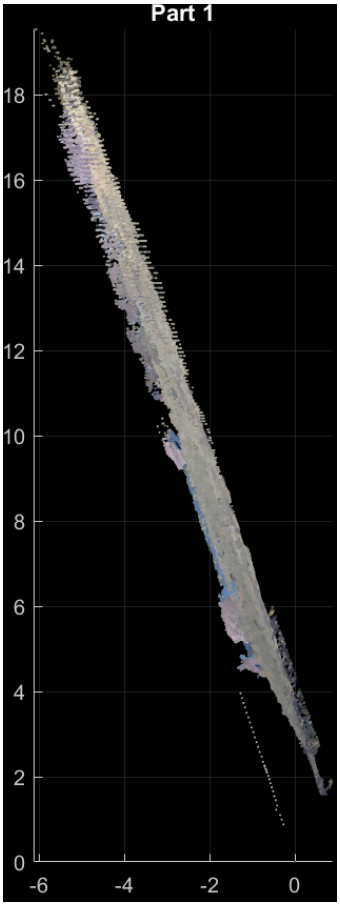
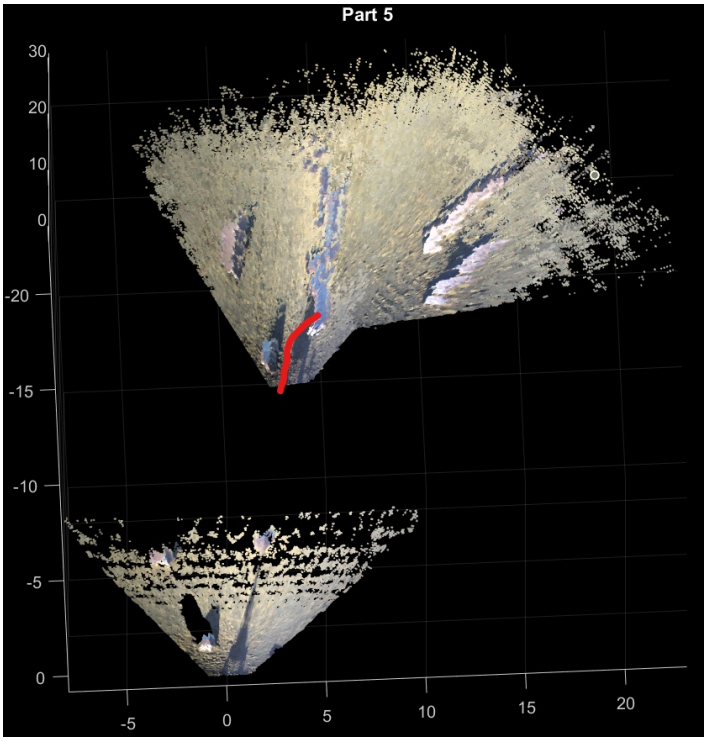


Figure 37: Left side view of part 1

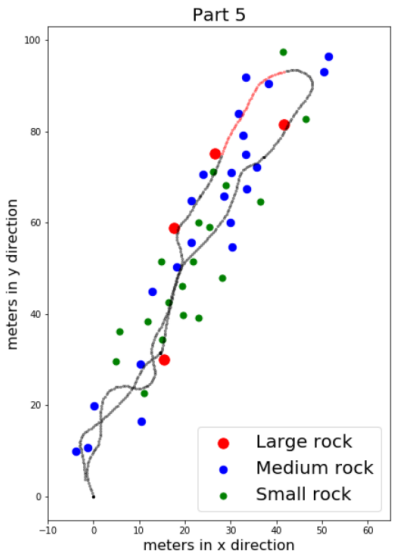
The side views of Figure 36 (a) are displayed in Figure 37. The scenes are angled, which is expected due to the angle of the camera. The height problems encountered in visual SLAM maps are not present, this indicates that the rotation of the point clouds is correct.

All images of part 2, 3, 4, and the right side view of part 1 can be found in appendix D.

9.2.7 Part 5



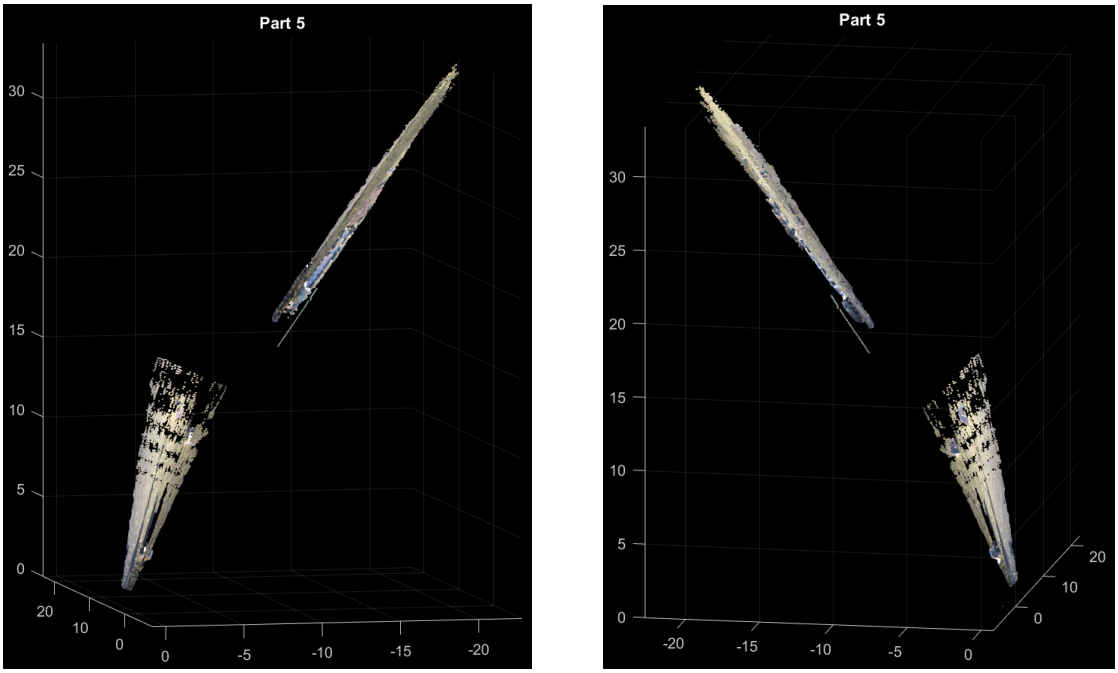
(a) Map and route of part 5 according to the combined method. Originally the route is a white dotted line, but for visualisation purposes it is accentuated in red. The original image can be found in the appendix.



(b) The ground truth route of part 5 is displayed in red.

Figure 38: Map and route of part 5 according to the combined method (a) and ground truth (b).

Figure 38 (a) displays the map and route of part 2 according to the combined method. The figure shows two scenes rather than one. The likely cause is inaccurate location estimation, which was not accurately filtered out. The upper point cloud is the depiction of the route and map. The rocks are stretched but can be identified, especially when compared to the ground truth in Figure 38 (b).

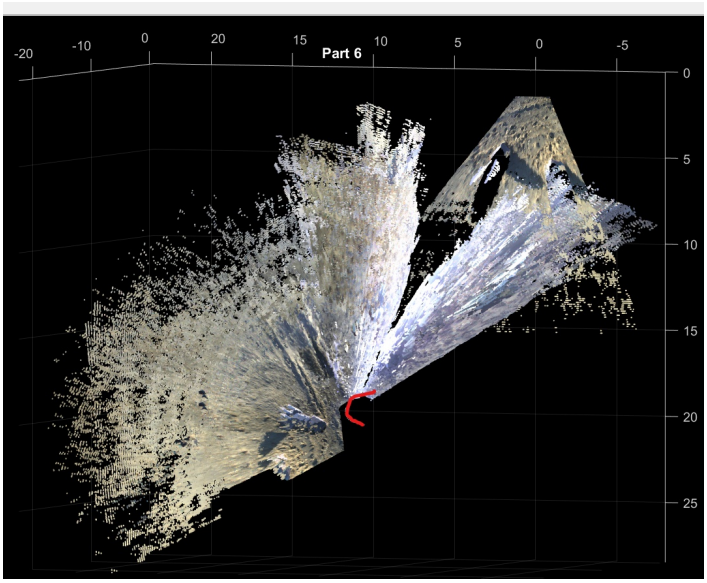


(a) (b)

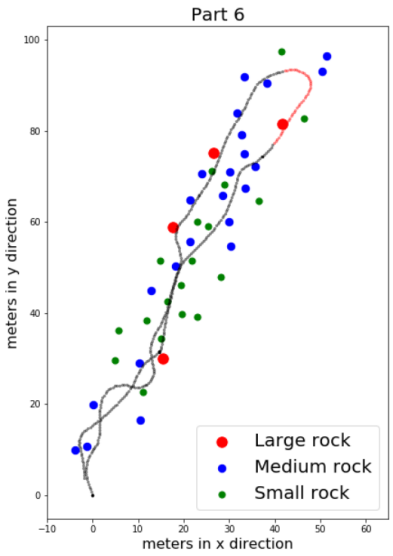
Figure 39: Left(a) and right(b) side view of part 5.

The side views of part 5, depicted in Figure 39, show that there is little height difference in the final merged point cloud.

9.2.8 Part 6



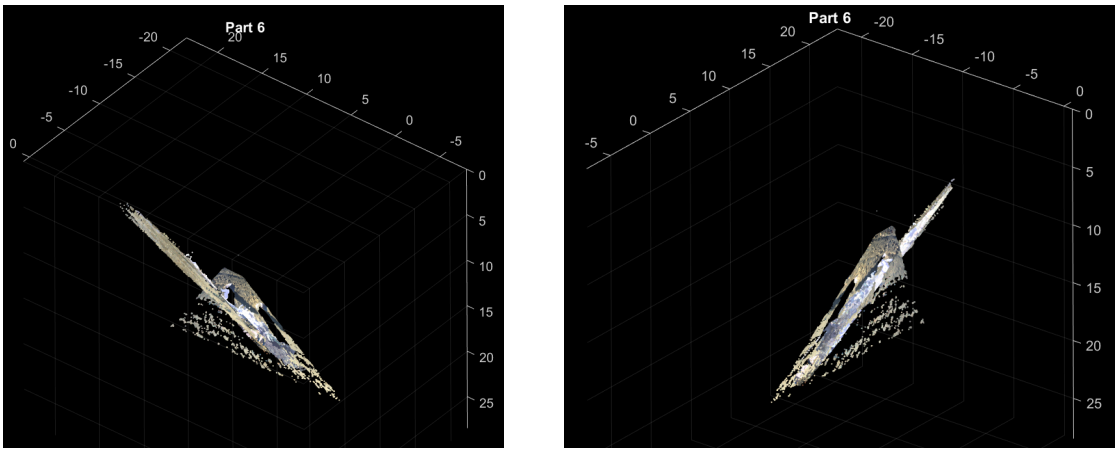
(a) Map and route of part 6 according to the combined method. Originally the route is a white dotted line, but for visualisation purposes it is accentuated in red. The original image can be found in the appendix.



(b) The ground truth route of part 6 is displayed in red.

Figure 40: Map and route of part 6 according to the combined method (a) and ground truth (b).

Figure 40 displays part 6, which is the section where the blooming images start to occur. In the upper right corner of Figure 40 (a) is a separate point cloud visible. This is caused by noise that was not accurately recognised and filtered out. All following parts have blooming and their results are very similar to the second half of part 6.



(a) (b)

Figure 41: Left(a) and right(b) side view of part 6.

Figure 41 displays the side view of part 6. Due to the noise it is difficult to visualise, but there is little height difference in the map representation.

The result form all following parts can be found in appendix D.

10 Discussion

10.1 Ground Truth

The Used GPS approach has a very high accuracy level, with at most a 9 millimeter deviation. However, it appears that the locations in images do not match the locations of the GPS. Considering that it is very unlikely that the GPS itself has this level of inaccuracy, is the most likely cause a problem with the timestamps.

Because the timestamps of the datasets might be incorrect is it difficult to make a direct comparison between the ground truth location and the locations according to the SLAM techniques.

10.2 Visual SLAM

The maps calculated through visual SLAM show stretched rocks. This indicates a problem with the merging of point clouds. This problem is likely to do with the ICP algorithm. This algorithm does not always perform flawlessly, nor does it correct its errors, leading to an accumulation error. Additionally, the ICP algorithm performs better when images have a larger amount of distinctive features, sand is not that distinctive. This dataset is recorded on the Dutch seaside and should therefore display a terrain which is fairly level. However, the resulting maps show a terrain with a lot of height difference, this indicates that the ICP algorithm incorrectly determines the orientation of the cameras. Overall the routes and maps are inaccurate, but there are occasions, such as straight sections, where they appear to be more correct.

A possible solution is to apply a Kalman filter [14]. A Kalman filter uses two or more location estimations and bases the ultimate location of the camera upon the certainty of these estimations. The Katwijk beach dataset includes an Internal Measurement Unit and an odometry recording. By incorporating one of these files into the location estimation via a Kalman filter, a more correct location estimate could be made. The assumption is that a more accurate location will lead to a better point cloud merge.

10.3 Structure from Motion

The route calculated through Structure from Motion shows great similarity to the ground truth. The position of the rocks is unclear, because the map lacks distinctive features and meaningful color.

Even though the route is more similar to the ground truth than the visual SLAM route, it still deviates. The bundle adjustment could have improved the overall map quality, but because the route had to be split up into ten parts, it was unable to make use of loop closure, which could have improved both the route and the map. The application of bundle adjustment, though limited, is a likely reason why this approach suffers less from accumulation errors than visual SLAM. However, it does not solve the lack of discernible objects. The sections of the route with more corners are less accurate, this indicates that there is rough corner handling by the estimation matrix.

Blooming also causes deviation with this method, the route quality lessens with blooming images. This could also be one of the causes of the deviation from ground truth. The sections where blooming is present have less key points, this makes it more difficult to merge the point clouds. Again, the Kalman filter could offer a possible solution for both the blooming and the limited features problem, for similar reasons as before.

The coordinate system of Structure from Motion has not been converted to meters for this approach, which limits its comparability to ground truth in terms of distance. However, it is apparent that the route of Structure from Motion has a closer resemblance to the ground truth than the visual SLAM route.

10.4 Combined Method

The route in this method is the same as the route in Structure from Motion and has therefore the same qualities and shortcomings. The rocks in the maps produced by the combined method appear to have the same flaw as the rocks from visual SLAM, they are stretched and therefore difficult to identify. This indicates that there is still a merging problem. This could be caused by the ICP algorithm, but another likely cause is that the camera coordinate system from Structure from Motion was not correctly translated to a meter coordinate system.

However the map from this method is arguably more comprehensible than the map from visual SLAM, because there is less height difference between the point clouds. This indicates that the orientation of the cameras was correctly determined by Structure from Motion. In some images there are extra point clouds visible, this is noise, likely caused by a faulty location estimate from Structure from Motion which was not correctly filtered. A possible solution would be another processing step which removes point clouds of which the location deviates too much from the previous location.

Each technique has benefits and disadvantages but overall the desired outcome, a completely accurate map and accurate location description, was not achieved.

11 Conclusion

In summary, the goal of this thesis was to study the routes and maps produced by visual SLAM, Structure from Motion, and a combination of both techniques. This was done by computing their individual results and comparing their quality relative to each other and to the ground truth.

Results indicated that none of the techniques are ideal. The visual SLAM approach had, when applied to small numbers of images, a clear map representation. But when it is applied to a larger dataset the location of the HDPR is inaccurate due to an accumulation error which causes the images to no longer merge without fault. The resulting map therefore looks stretched and with each added image the map becomes less clear. The route depicted by Structure from Motion, according to the results from this thesis, appears to have a closer resemblance to ground truth. However, the map is made of dots without a meaningful color. This makes it difficult to comprehend the map from a human standpoint. Also, the route needed to be separated in smaller parts in order to prevent the HDPR from getting lost. This is because the images have very little distinctive features. All sand looks the same to the HDPR. This led to faulty locations estimations which could not be rectified.

By combining the approaches the goal was to use the best of both, namely the comprehensible map from visual SLAM and the location description from Structure from Motion. The main problem with applying this technique was transforming the coordinate system from Structure from Motion into a coordinate system in meters. In the results the images do not merge well together and rocks are depicted multiple times. This is probably caused by the coordinates system transformation malfunctioning. However, the map is still easier to comprehend than the map from Structure from Motion and, while not perfect, is also more accurate and has less height difference than the map from visual SLAM.

Overall the techniques did not produce the intended outcome, which was both an accurate map and route. But the results indicate possible enhancement which could improve the quality. One of those enhancements could be a Kalman filter which prevents the HDPR from getting lost due to the indistinguishable surroundings. Another could be taking a different approach for transforming the Structure from Motion coordinate system into a meter coordinate system.

11.1 Future Work

The Katwijk beach dataset consists of multiple sensors which could be used for SLAM. In this thesis the choice was made to focus on the stereo cameras, however LiDAR could very well be more suited for Martian terrain applications. LiDAR stands for Light Detection And Ranging and is a type of sensor which uses laser pulses to make a 3D image. LiDAR might be more suited because it does not rely on a stable light source to produce quality images. This would solve the blooming problems encountered in this thesis. In fact, LiDAR does not rely on a light source at all, which makes it possible to also explore terrain on the dark side of Mars, underground, or in craters where there is little light.

The author of the article which describes the Katwijk beach dataset, Robert A Hewitt, has done his PhD thesis on bundle adjustment for LiDAR, applying this technique to the dataset could give interesting results.

Furthermore the techniques used in this thesis could be expanded on by applying the earlier mentioned Kalman filter [14]. The dataset includes odometry data and an Internal Measurement Unit which should make this possible. However, due to incomplete data descriptions and unusual sensors this was not applied in this thesis. This approach therefore requires more research into how to use the datasets and would require contact with the producers.

While producing the visual SLAM point clouds, there are a relatively large number of variables, including but not limited to downsizing for merging and how to merge the final point clouds. More research can be done on deciding which variables function optimally.

It is likely that the camera coordinate system was not correctly translated to a meter coordinate system. An approach which also takes the possible warping of the terrain into account and which better corrects for noise could potentially improve the quality.

The map produced by Structure from Motion is difficult to understand and can be improved by creating a more dense point cloud. This can be achieved by tracking key points across frames. This is an enhancement because it also stores information which describes the location of key points relative to each other. Making it possible to produce a point cloud with more accuracy and more key points, allowing for a more comprehensive map and a better route. A possible approach for this is the Kanade-Lucas-Tomasi (KLT) algorithm [19][26].

In some articles Structure from Motion was used in such a way that there was RGB data [7]. RGB data could improve map legibility, which would make merging visual SLAM and Structure from Motion obsolete. Research can be done into how to possibly apply this to the dataset.

For this thesis only the results of route 3 were computed. Route 1 and 2 are driven with higher speeds and have less corners. Their results are assumed to be similar to the results of part 2 of route 3. This because part 2 is also relatively straight, but it could be interesting to analyse if this is correct.

Sadly the opportunity of loop closure could not be tested in this thesis due to computation limits. With more computing power the influence of loop closure on the route could be studied, possibly resulting in more accuracy.

References

[1] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (surf). *Computer Vision and Image Understanding*, 110(3):346–359, 2008.

[2] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *European conference on computer vision*, page 407. Springer, 2006.

[3] Evangelos Boukas, Robert A Hewitt, Marco Pagnamenta, Robin Nelen, Martin Azkarate, Joshua A Marshall, Antonios Gasteratos, and Gianfranco Visentin. Hdpr: A mobile testbed for current and future rover technologies. *i-SAIRAS 2016*, Jun 2016.

[4] Hongyu Chen, Zhijie Yang, Xiting Zhao, Guangyuan Weng, Haochuan Wan, Jianwen Luo, Xiaoya Ye, Zehao Zhao, Zhenpeng He, Yongxia Shen, and et al. Advanced mapping robot and high-resolution dataset. *Robotics and Autonomous Systems*, page 1–2, Jun 2020.

[5] Yang Chen and Gérard Medioni. Object modelling by registration of multiple range images. *Image and Vision Computing*, 10(3):145–155, 1992.

[6] Paul Timothy Furgale. *Extensions to the visual odometry pipeline for the exploration of planetary surfaces*. Citeseer, 2011.

[7] Susie Green, Andrew Bevan, and Michael Shapland. A comparative assessment of structure from motion methods for archaeological research. *Journal of Archaeological Science*, 46:173–181, 2014.

[8] C. Harris and M. Stephens. A combined corner and edge detector. *Proceedings of the Alvey Vision Conference*, page 147–151, Aug 1988.

[9] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*, page 302. Cambridge University Press, ISBN: 0521540518, second edition.

[10] Robert Hewitt et al. *Intense Navigation: Using Active Sensor Intensity Observations To Improve Localization and Mapping*. PhD thesis, Queen's University at Kingston, 2018.

[11] Robert A Hewitt, Evangelos Boukas, Martin Azkarate, Marco Pagnamenta, Joshua A Marshall, Antonios Gasteratos, and Gianfranco Visentin. The Katwijk beach planetary rover dataset. *The International Journal of Robotics Research*, 37(1):3–12, 2018.

[12] H. Hirschmuller. Accurate and efficient stereo processing by semi-global matching and mutual information. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR05)*, page 807–814, 2005.

[13] L. Javernick, J. Brasington, and B. Caruso. Modeling the topography of shallow braided rivers using structure-from-motion photogrammetry. *Geomorphology*, 213:166–182, May 2014.

[14] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. 1960.

[15] Kurt Konolige. Small vision systems: Hardware and implementation. *Robotics Research*, page 203–212, 1998.

[16] Kenneth Levenberg. A method for the solution of certain non-linear problems in least squares. *Quarterly of Applied Mathematics*, 2(2):164–168, 1944.

[17] Hugh Christopher Longuet-Higgins, U Öpik, Maurice Henry Lecorney Pryce, and RA Sack. Studies of the jahn-teller effect. ii. the dynamical problem. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 244(1236):1–16, 1958.

[18] David G Lowe. Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pages 1150–1157. Ieee, 1999.

[19] Bruce D Lucas, Takeo Kanade, et al. An iterative image registration technique with an application to stereo vision. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*. Vancouver, British Columbia, Aug 1981.

[20] WJ Maegley and DP Diederich. Martian sandstorms and their effects on the 1975 viking lander system. *Journal of Testing and Evaluation*, 3(5):380–388, 1975.

[21] D Marquardt. A method for the solution of certain problems in least-squares. *SIAM J. Appl. Math*, 11(431-441), 1963.

[22] David Marr and Ellen Hildreth. Theory of edge detection. *Proceedings of the Royal Society of London. Series B. Biological Sciences*, 207(1167):187–217, 1980.

[23] Adam Mathews and Jennifer Jensen. Visualizing and quantifying vineyard canopy using an unmanned aerial vehicle (uav) collected high density structure from motion point cloud. *Remote Sensing*, 5(5):2164–2183, 2013.

[24] Marcos Nieto, Carlos Cuevas, Luis Salgado, and Narciso García. Line segment detection using weighted mean shift procedures on a 2d slice sampling strategy. *Pattern Analysis and Applications*, 14(2):149–163, 2011.

[25] Niko Sünderhauf, Kurt Konolige, Simon Lacroix, and Peter Protzel. Visual odometry using sparse bundle adjustment on an autonomous outdoor vehicle. In Paul Levi, Michael Schanz, Reinhard Lafrenz, and Viktor Avrutin, editors, *Autonome Mobile Systeme 2005*, pages 157–163, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[26] Carlo Tomasi and Takeo Kanade. Shape and motion from image streams: a factorization method. *Proceedings of the National Academy of Sciences*, 90(21):9795–9802, 1993.

[27] Khalid Yousif, Alireza Bab-Hadiashar, and Reza Hoseinnezhad. An overview of visual odometry and visual slam: Applications to mobile robotics. *Intelligent Industrial Systems*, 1(4):289–311, 2015.

images/

A Ground Truth

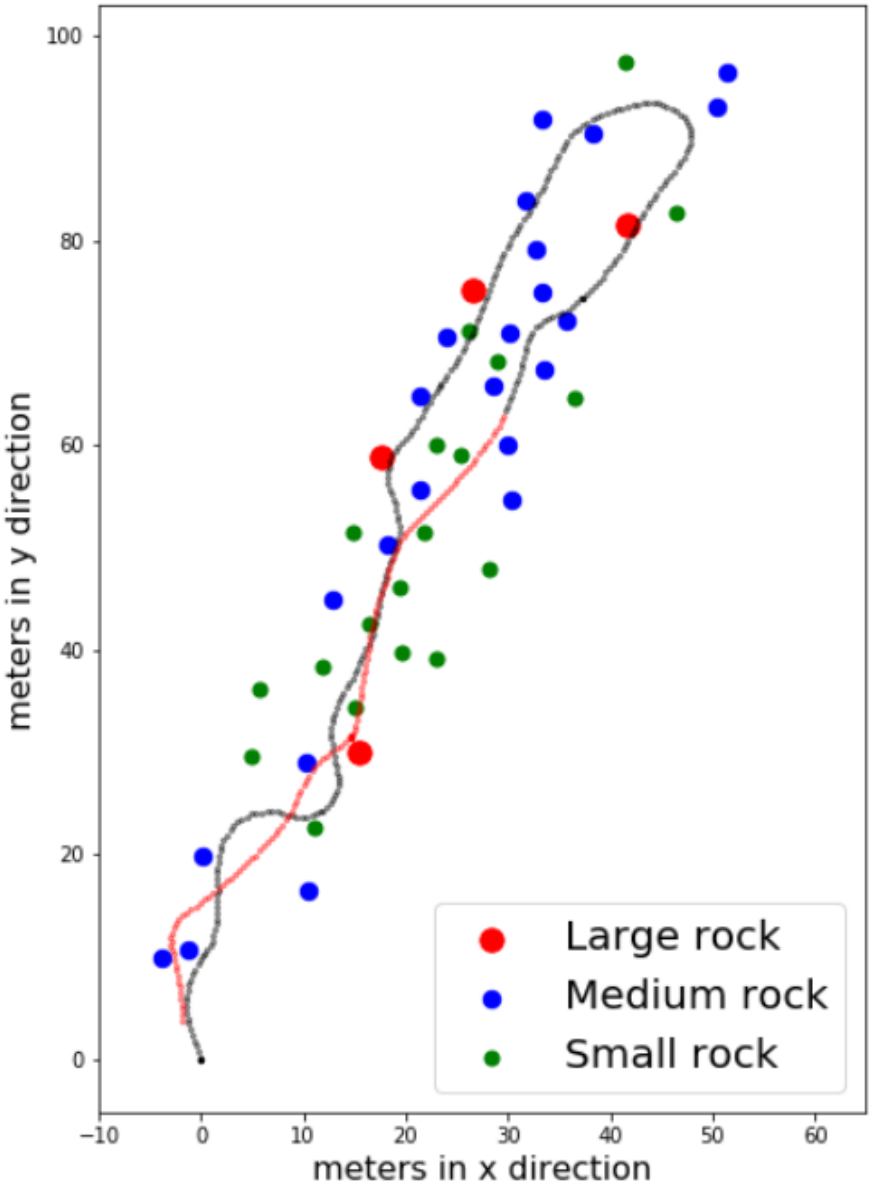
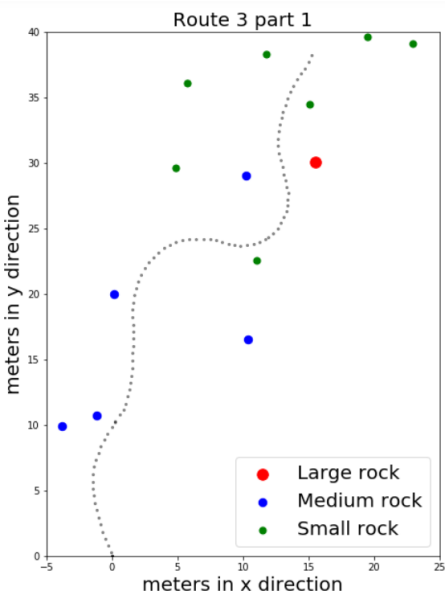


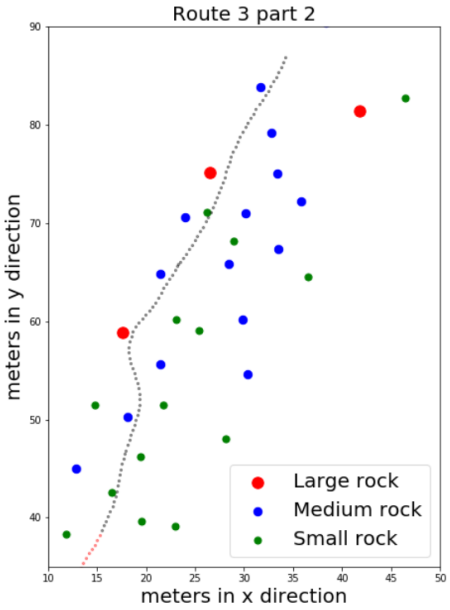
Figure 42: Complete ground truth of route. Some parts of the route are displayed in red for clarity reasons.

The following five images display the ground truth of the five parts of the route. The last positions of the previous part are displayed in red to visualise how the individual

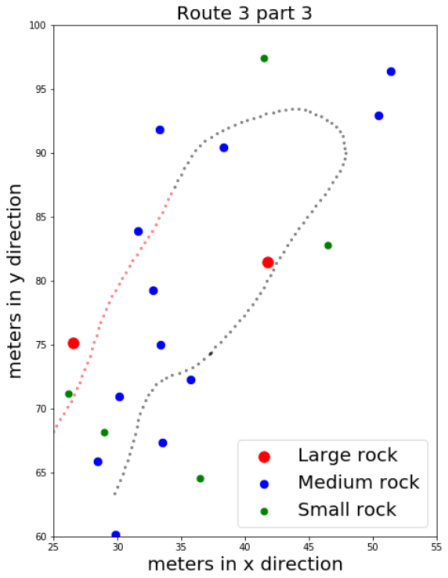
parts fit in the route.



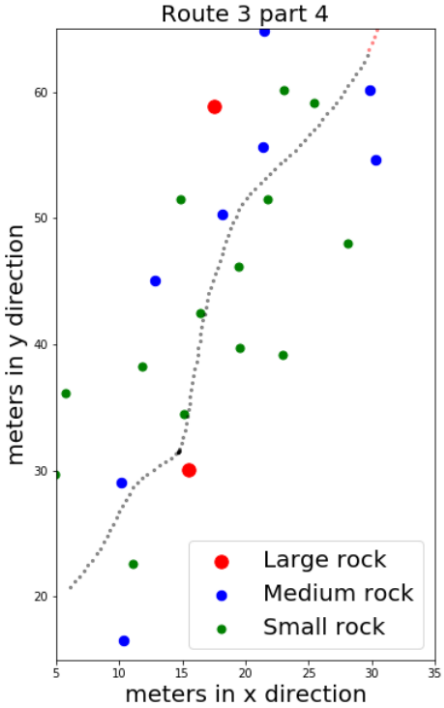
(a) Ground truth of part 1.



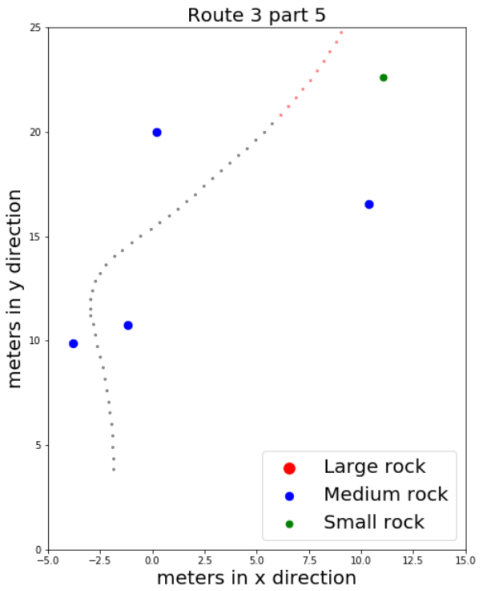
(b) Ground truth of part 2.



(c) Ground truth of part 3.

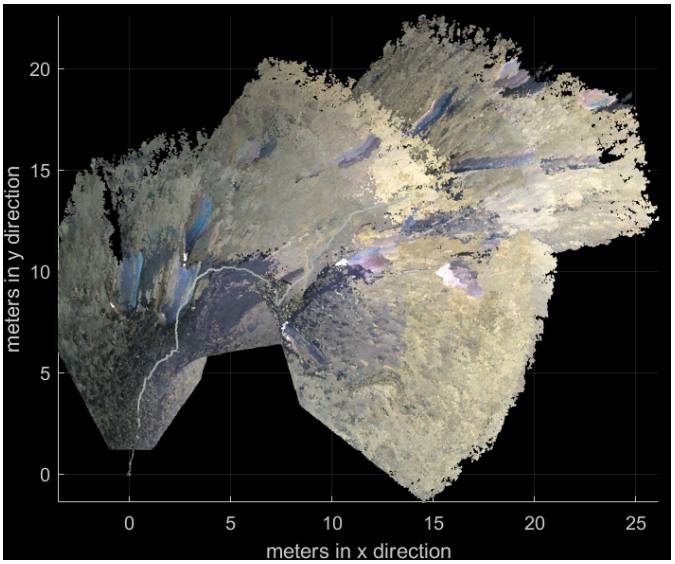


(d) Ground truth of part 4.

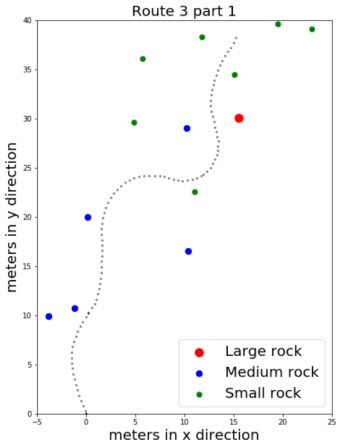


B Visual SLAM Using Point Clouds

B.0.1 Part 1



(e) Map and route of part 1 according to visual SLAM. The route can be seen as a gray line.



(f) Map and route of part 1 according to ground truth.

Figure 43: Map and route of part 1 according to visual SLAM (e) and ground truth (f).

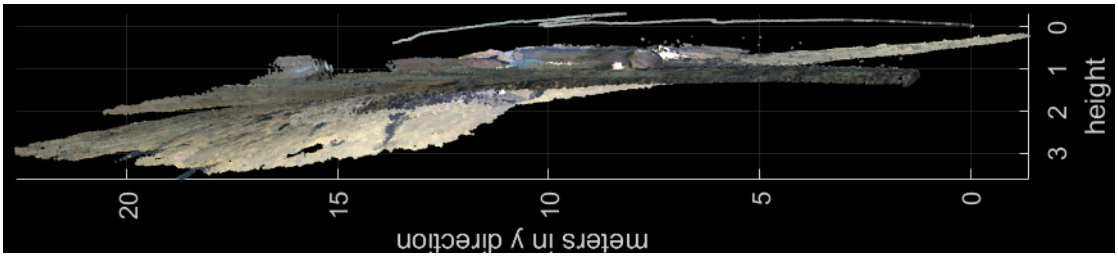


Figure 44: Left side view of part 1, the route can be seen floating above the scene.

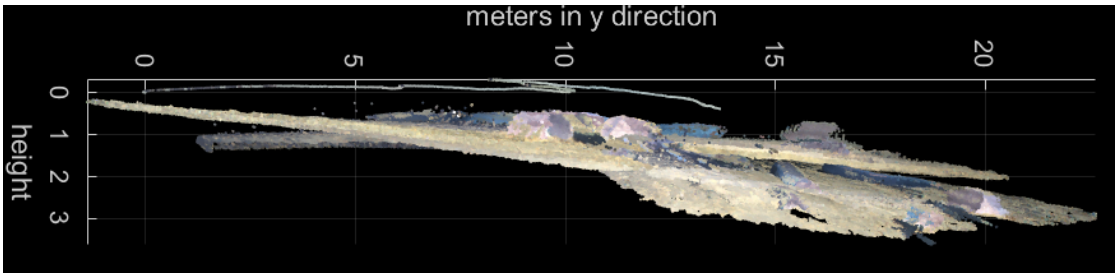


Figure 45: Right side view of part 1, the route can be seen floating above the scene.

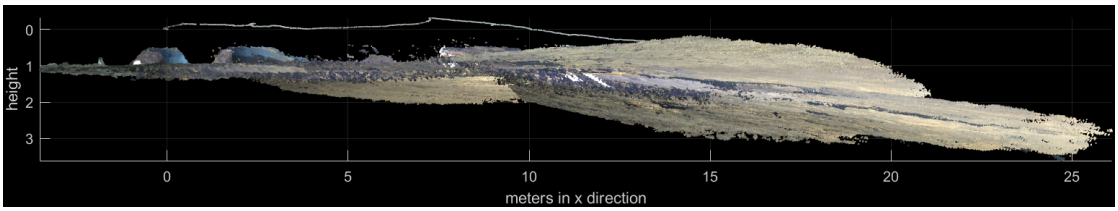
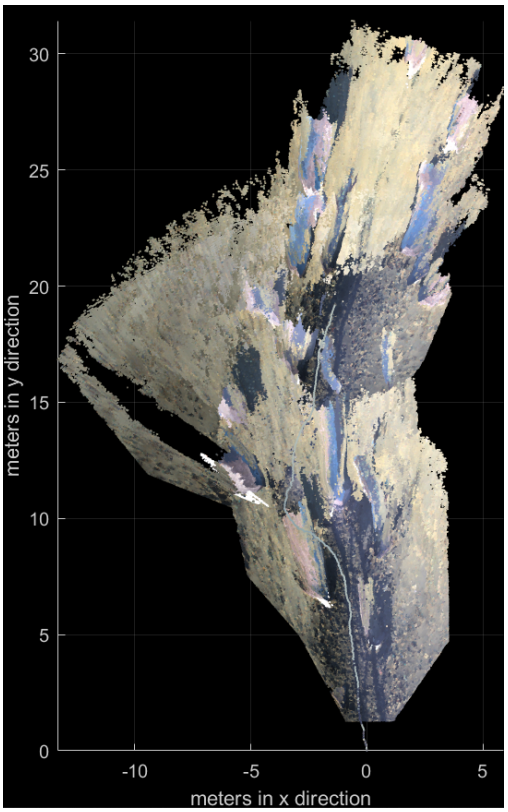
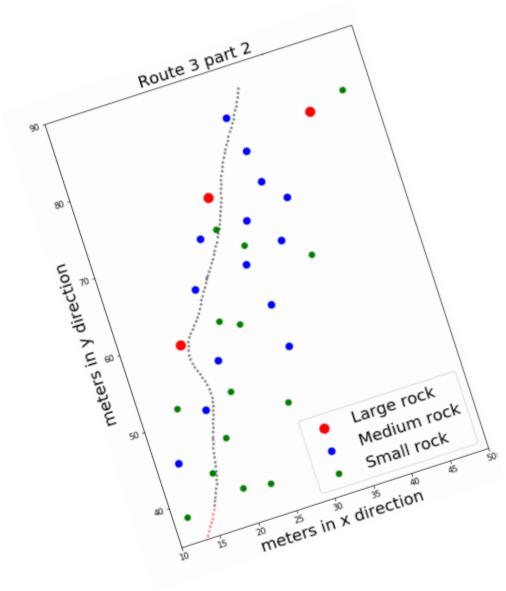


Figure 46: Frontal view of part 1, the route can be seen floating above the scene.

B.0.2 part 2



(a) Map and route of part 2 according to visual SLAM. The route can be seen as a gray line.



(b) Map and route of part 2 according to ground truth.

Figure 47: Map en route of part 2 according to visual SLAM (a) and ground truth (b).

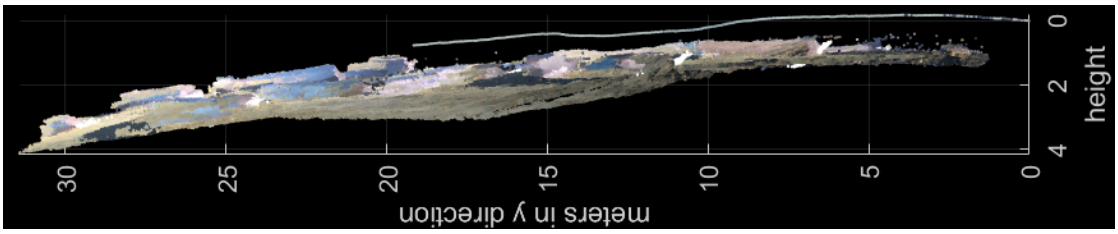


Figure 48: Left side view of part 2, the route can be seen floating above the scene.

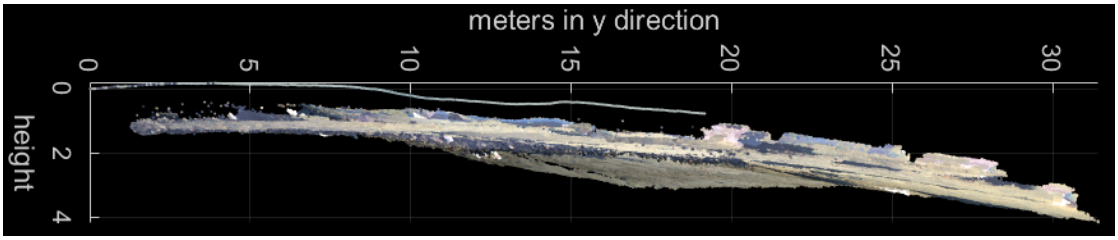


Figure 49: Right side view of part 2, the route can be seen floating above the scene.

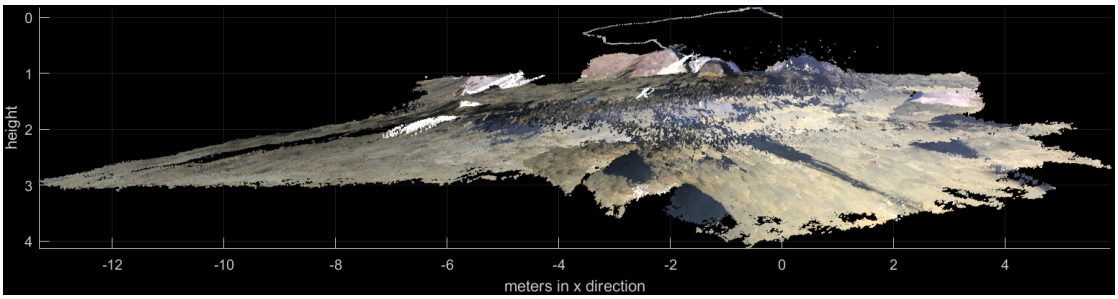
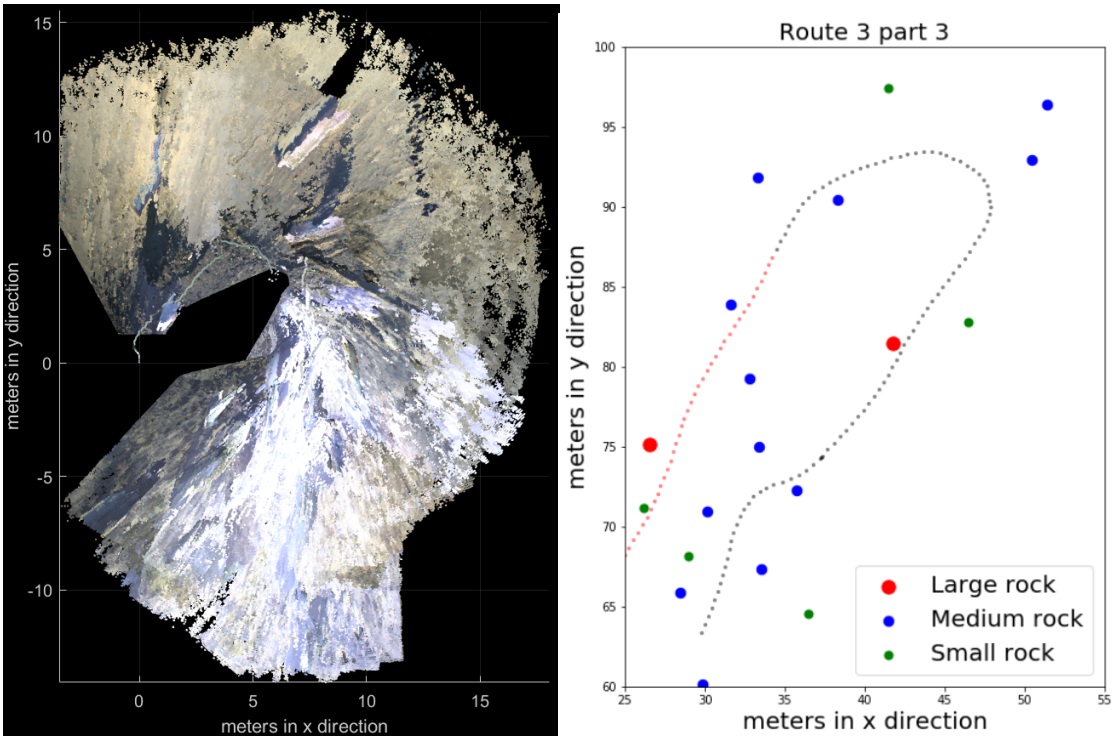


Figure 50: Frontal view of part 2, the route can be seen floating above the scene.

B.0.3 Part 3



(a) Map and route of part 3 according to visual SLAM. The route can be seen as a gray line.

(b) Map and route of part 3 according to ground truth.

Figure 51: Map and route of part 3 according to both visual SLAM and ground truth

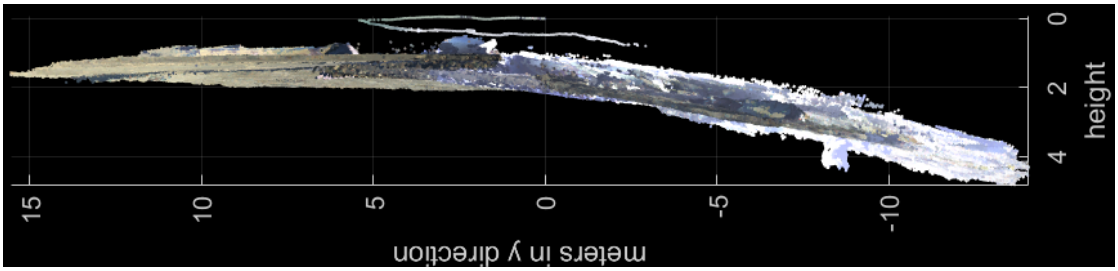


Figure 52: Left side view of part 3, the route can be seen floating above the scene.

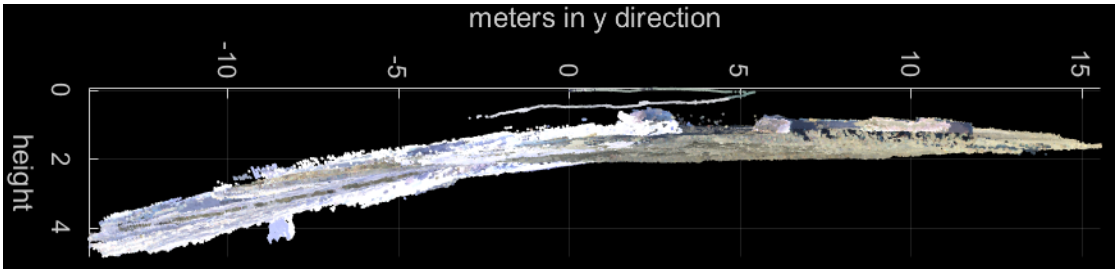


Figure 53: Right side view of part 3, the route can be seen floating above the scene.

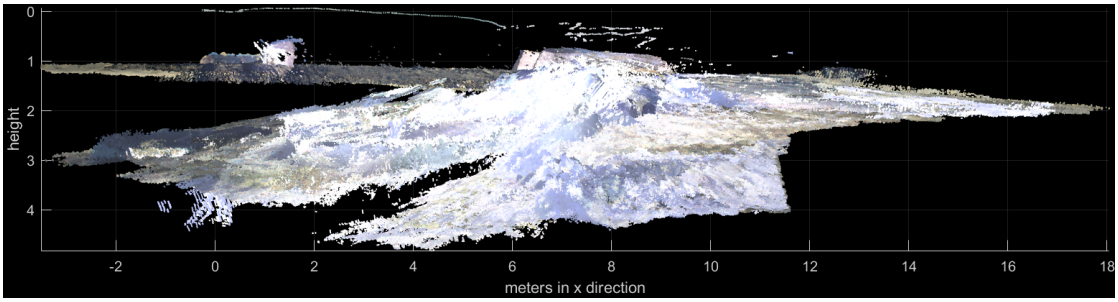
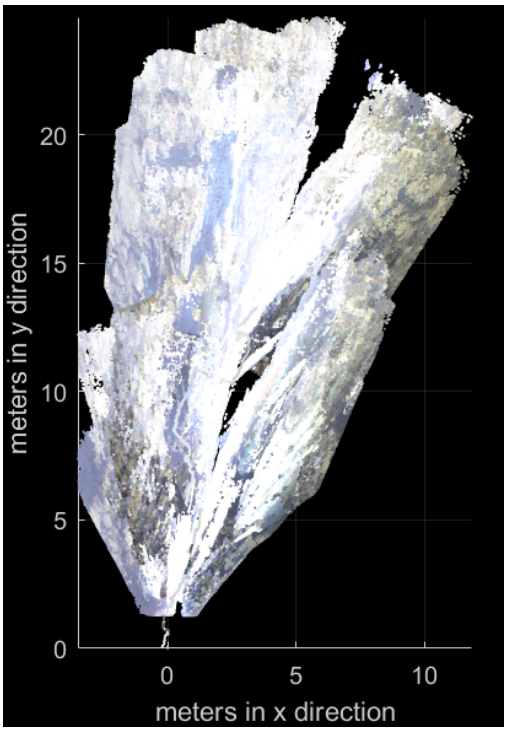
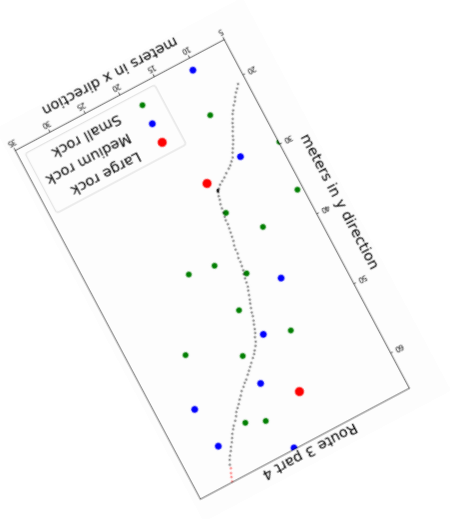


Figure 54: Frontal view of part 3, the route can be seen floating above the scene.

B.0.4 part 4



(a) Map and route of part 4 according to visual SLAM. The route can be seen as a gray line.



(b) Map and route of part 4 according to ground truth.

Figure 55: Map and route of part 4 according to both visual SLAM and ground truth

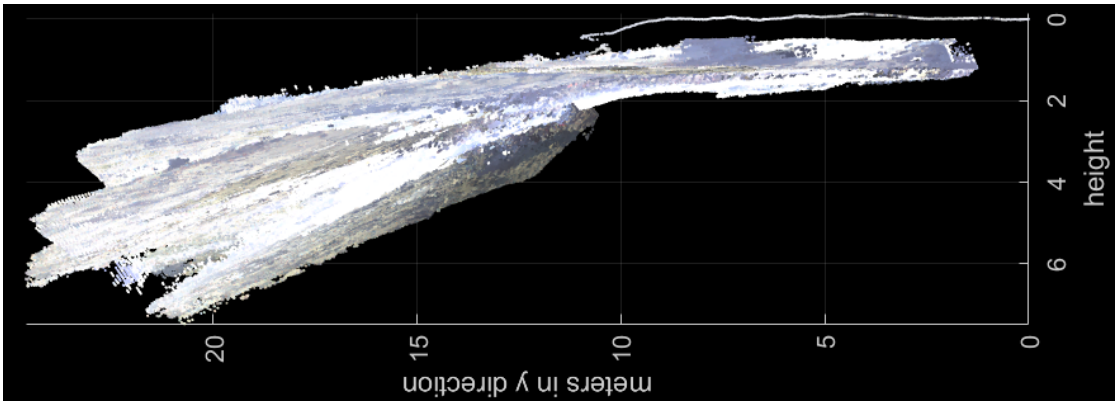


Figure 56: Left side view of part 4, the route can be seen floating above the scene.

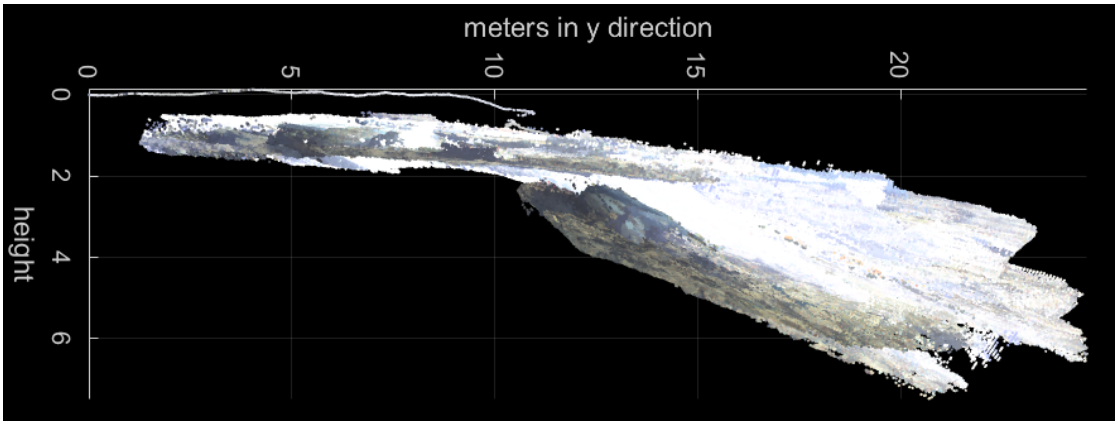


Figure 57: Right side view of part 4, the route can be seen floating above the scene.

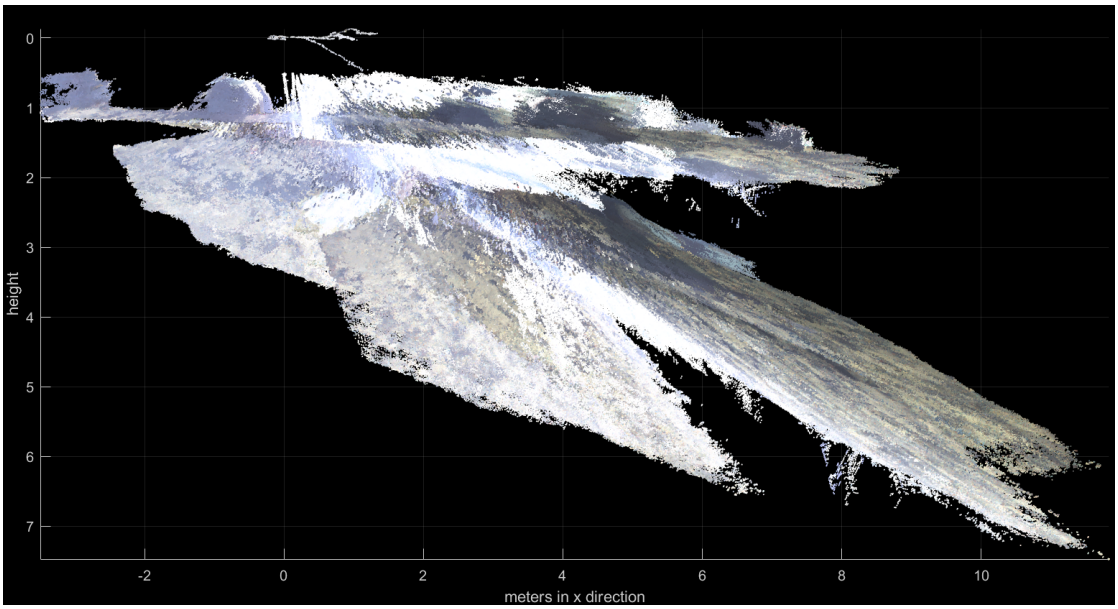
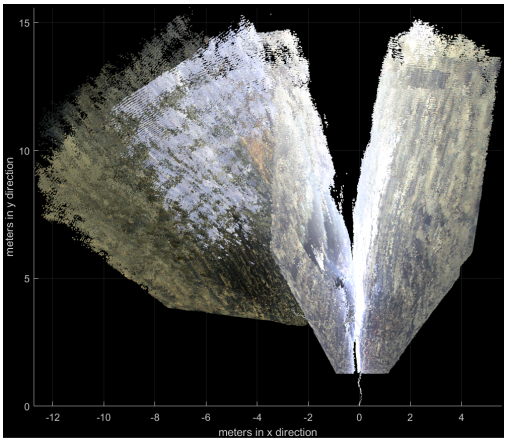
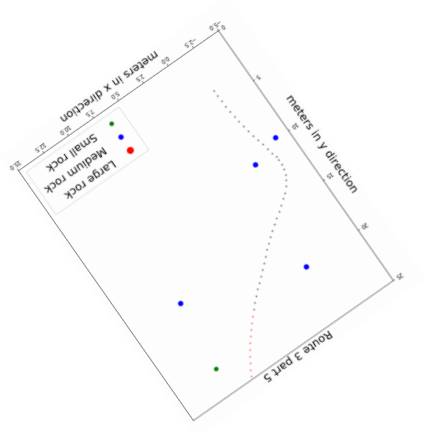


Figure 58: Frontal view of part 4, the route can be seen floating above the scene.

B.0.5 part 5



(a) Map and route of part 5 according to visual SLAM. The route can be seen as a gray line.



(b) Map and route of part 5 according to ground truth.

Figure 59: Map and route of part 5 according to both visual SLAM and ground truth

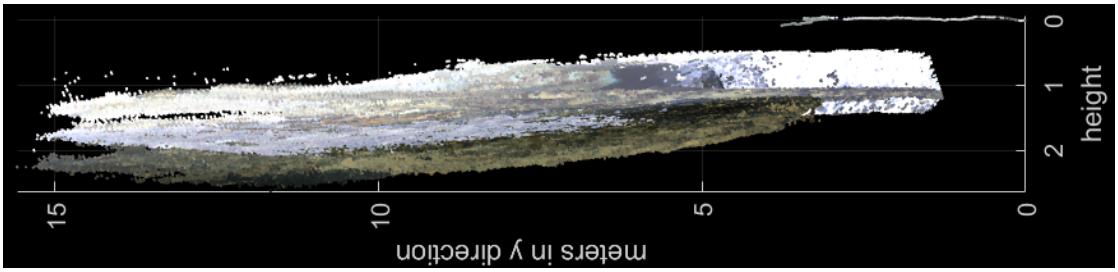


Figure 60: Left side view of part 5, the route can be seen floating above the scene.

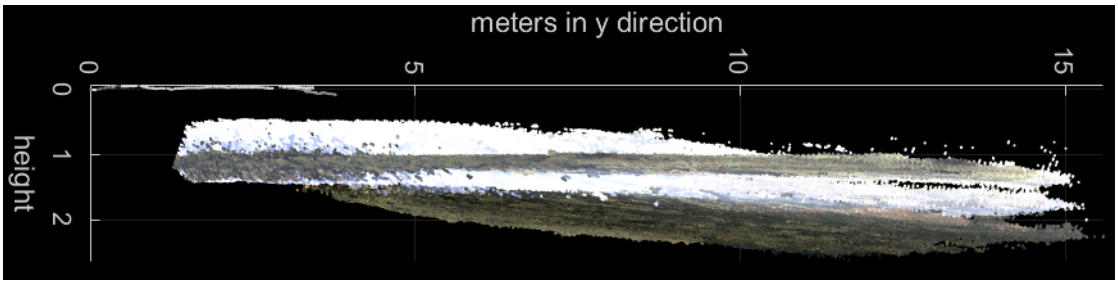


Figure 61: Right side view of part 5, the route can be seen floating above the scene.

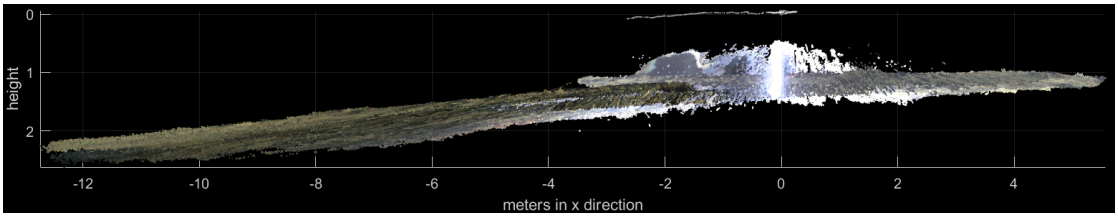
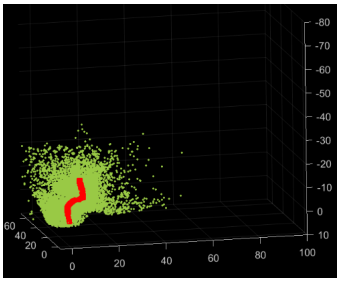
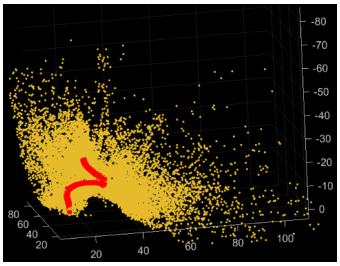


Figure 62: Frontal view of part 5, the route can be seen floating above the scene.

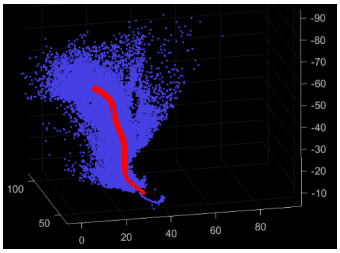
C Structure from Motion



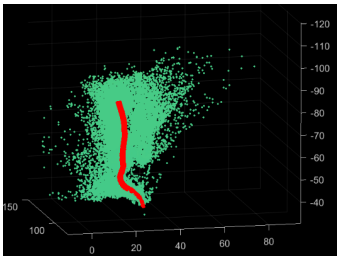
(a) Part 1.



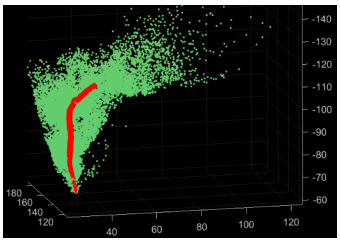
(b) Part 2.



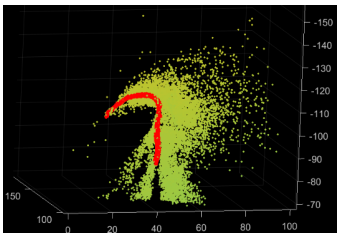
(c) Part 3.



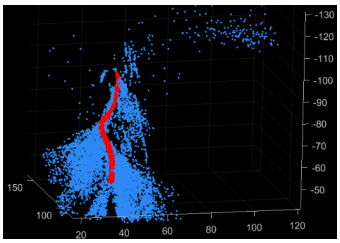
(d) Part 4.



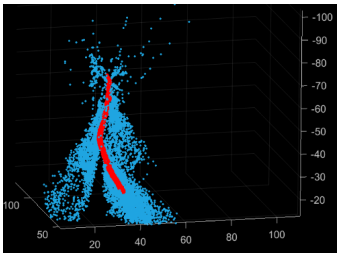
(e) Part 5.



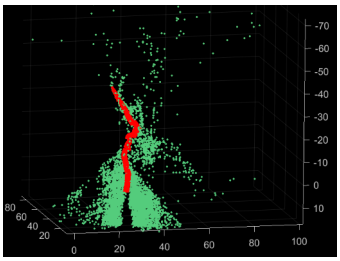
(f) Part 6.



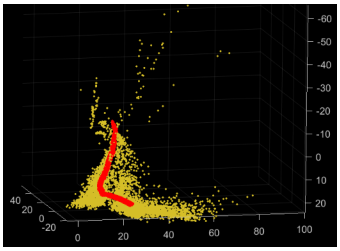
(g) Part 7.



(h) Part 8.



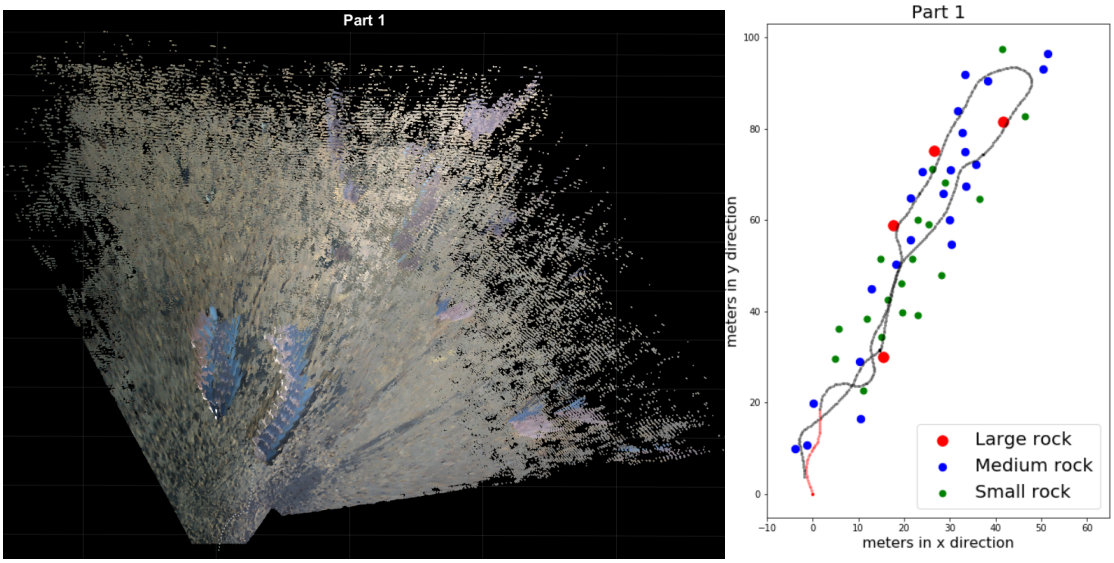
(i) Part 9.



(j) Part 10.

Figure 63: Structure from Motion map and route of each part.

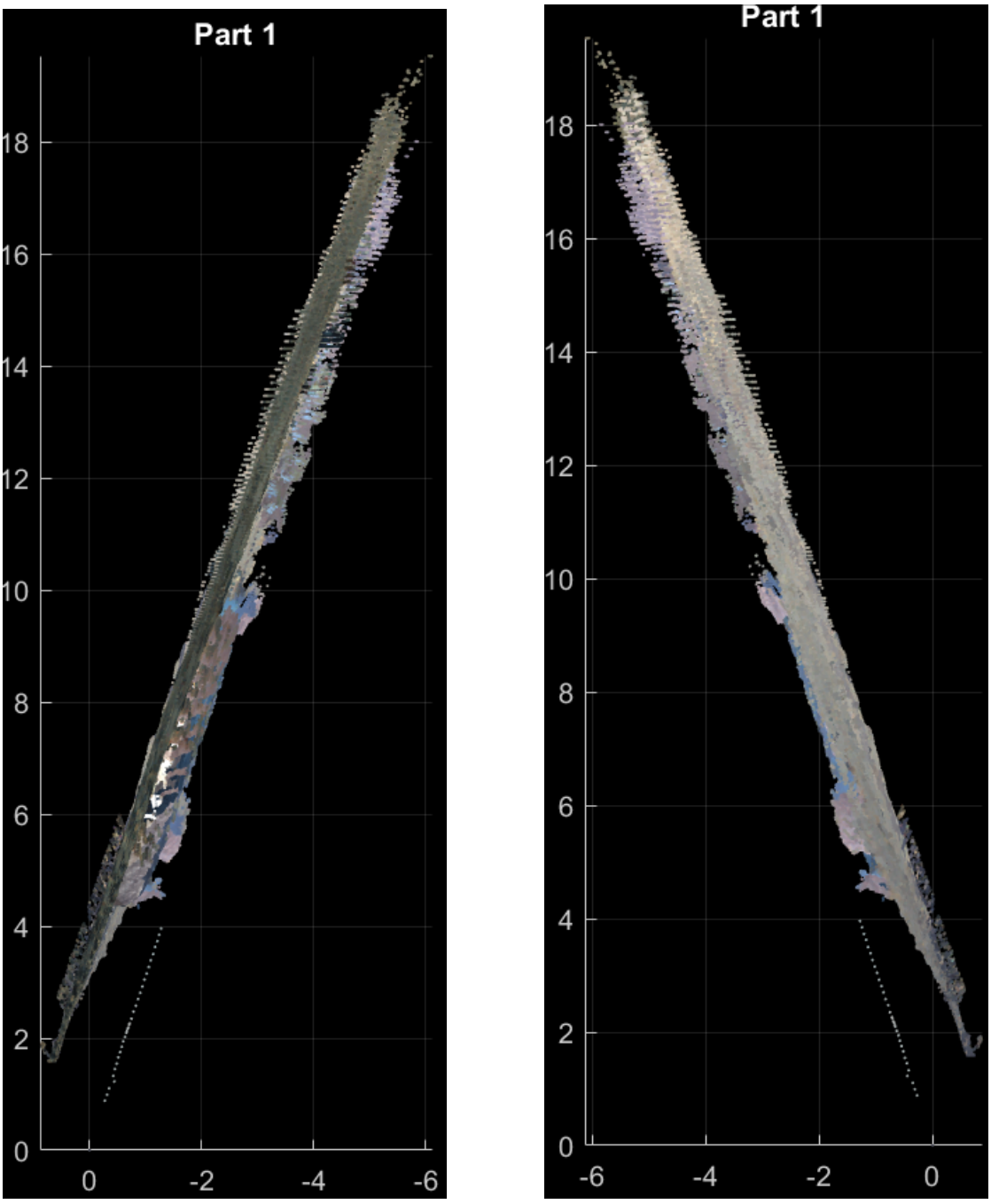
D Combined method



(a) Map and route of part 1 according to combined method. The route can be seen as a white dotted line.

(b) The ground truth route of part 1 is displayed in red.

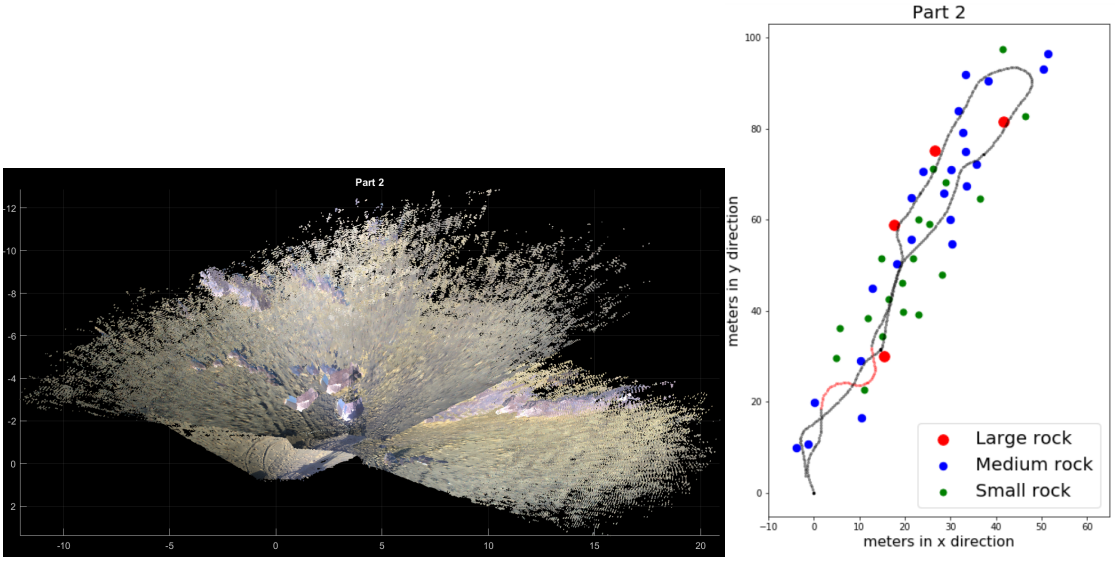
Figure 64: Map and route of part 1 according to the combined method (a) and ground truth (b).



(a) (b)

Figure 65: Left(a) and right(b) side view of part 1.

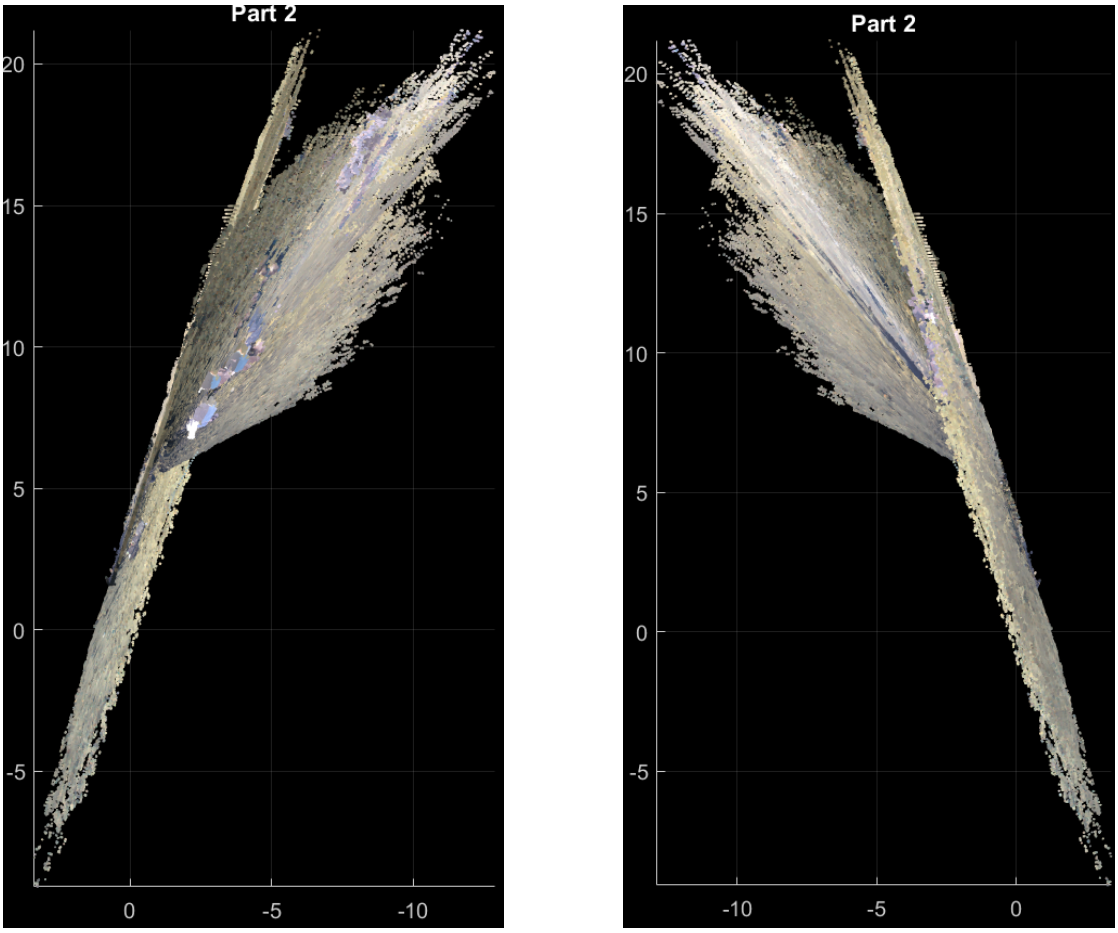
D.1 Part 2



(a) Map and route of part 2 according to combined method. The route can be seen as a white dotted line.

(b) The ground truth route of part 2 is displayed in red.

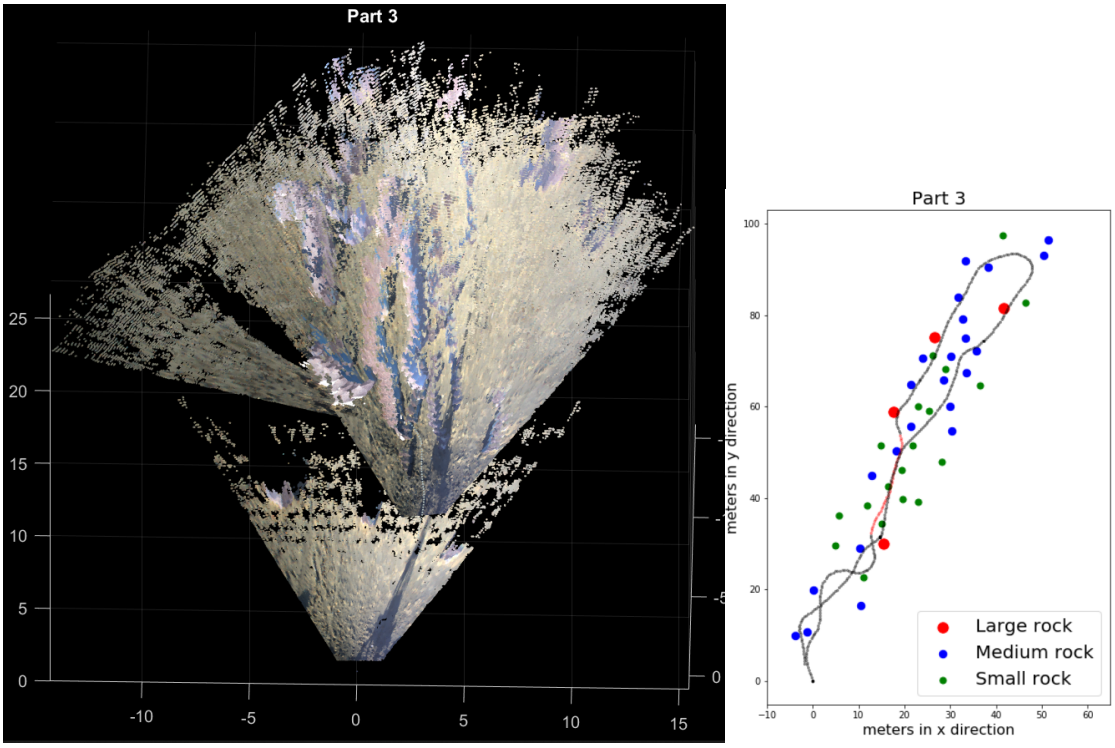
Figure 66: Map and route of part 2 according to the combined method (a) and ground truth (b).



(a) (b)

Figure 67: Left(a) and right(b) side view of part 2.

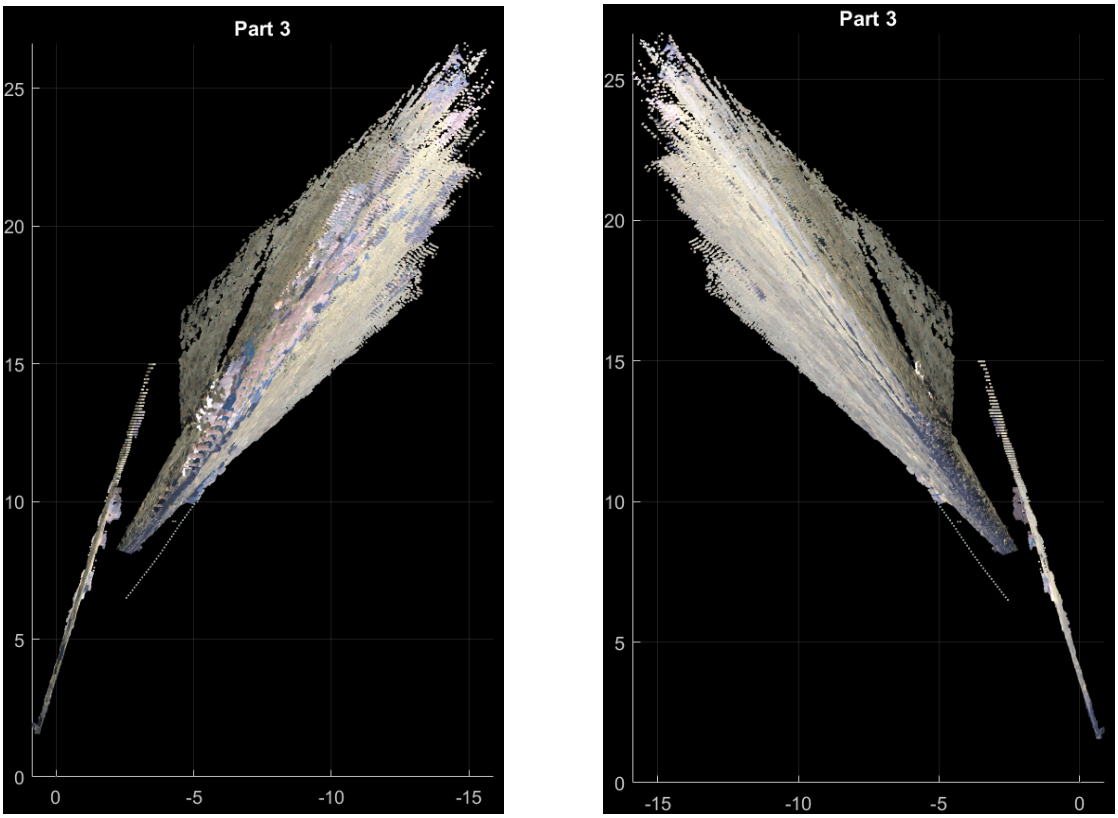
D.2 Part 3



(a) Map and route of part 3 according to combined method. The route can be seen as a white dotted line.

(b) The ground truth route of part 3 is displayed in red.

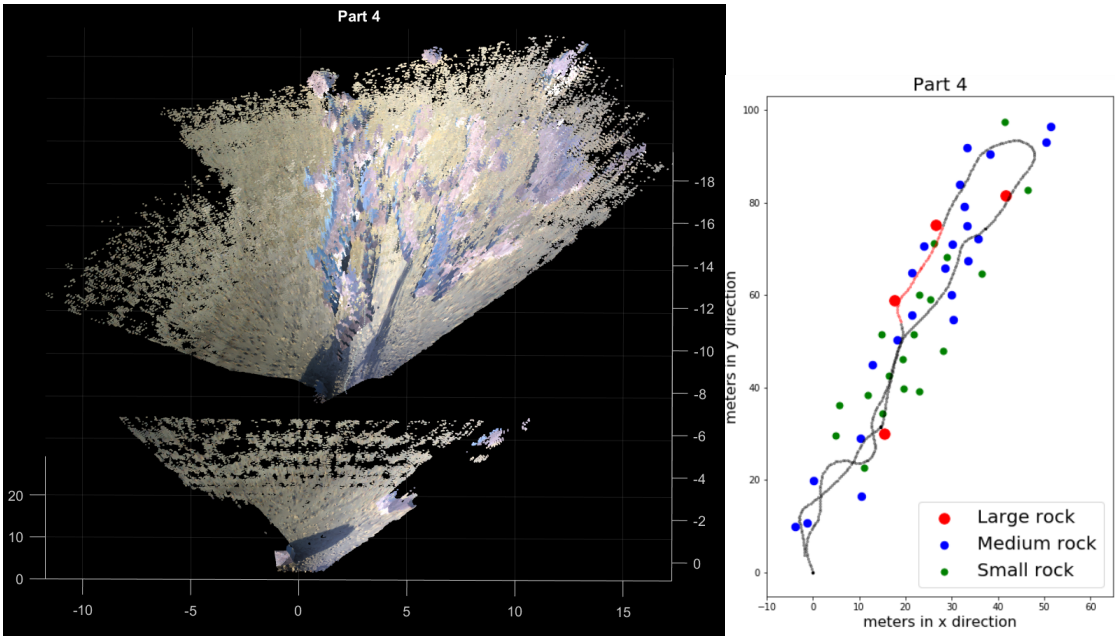
Figure 68: Map and route of part 3 according to the combined method (a) and ground truth (b).



(a) (b)

Figure 69: Left(a) and right(b) side view of part 3.

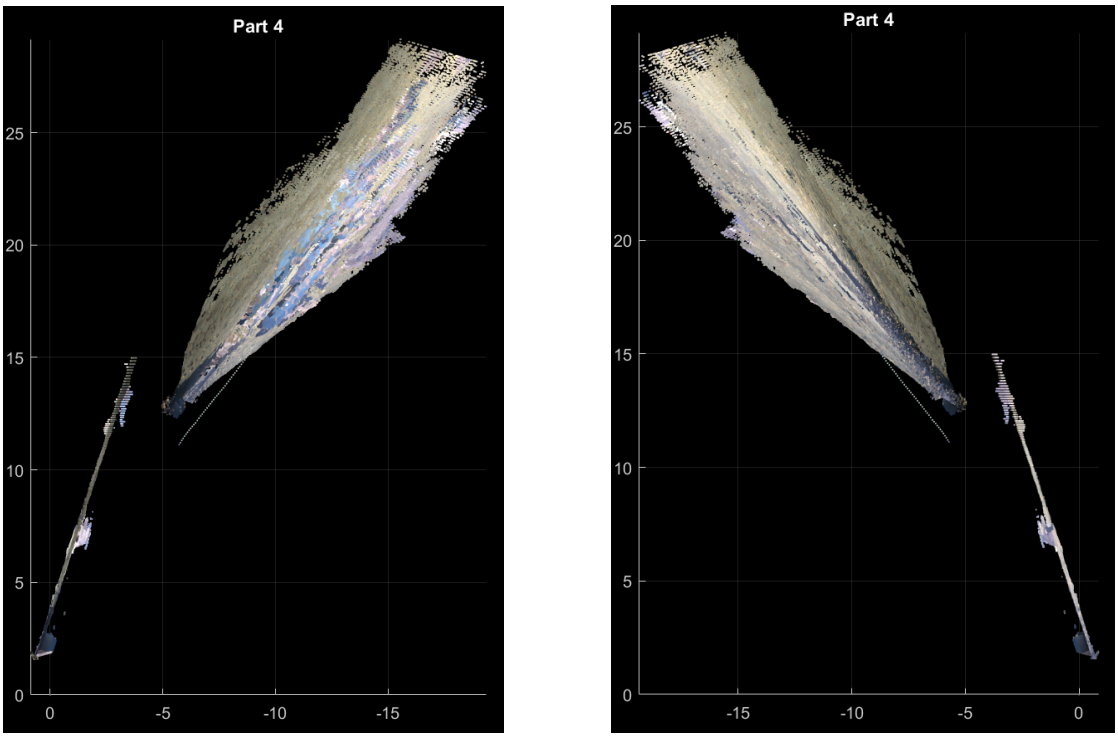
D.3 Part 4



(a) Map and route of part 4 according to combined method. The route can be seen as a white dotted line.

(b) The ground truth route of part 4 is displayed in red.

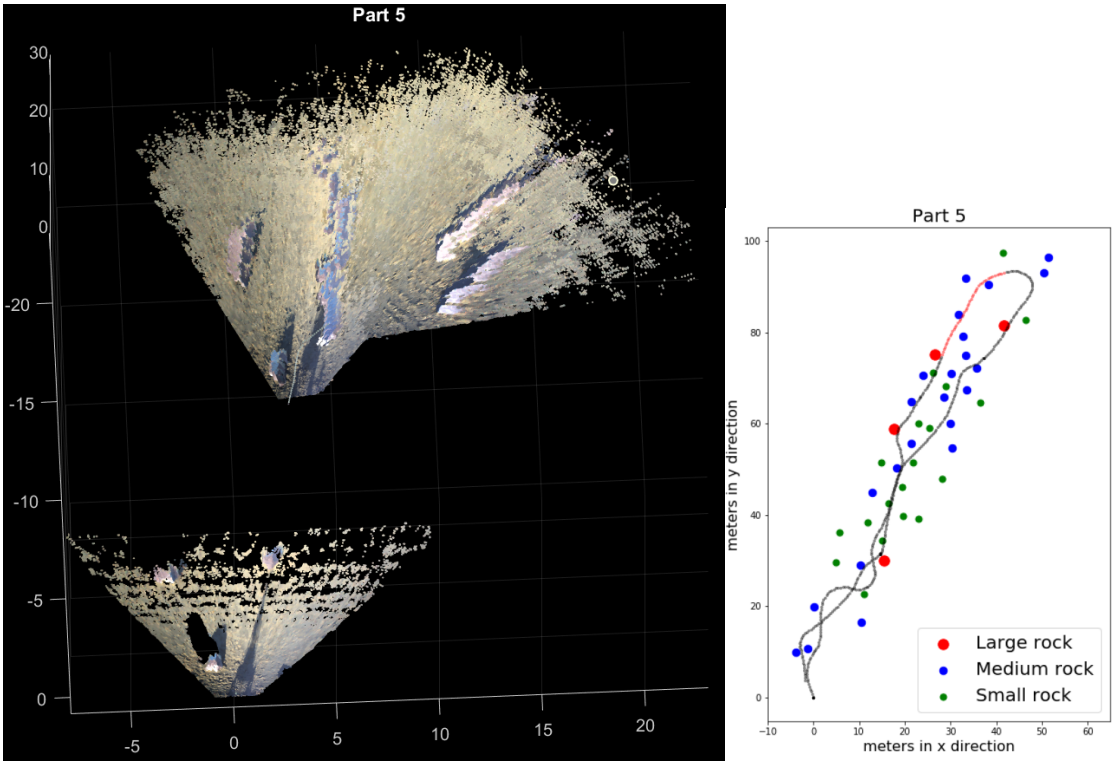
Figure 70: Map and route of part 4 according to the combined method (a) and ground truth (b).



(a) (b)

Figure 71: Left(a) and right(b) side view of part 4.

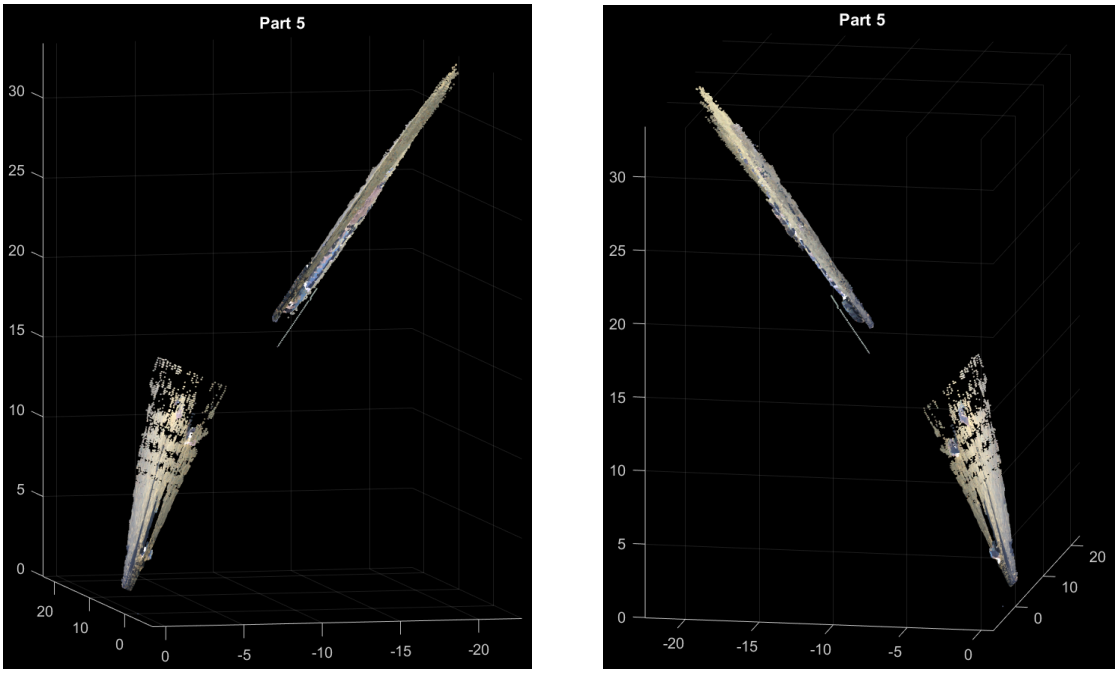
D.4 Part 5



(a) Map and route of part 5 according to combined method. The route can be seen as a white dotted line.

(b) The ground truth route of part 5 is displayed in red.

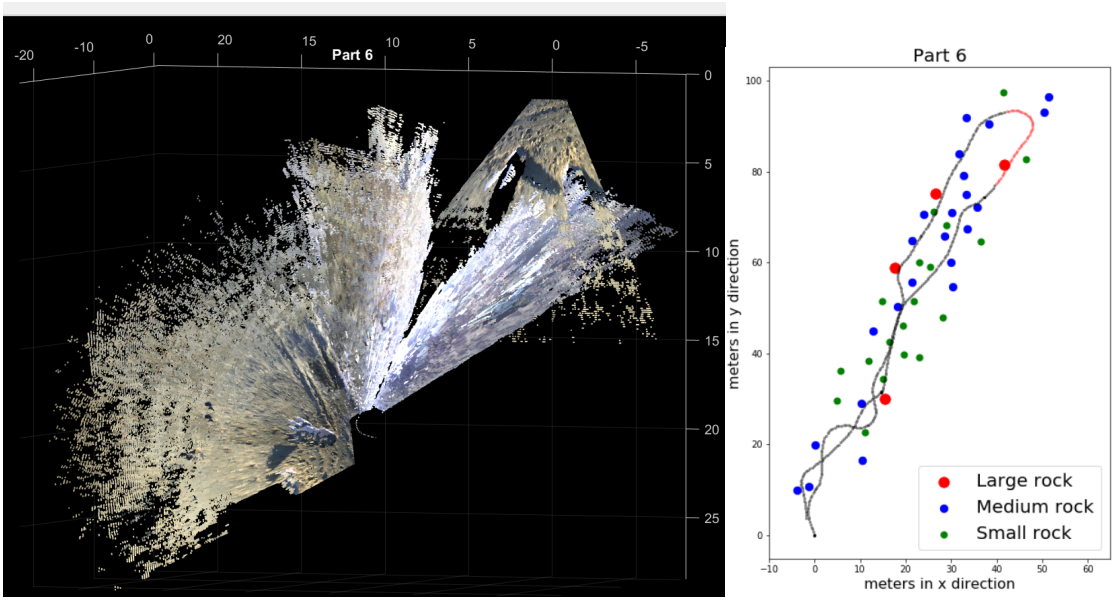
Figure 72: Map and route of part 5 according to the combined method (a) and ground truth (b).



(a) (b)

Figure 73: Left(a) and right(b) side view of part 5.

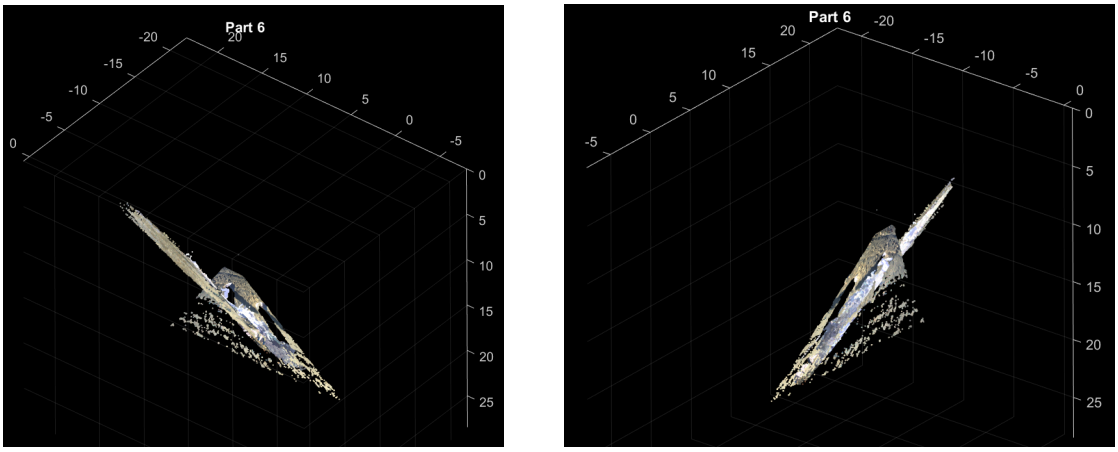
D.5 Part 6



(a) Map and route of part 6 according to combined method. The route can be seen as a white dotted line.

(b) The ground truth route of part 6 is displayed in red.

Figure 74: Map and route of part 6 according to the combined method (a) and ground truth (b).

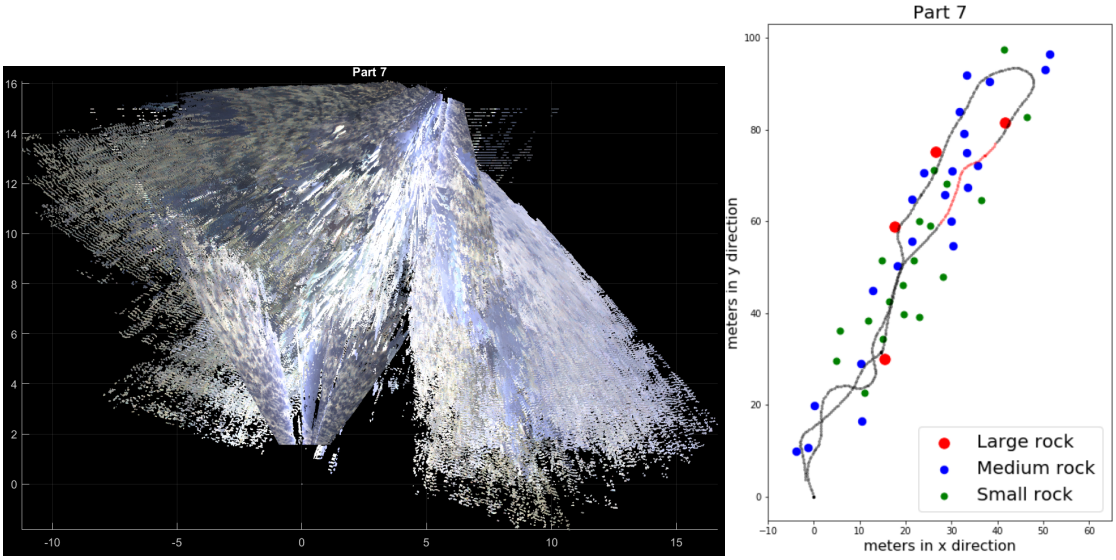


(a)

(b)

Figure 75: Left(a) and right(b) side view of part 6.

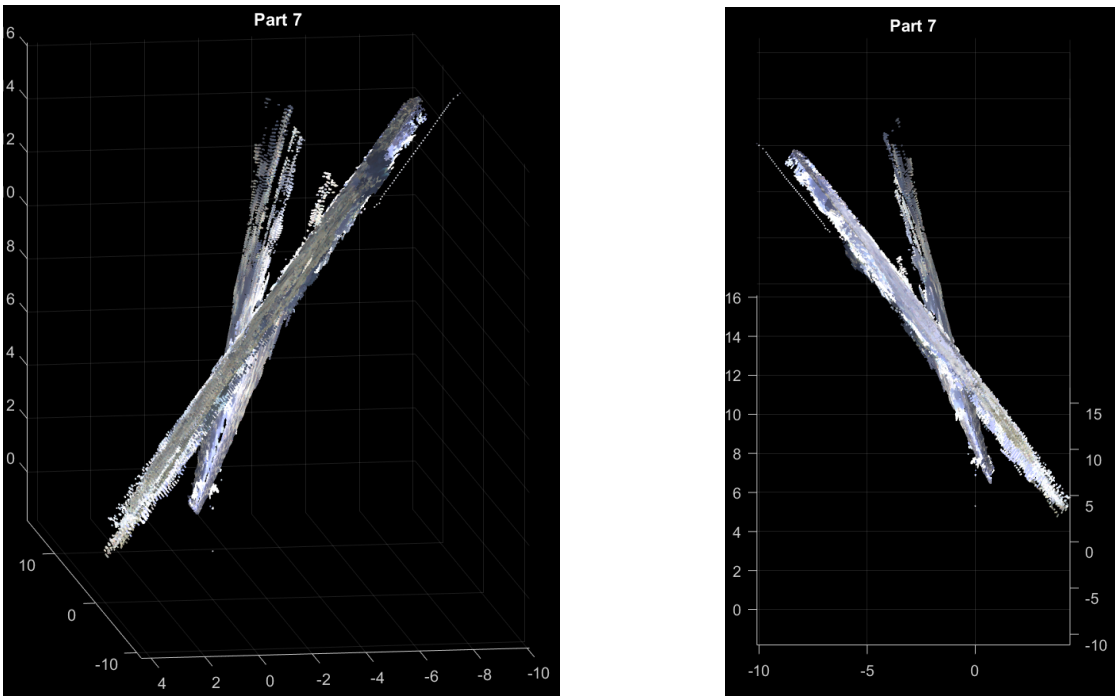
D.6 Part 7



(a) Map and route of part 7 according to combined method. The route can be seen as a white dotted line.

(b) The ground truth route of part 7 is displayed in red.

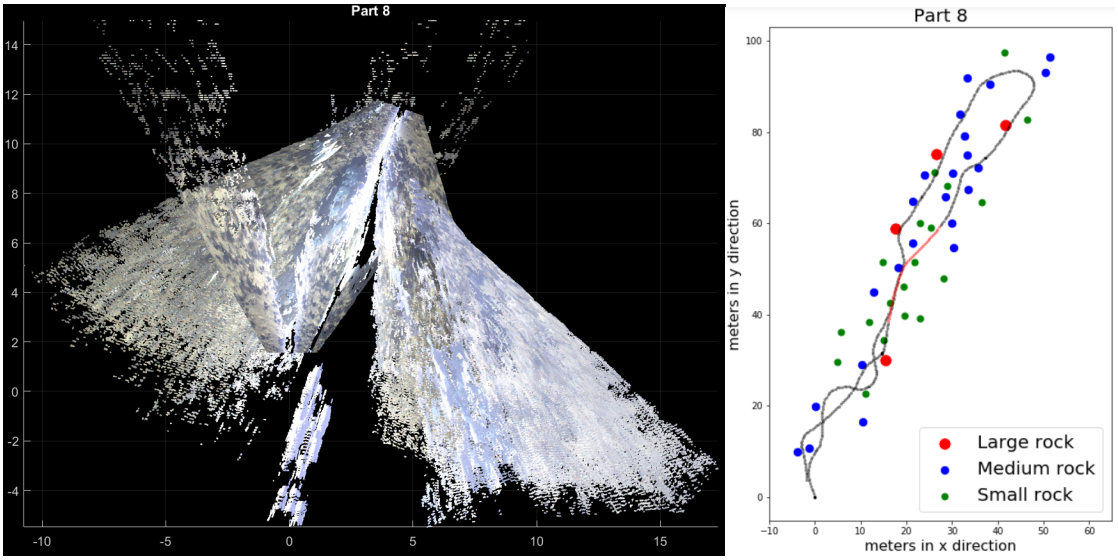
Figure 76: Map and route of part 7 according to the combined method (a) and ground truth (b).



(a) (b)

Figure 77: Left(a) and right(b) side view of part 7.

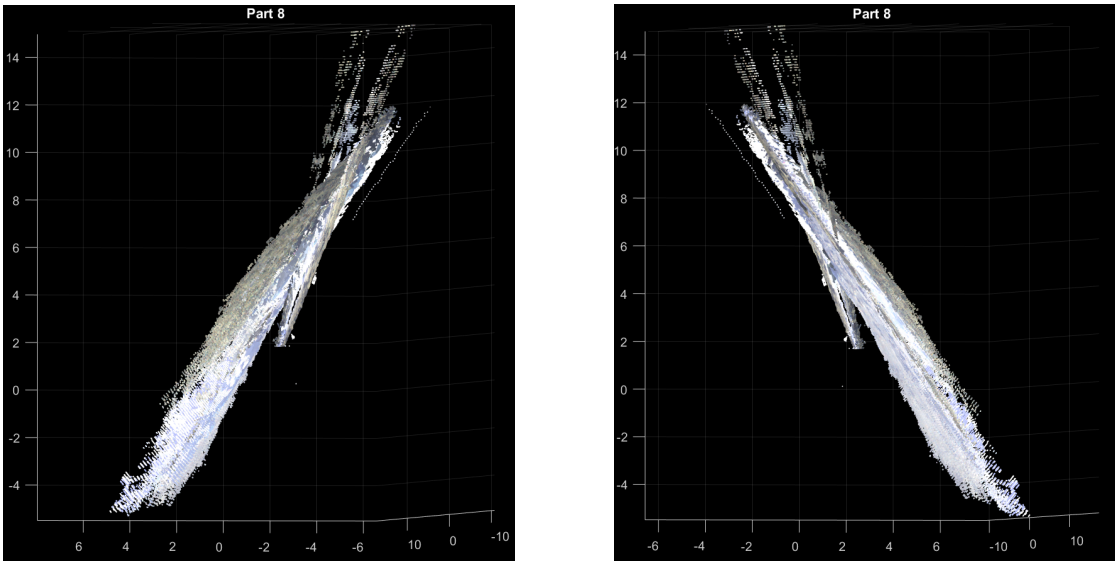
D.7 Part 8



(a) Map and route of part 8 according to combined method. The route can be seen as a white dotted line.

(b) The ground truth route of part 8 is displayed in red.

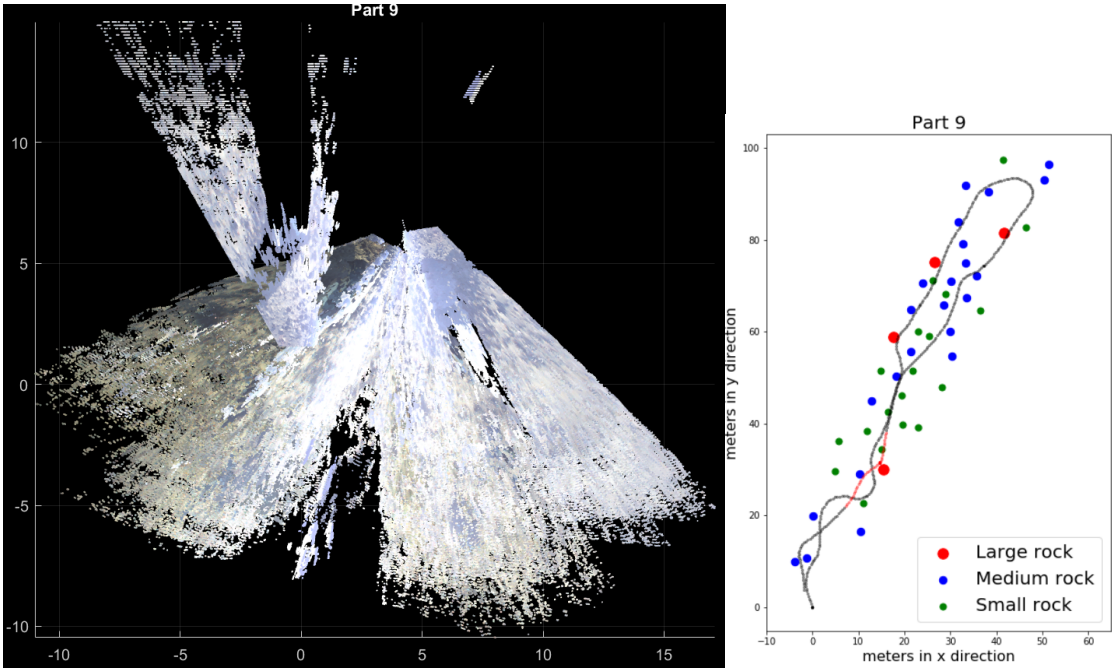
Figure 78: Map and route of part 8 according to the combined method (a) and ground truth (b).



(a) (b)

Figure 79: Left(a) and right(b) side view of part 8.

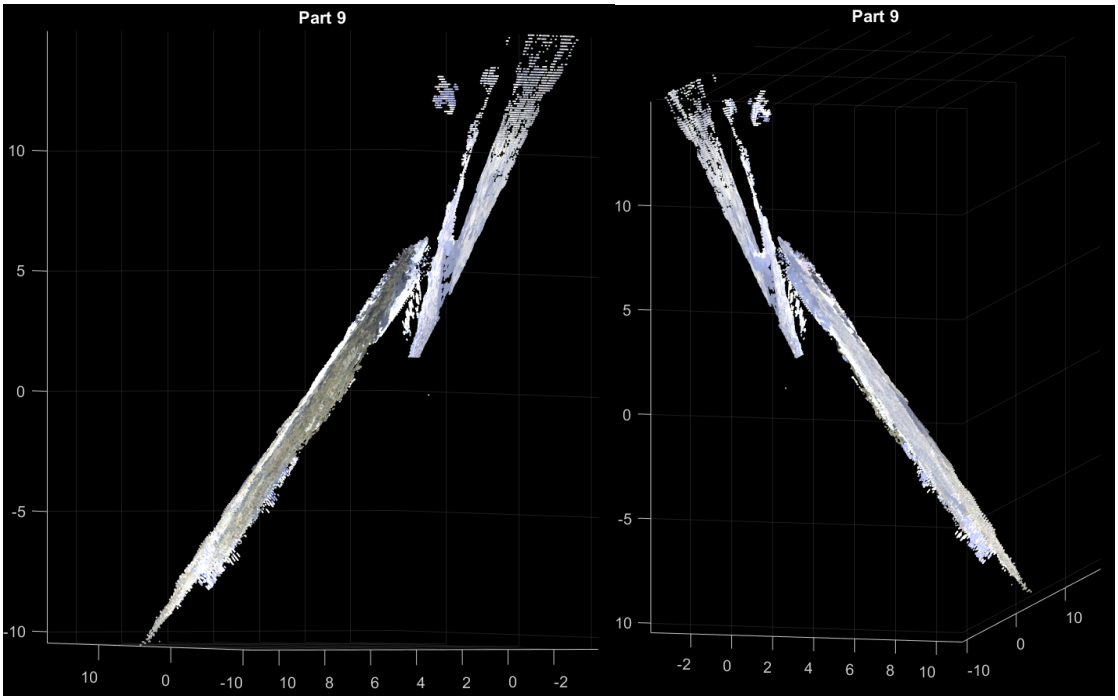
D.8 Part 9



(a) Map and route of part 9 according to combined method. The route can be seen as a white dotted line.

(b) The ground truth route of part 9 is displayed in red.

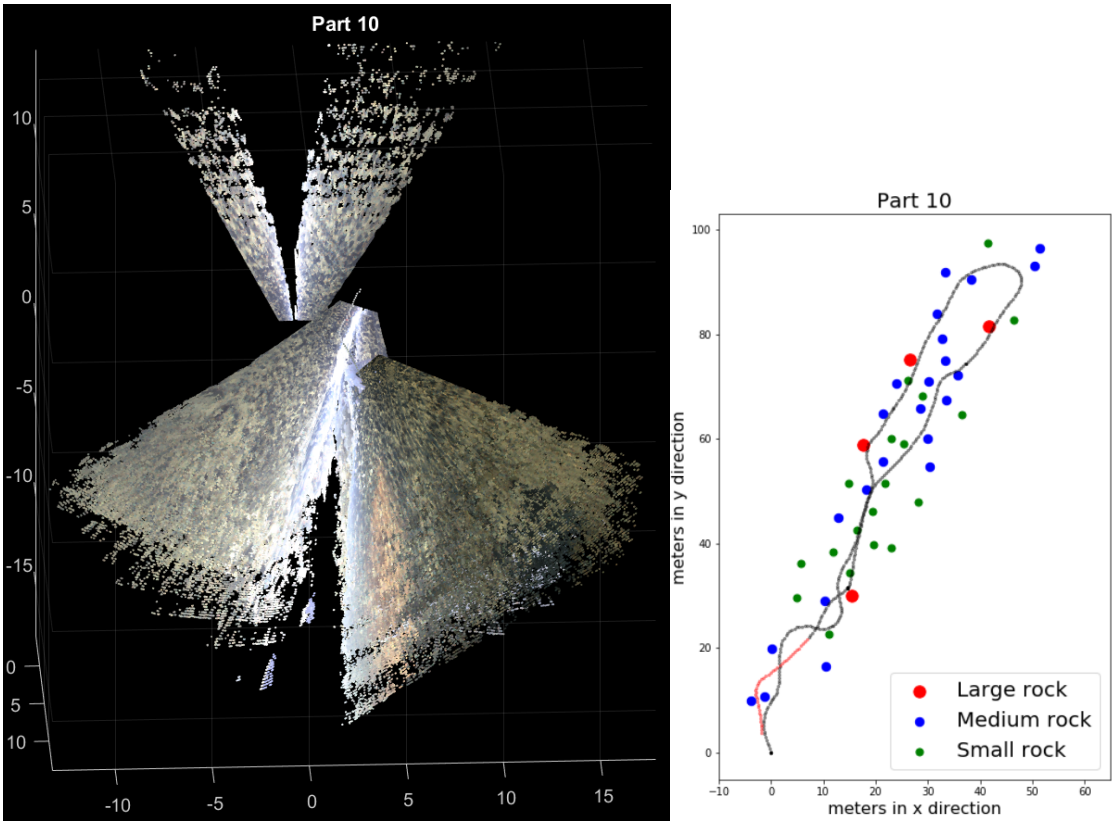
Figure 80: Map and route of part 9 according to the combined method (a) and ground truth (b).



(a) (b)

Figure 81: Left(a) and right(b) side view of part 9.

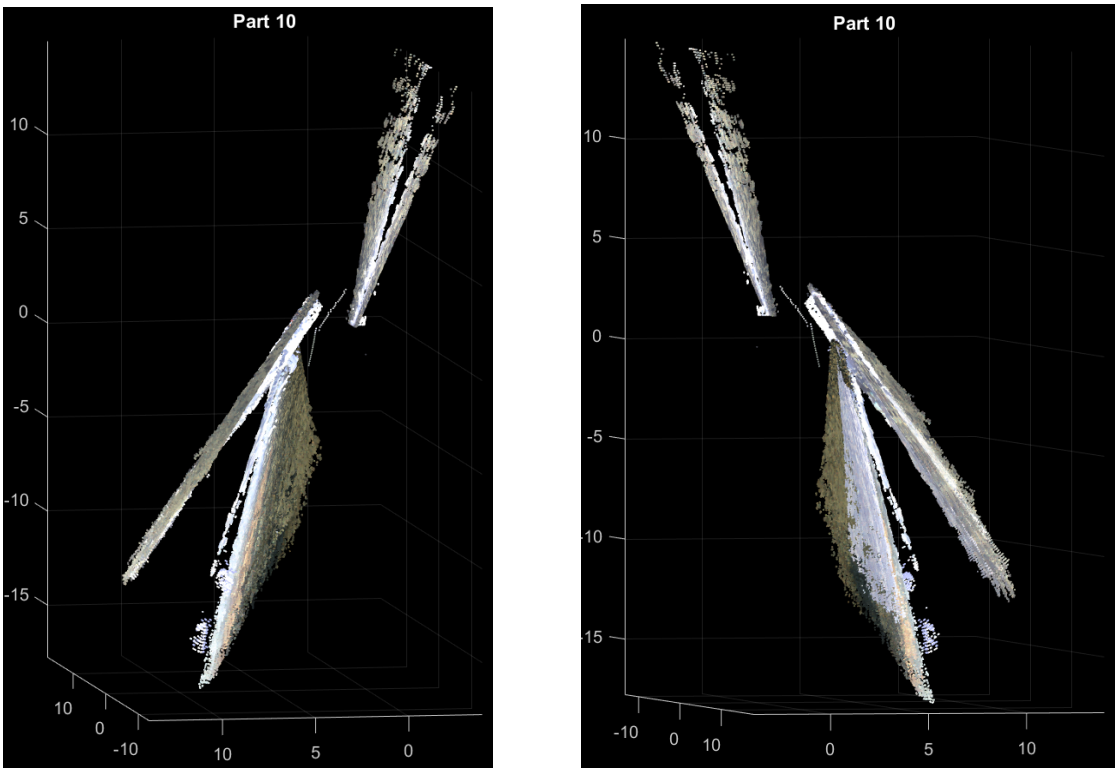
D.9 Part 10



(a) Map and route of part 10 according to combined method. The route can be seen as a white dotted line.

(b) The ground truth route of part 10 is displayed in red.

Figure 82: Map and route of part 10 according to the combined method (a) and ground truth (b).



(a) (b)

Figure 83: Left(a) and right(b) side view of part 10.