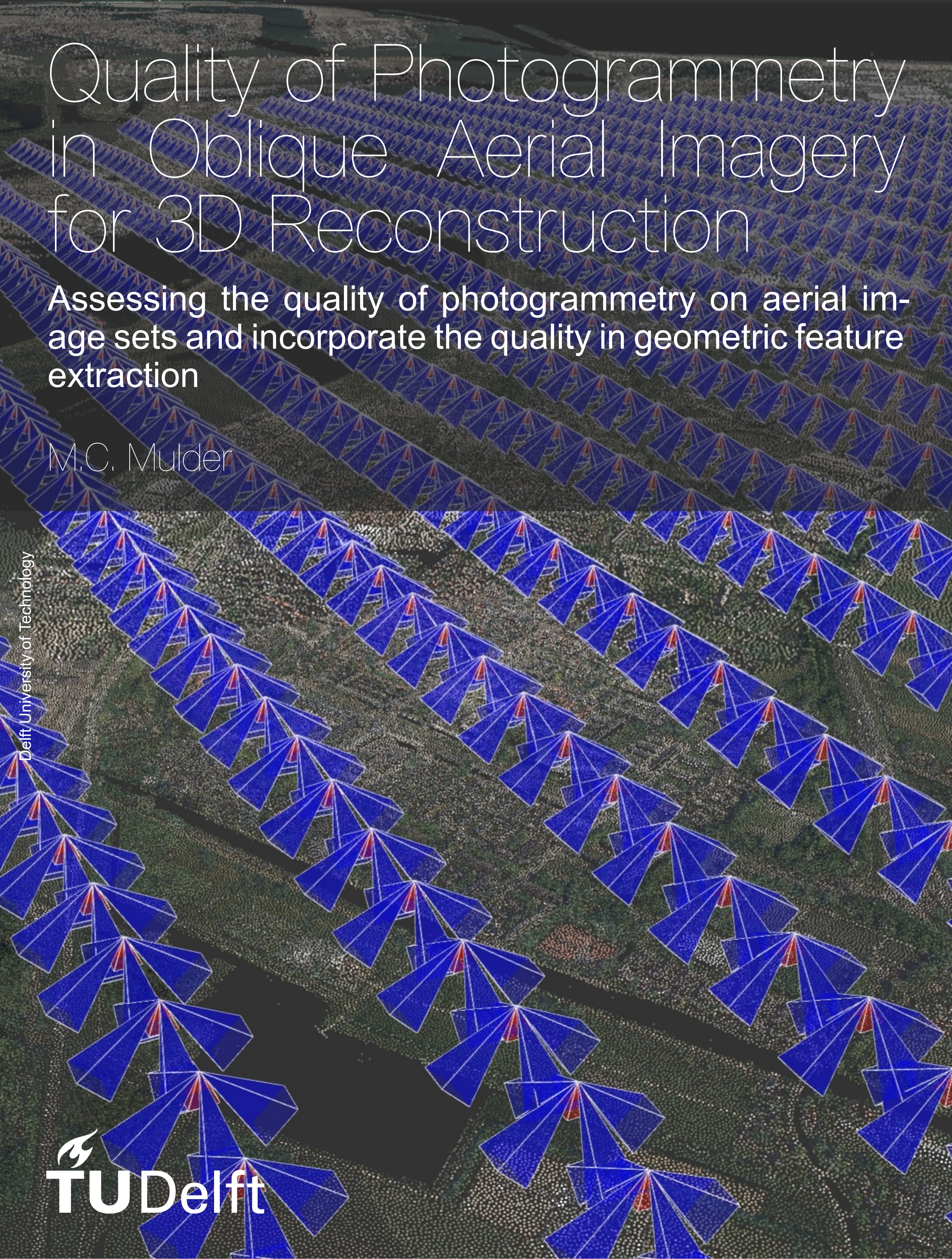


Quality of Photogrammetry in Oblique Aerial Imagery for 3D Reconstruction

Assessing the quality of photogrammetry on aerial image sets and incorporate the quality in geometric feature extraction

M.C. Mulder

Delft University of Technology

Quality of Photogrammetry in Oblique Aerial Imagery for 3D Reconstruction

Assessing the quality of photogrammetry on aerial image sets and incorporate the quality in geometric feature extraction

by

M.C. Mulder

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Friday June 2, 2023 at 14:00.

Student number:	4310551	
Project duration:	July 11, 2022 – June 2, 2023	
Thesis committee:	Dr. R.C. Lindenbergh	TU Delft Supervisor
	Ir. M.P. Kodde	Geodelta B.V.
	Dr. A.R. Amiri-Simkooei	TU Delft Supervisor

Cover: Oblique image cones in Omnibase by Geodelta
Style: TU Delft Report Style

TUDelft

Abstract

Photogrammetry is a well-established technique that has a significant impact on the use of images for mapping purposes. Employing feature extraction, matching, and the bundle adjustment, a large number of images can be automatically processed. Ingenieursbureau Geodelta developed an application for measurements in processed (oblique) aerial images. However, the quality of these measurements and the added value of oblique images on the adjustment's quality were unclear. This will be assessed by means of the theoretical standard deviation resulting from the bundle adjustment. Upon estimating this quality, an adapted RANSAC method is proposed that incorporates this quality as weights within its algorithm to extract geometric features. The objective is to evaluate whether this enhances the RANSAC results and could be applied to 3D reconstructions.

The results indicate three key factors that influence the theoretical standard deviation: high tie point availability, larger observation angles, and image viewing direction. With this, the theoretical standard deviation for tie points in both Nadir and Oblique image sets separately approximates 3 centimeters in the horizontal direction and about 10 in the height direction. Combining the two sets enhances the results by nearly a factor of three in all directions because the Nadir images connect the Oblique images, combining the strong characteristics of both sets. This demonstrates the value of both Nadir and Oblique imagery. For image exteriors, the improvement is even more pronounced, yielding an improvement factor of 4 or 5. However, propagating this quality metric through a dense matching algorithm in an adapted, weighted RANSAC algorithm does not show significant improvements in the number of planes found or the percentage of points classified as inliers of those planes. Furthermore, the RANSAC method does not converge to a better result in fewer iterations using the proposed method.

Contents

1 Introduction

In this chapter, an introduction in the topic is given and the main goals for the research performed in this thesis are described, along with the research questions that will be answered in the thesis.

1.1. Surveying and photogrammetry in the Netherlands

In the Netherlands, all municipalities are legally required to maintain a comprehensive map (BGT, section 2.1) that contains information on the location and nature of objects such as buildings, roads, water bodies, and vegetation within their territory. Various techniques, including physical in-situ measurements, laser scanning, and image-based measurements, are used to update this map. However, physical measurements are time-intensive and expensive, and thus, researchers have explored alternative remote measurement techniques.

One such technique is photogrammetry, which involves using images to identify and locate specific objects or points across multiple images to calculate their coordinates. Many of the processes involved in this technique are now automated. Currently, the BGT is maintained in 2D, primarily using measurements from nadir images. However, this approach has limitations. For instance, objects partially obscured by other objects may not be visible, rendering them unidentifiable. Additionally, photogrammetry using only nadir images is less accurate in the vertical dimension due to the small intersection angles between the view lines of images.

Therefore, it may be beneficial to use oblique images captured from an angle. This approach enables the measurement of object sides, resulting in more precise measurements since objects are visible from multiple sides. It also allows for the creation of a 3D model of the built environment, enabling the maintenance of the BGT in 3D. However, the accuracy of this approach remains a crucial question. Geodelta has developed the Omnibase application, which supports measuring in oblique images. Nevertheless, its current precision is based only on the final part of the calculation and does not account for the accuracy of the camera's location and orientation.

1.2. Research Question

In this section, a goal is drafted and the research questions are formulated.

1.2.1. Objective

With the situation described above, an objective can be formulated. As the quality of measurements in oblique images is stated as an unknown problem, and the 3D reconstruction as a possible result of using the oblique images, the goal of this thesis will be to assess the theoretical quality of the tie points and orientations of the images, and use this as an improvement base in a 3D reconstruction algorithm.

Quality of photogrammetry in oblique imagery

The first part of the objective of this thesis will be to find the quality of the results of the bundle adjustment, and find some characteristics of the image set that define mostly the theoretical quality characteristics of oblique images. Particularly the difference between the image sets is investigated, to see what added value oblique images have. In this thesis, if there is referred to quality, mostly the theoretical standard deviations resulting from the adjustment of the image set is meant.

Geometric feature extraction
The second part of the objective of this thesis is to generate a 3D model of different buildings based on oblique image observations. For visualisation purposes, it can be very practical to generate a 3D model based on oblique images. The goal here will be to see if it is possible and useful to take the quality into account and improve a geometric feature extraction algorithm. An important objective is here to propagate and use the obtained quality information in this 3D feature extraction. The quality of the points can act as weights, and the goal is to let the algorithm be more likely to use better points.

1.2.2. Research Questions

With the objectives described above, the main question for this thesis is going to be:
What is the quality of oblique aerial imagery compared to nadir imagery for photogrammetry purposes, and how can this quality be incorporated in a 3D reconstruction of buildings?

1. How can the theoretical quality of a point coordinate in oblique images be found?
2. What is the difference in quality between the oblique image observations and nadir image observations?
3. What are the characteristics of the observation parameters and surroundings that have influence on the quality of the results?
4. How can the quality of points be weighted in a plane fitting algorithm?
5. Does the RANSAC method improve performance by incorporating the quality as a weighting metric?

1.3. Thesis outline

Chapter 2 of this thesis presents the literature review, where the Basisregistratie Grootschalige Topographie (BGT) and the photogrammetric method are explained in detail. Additionally, the geometric feature extraction algorithm RANSAC is discussed in this chapter. In chapter 3, the data used in this study is specified, and the area on which this data is applicable is described. The methodology developed to answer the research questions is presented in chapter 4, with a step-by-step explanation of the procedures followed. The results obtained from the application of this methodology are discussed in chapter 5. Chapter 6 provides a discussion of the results, along with recommendations for future research. Finally, the conclusion is presented in chapter 7.

2

Photogrammetry and 3D reconstruction for the built environment

In this chapter, the literature will be described that is used to perform the research that is done in this thesis. In section 2.1, the general map of the Netherlands and its purpose is discussed. Then in section 2.2, the principles of photogrammetry are explained and in section 2.3 relevant photogrammetry software is described. In section 2.4, methods to extract and reconstruct geometric features for 3D models are discussed.

2.1. Basisregistratie Grootschalige Topografie

In the Netherlands, the government has declared that the large-scale topography of the entire country must be mapped and be updated regularly. This is gathered in one uniform map, the Basisregistratie Grootschalige Topografie (BGT), which translates roughly to Key Register Large-scale Topography. It has been developed by the government since 2008 in cooperation with all stakeholders, and has come to completion in 2017. In the BGT, the following classes are represented:

- Buildings
- Roads
- Water
- Vegetation
- Railways
- Civil engineering structures

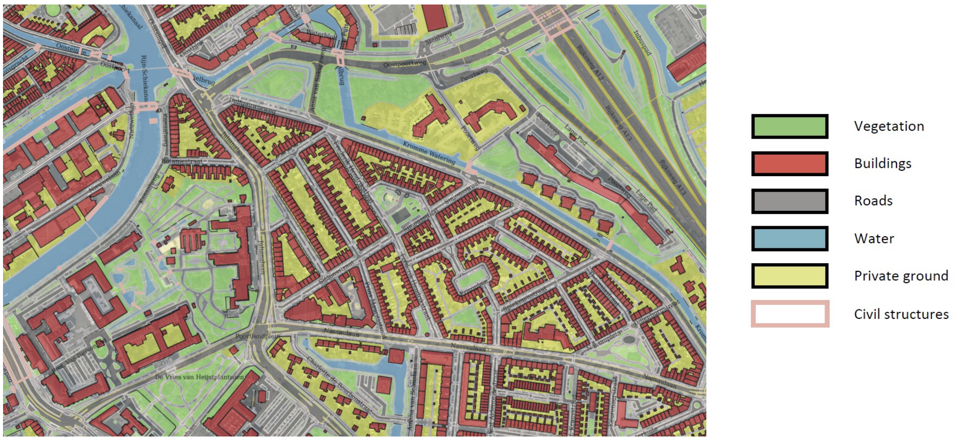


Figure 2.1: BGT with classes as described above [5]

The scale of this map is 1:500 - 1:5000, which results in a precision of around 20cm.

2.1.1. Utilities of the BGT

The BGT is the formal topographic map of the Dutch government, and can therefore be used as underlying layer of projects, to plan navigation routes for cycling, driving, or sailing.

For all governing bodies (municipalities, provinces, water authorities, the ministries and others) in the Netherlands, it is mandatory to use the BGT as base layer for all projects. These projects include vegetation maintenance, determination of demographics, presenting of urban renewal plans, or for instance planning of evacuation routes.
The BGT is available for free to anyone who wants to use it.

2.1.2. Maintenance and requirements of the BGT

Every governing body (municipalities, provinces, water authorities, different ministries and ProRail) in the Netherlands is required by law to update the BGT of their own territories regularly. They are required to submit this information to the Kadaster, who collects this data from all source data holders and processes it.

In Figure 2.2, the following source data holders can be seen:

- Municipality Delft
- Ministry of Infrastructure and Water Management
- Water authority Hoogheemraadschap van Delfland
- Province Zuid-Holland

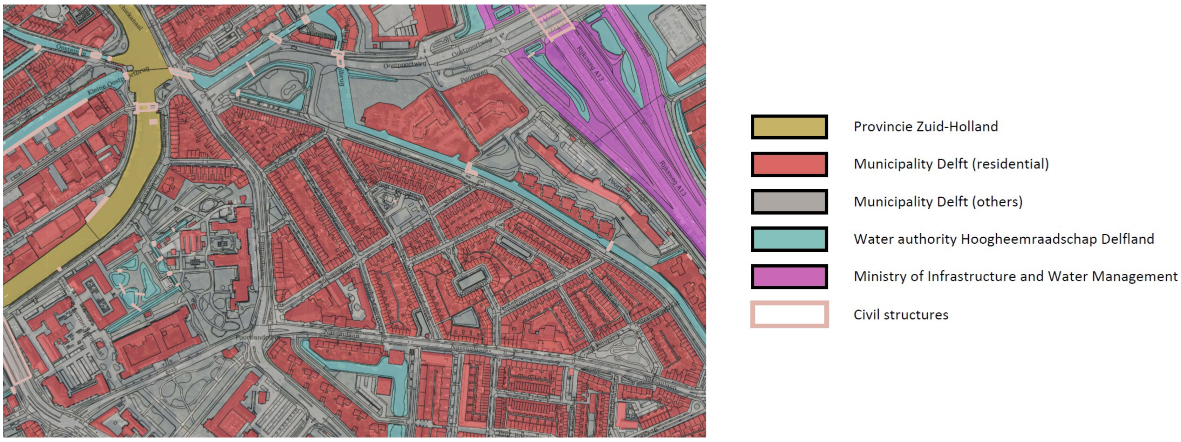

Provincie Zuid-Holland
Municipality Delft (residential)
Municipality Delft (others)
Water authority Hoogheemraadschap Delfland
Ministry of Infrastructure and Water Management
Civil structures

Figure 2.2: BGT with source data holders as described above [5]

Requirements

The specifications and requirements of the BGT are collected and described by Geonovum [6], last updated in July 2020. The main requirements are that the map is a closed, polygonal surface description, without holes. The vertex points are measured points, and bordering objects share the same edge. The required positional accuracy of most the objects is 20 cm, but for objects with less important functionality, the requirement is 40 cm. Source data holders are free to obtain better quality data. For each object class, there is also an update requirement, which is 6 months in most cases (meaning a change should be updated within 6 months).

Maintenance

Keeping the BGT up to date is a difficult task, because the acquisition of data should be done frequently, which does not happen with for instance an existing data-acquisition source as the AHN project. Therefore there should be a way that the BGT can be updated with a different source. The acquisition of data is now done by physically measuring in the field, or use nadir images, which both has upsides and downsides. An in-situ survey is time intensive, and a lot of work; nadir images have less information in vertical direction which makes it for instance difficult to distinguish between a house extension or just an overhang or balcony. Also other BGT objects that are on top of each other are not possible to measure in this way.

2.2. Photogrammetry

Photogrammetry is a remote sensing technology of extracting information from camera images about the physical environment, to map or survey or reconstruct this environment or object. The processing is nowadays almost always done with automated procedures, due to the large scale availability of increasingly detailed images and therefore the large amount of available tie points in an image.
Because of the increased computational capacity of computers, this technique has gone and is still going through a huge development. Images with huge amounts of observations can be linked through algorithms making the system of equations larger and more complicated adjustments can be done (Förstner and Wrobel [3]). On top of that, most smartphones have good cameras, and even apps to do some basic photogrammetry application, making it widely available even for people with less knowledge on photogrammetry.
In this section, the principles of photogrammetry are explained.

2.2.1. Collinearity

The principles of most cameras with a central projection are such, that light travels in an (almost) straight line from the object, through the pinhole of the camera onto the 'negative' sensor in the camera. This sensor is on the negative side of the pinhole. In this way, the image is a central projection of the object. The light travelling in a straight line creates an important geometry condition, also known as the collinearity condition [9].
The collinearity equations are given below, with a visualisation in Figure 2.3.

$$x_a = x_0 - f\frac{r_{11}(X_A - X_L) + r_{12}(Y_A - Y_L) - r_{13}(Z_A - Z_L)}{r_{31}(X_A - X_L) + r_{32}(Y_A - Y_L) - r_{33}(Z_A - Z_L)} \tag{2.1}$$

$$y_a = y_0 + f\frac{r_{21}(X_A - X_L) + r_{22}(Y_A - Y_L) - r_{23}(Z_A - Z_L)}{r_{31}(X_A - X_L) + r_{32}(Y_A - Y_L) - r_{33}(Z_A - Z_L)} \tag{2.2}$$

Where:

x_a, y_a	= Image coordinates of point
x_0, y_0	= Image coordinates of principal point of camera
X_A, Y_A, Z_A	= Position of the point in real world
X_L, Y_L, Z_L	= Position of the camera in real world
r_{ii}	= Elements of the rotation matrix of the camera
f	= Focal distance

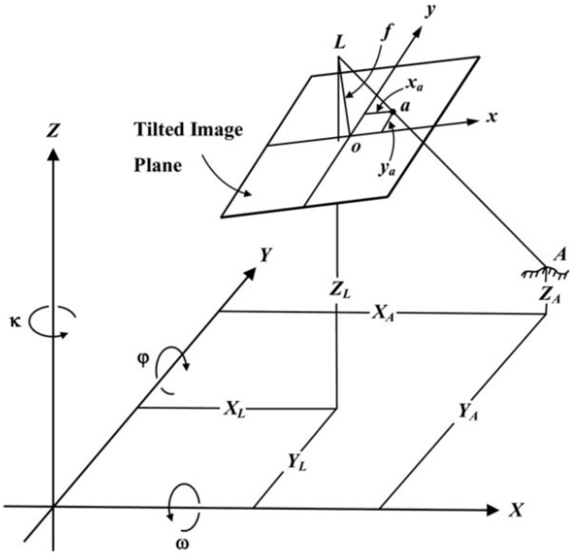


Figure 2.3: The collinearity condition. In this case, the image plane is depicted on the positive side. The collinearity condition states that camera position L, projection a and object point A are on the same line [8].

This condition is, unfortunately, under-determined, because a real-world point has three unknown coordinates (X, Y, Z), and the system has only two equations. So at least a second pair of equations in a second observation (image) is needed to be able to solve these equations, and create a system with still three unknowns but now four equations.

2.2.2. Forward intersection

When a point in the real world is captured by two or more cameras and the point is recognised in all images (image coordinates are known), and the camera position and orientation is known, geometry dictates that the 3D coordinates of the point can be calculated. This is also known as a forward intersection. A matched feature of which the coordinates are found in this way is a successfully triangulated point. A feature must have multiple images matched to the feature, otherwise it will not succeed.

2.2.3. Backward intersection (resection)

As opposed to the previous section, also the opposite of a forward intersection is possible. If the coordinates of at least three points in the real world are known, and of these points also the image coordinates in an image are detected, the exterior orientations can be calculated. The position (X, Y, Z) and orientation (ω, ϕ, κ) of the camera position of that image form 6 unknowns, and there are 6 observation equations making it a solvable system.

2.2.4. Least Squares adjustment/Bundle Adjustment

To solve the system of observation equations that can be formed from the collinearity equations, one can use a least squares adjustment. A step must be taken to linearise the collinearity equations, as these are non-linear. After this, a system of observations and linear observation equations can be initialised. The equation for a non-linear least squares problem is shown in Equation 2.3, where y is a vector with all observations, the A matrix is the system of linearised collinearity equations and in the vector x are all unknown camera orientations, camera positions, and tie point and control point coordinates (Kraus [9], Granshaw [7], Triggs et al. [18]). The control points are measured points on the ground, acting as a frame to reference the model.

Observations ($\underline{y}$)	Unkowns(x)
(x, y) coordinates of all tie points and control points in each image	(X, Y, Z) coordinates of each tie point
(X, Y, Z) coordinates of each control point	(X, Y, Z) coordinates of each image projection centre
	(ω, ϕ, κ) of each image

Table 2.1: Observations and unknowns

$$y = A(x) + e \tag{2.3}$$

Because the observations in y do not match the result of A(x) with initial values x_0, there is a residual vector e. For the first iteration, this is set equal to Δy_0. This first iteration of the solution process is shown in three steps (Equations 2.4, 2.5, 2.6), where x_1 is reinserted in the first equation after the last step. This process continues until the adjustments fall below a certain threshold or a maximum number of iterations is reached. The threshold is usually set based on the accuracy requirements of the problem, while the maximum number of iterations is set to prevent the method from running indefinitely [16].

$$\Delta y_0 = y - A(x_0) \tag{2.4}$$

$$\Delta x_0 = (A_0^T Q_y^{-1} A_0)^{-1} A_0^T Q_y^{-1} \Delta y_0 \tag{2.5}$$

$$x_1 = x_0 + \Delta x_0 \tag{2.6}$$

Theoretical Quality

In most cases, the observations have an uncertainty known in advance. In these cases it is assumed that the observations do not have any correlation, therefore the variance matrix Q_y will be a diagonal matrix with on the diagonal the squared standard deviation of each adjusted parameter, σ^2 of each observation. With this, also the variance matrix $Q_{\hat{x}}$ of the estimated unknowns can be calculated following Equation 2.7 [16]. This is independent from the actual observations, so it even could be calculated beforehand, but in most cases in aerial photogrammetry, it is not yet known what will be observed.

$$Q_{\hat{x}} = (A^T Q_y^{-1} A)^{-1} \tag{2.7}$$

2.2.5. Feature extraction and matching

For all the discussed photogrammetric processing steps, the core condition is that for the matching of two images, the images have points that are recognised in both images. Otherwise, the system of collinearity equations cannot be solved, the points can not be triangulated and no information can be extracted from the images. Handpicking these can be done (and is done in the past), but is with an increasing amount of images this is not recommendable. Multiple methods have been developed to extract such recognisable features automatically, of one is described in this section.

SIFT

Scale Invariant Feature Transform (SIFT, Lowe [10]) is a method to extract distinctive features in images. One of the biggest advantages of SIFT is that it is supposed to be invariant to scaling, rotations and translations. This robustness is one of the main reasons that SIFT is still widely used, although it has been around some time. The invariant features with its highly distinctive description makes it very suitable for the matching against large numbers of features, which is needed in photogrammetry. It works in four steps:

- Scale space extrema detection
- Keypoint localization
- Orientation assignment
- Keypoint Descriptor

First, in the Scale space extrema detection the image is searched for potential features invariant to scale and orientation. This is done by using a difference-of-Gaussian function (DoG). DoG is an approximation of the Laplacian of Gaussian (LoG). The LoG and DoG are both using Gaussian blurred images and down scaling to make a scale invariant feature detection. However, the LoG uses second order derivatives and the DoG uses a subtraction. As a result the DoG is faster, but there is a slight loss in precision. Because a Difference of Gaussian is used, the features are based on the image gradient. Second, in Keypoint localization, the candidate features are fitted to a model to determine location and scale. The candidate features are also evaluated on stability. Third, in the orientation assignment each keypoint is assigned an orientation based on local image gradients. This is added to the location/scale model of the feature. Future operations on these keypoints can use the orientation/location/scale model to make the features invariant to these transformations. The last step is the Keypoint Descriptor step, where a description is generated of all keypoints to create a feature. The description uses local image gradients at an selected scale, which makes it resilient to shape distortion and change in illumination.

Matching the features between images is the next challenge. SIFT creates many features in an image which can all be linked to features from other images. This is done with the Euclidean distance between the features, choosing the feature with the closest description vector. The biggest problem with this method is that the features that are not occurring in other images will be matched to the 'closest' feature in another image, which is wrong. The solution used to overcome this challenge is to use a distance threshold [10].

2.2.6. Dense Matching

Dense matching is a way to match images even more in detail, creating a dense point cloud based on images. Furukawa and Ponce [4] describe in their paper a method to create a dense point cloud in an accurate, dense and robust way. It makes use of the tie points found in the previous photogrammetry steps. The method consists of three main steps; matching, expansion, and filtering. In the first step, corners and other features are identified using a Difference of Gaussian operator. These are matched between images, to find a sparse set of patches. These patches are expanded in the expansion step until the patches cover the entire surface. After this, two filtering steps filter out patches that are either outside of the surface, or inside of the surface.

This algorithm is also incorporated in the COLMAP software package (subsection 2.3.2)

2.3. Photogrammetry software

There are various photogrammetry software packages available, and in this section the most relevant software packages have been described. Methodology that is used in this software is described earlier in this thesis (section 2.2), with some exceptions that will be further explained if needed.

2.3.1. Bundle5

The photogrammetry software of Geodelta 'Bundle5' is an algorithm that is capable of performing a bundle adjustment. Using the extracted and matched features recognised in the images as observations, the software needs all observations in a certain file format. Also a settings file defining the file paths and the uncertainty constraints of the observations should be provided, with in this settings file also the possibility to perform certain checks and tests. One of those is the Fisher test, which is an overall model test. This test is a test to see if the (adjusted) model fits the data within the uncertainty constraints given. This test needs only the covariance matrix and the calculated residuals in the bundle adjustment [17]. Another property of the Bundle software is to calculate the inverse of the A^TA mamatrix using a Cholesky decomposition. This is needed to be able to calculate the covariance matrix of the adjusted observations, as shown in Equation 2.2.4. As there are millions of observations, the inverse of this very large matrix takes a long to compute.

2.3.2. COLMAP

COLMAP is a open-source photogrammetry software package, that implements a Structure-from-Motion method by Schönberger and Frahm [15]. This means that it can process unordered, random oriented images and reconstruct a point cloud of an object or area. In this chapter, the SfM method is described together with the most important components of the COLMAP software package.

Feature extraction

The first step in the SfM is to extract features from all images. This SfM makes use of SIFT (subsection 2.2.5, [10]) to extract features that are recognisable under any radiometric or geometric changes. These are written away in a supporting database structure.

Feature matching

The next step of the SfM method matches images to each other. This can be done in different approaches. The default approach will match every image pair and every feature pair within the image pair. This is of a complexity $O(N_I^2 N_{F_i}^2)$ with N_I the number of images and N_{F_i} the number of features in the image, and for larger image sets this becomes too computatively expensive very fast. Other matching approaches are sequential matching (match only to the next picture), vocabulary tree matching, spatial matching (need location information), transitive matching, and custom matching. With custom matching, the image pairs are manually matched by an input file stating the image pairs.

Reconstruction

When a reconstruction is started, the SfM will start with an initial image pair. If all points are triangulated (or failed) in this pair, the SfM algorithm will search for the next image. This is done by the 'Next Best View' selection method, which chooses the image that 'sees' the most triangulated points. This is constantly updated for all remaining unregistered images. A local bundle adjustment is performed after each registered image, and once the model is grown for a certain percentage, a global bundle adjustment is carried out. When all images are placed, the algorithm is ended.

Dense Matching

The dense reconstruction of COLMAP uses the method explained in section 4.3. It needs a model with image locations and tie points, in the COLMAP database structure. Along with that, it needs access to the images. The output is written to a point cloud, with a number of other files, where a lot of information is stored.

2.4. 3D reconstruction

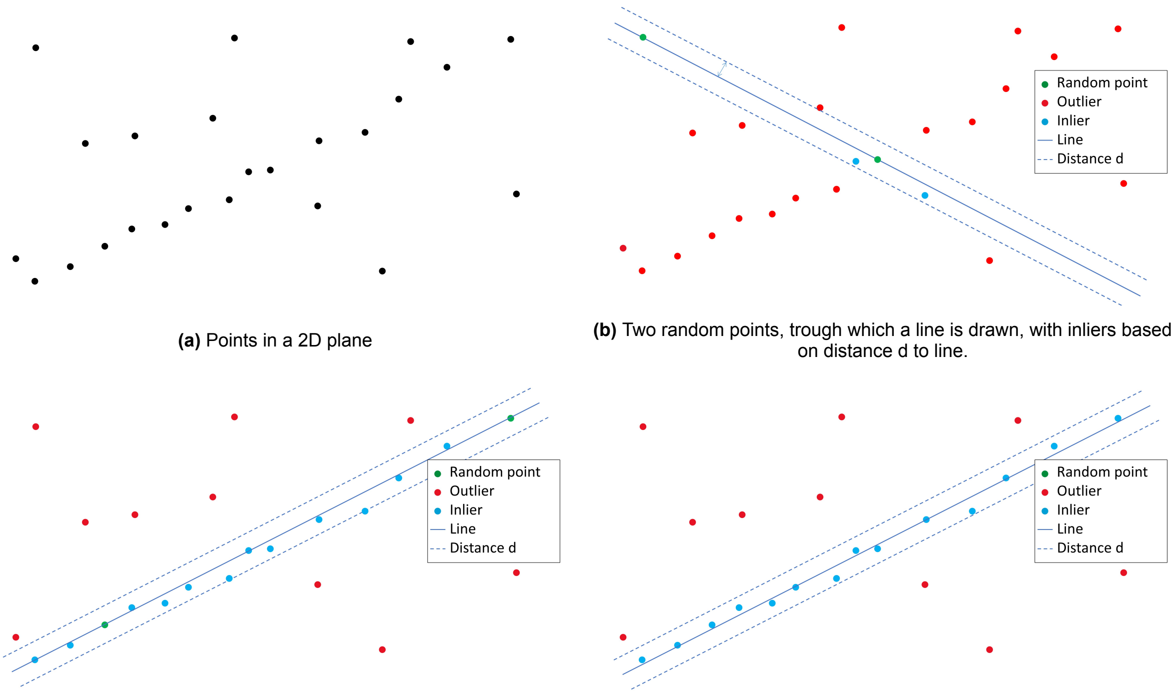
In this section relevant geometric feature extraction/reconstruction methods and existing 3D reconstruction methods are summarised.

2.4.1. RANSAC

A common used geometric feature extraction method to extract features from point clouds, is Random Sample Consensus (RANSAC). This is a method described by Fischler and Bolles [2], to fit geometric features to data with outliers. In earlier parameter estimation methods, the assumption is made that there are always enough good values compared to outliers. The parameters are fitted using all the data, after which the outliers are detected and removed and an improvement of the fit is carried out. In large data sets with a higher number of geometric features, it is impossible to retrieve features in this way because the points in other geometric features act as outliers, and therefore the number of outliers is too large. That is why Random Sample Consensus is proposed, this is a method that is capable of handling those large numbers of outliers.

RANSAC principles

The RANSAC method draws randomly a prefixed number of points from point cloud P. This prefixed number of points should be equal to the minimum number of points needed to span the geometric feature C that is searched for. Those random chosen points span the geometric feature, and all points p_i are checked based on some distance- or other threshold if p_i is a inlier of C. This is done until a threshold number of inlier is reached, or a maximum number of iterations is reached. C with the highest number of inliers is kept as a geometric feature, and all inliers are removed from P and the process is repeated. In Figure 2.4 a 2D example is given, where the a random generated feature (line) is shown in Figure 2.4b with inliers accepted on a distance threshold d. The line with the most inliers is shown in Figure 2.4c, and a refitted line is shown in Figure 2.4d.



(a) Points in a 2D plane

(b) Two random points, trough which a line is drawn, with inliers based on distance d to line.

(c) Result with highest amount of inliers after a number of iterations of random points

(d) On the best result, the geometric feature is fitted again with least squares

Figure 2.4: RANSAC principle with 2D point cloud and a straight line geometric feature, which needs 2 points.

2.4.2. 2.5D models

The existing BGT (section 2.1) is already used to create a 3D (or almost 3D) model. This is the 3DBAG model (Peters et al. [13]), which is developed at the Architecture faculty of TU Delft. This model uses the building outlines in the BGT, and pulls the surface height up to the height of the AHN3 point cloud [1]. This result looks then as if a blanket is dropped over the AHN point cloud. This is a 2.5D model, as for each horizontal point, there is one height level and therefore can not be called a full 3D model.

Model disadvantages

A large disadvantage of this model is that a lot of 3D information is lost, because of the 2.5 dimensional properties of the result. For each vertical point, only one height is adopted, and no information is available of what is beneath the highest point.

2.4.3. PolyFit

PolyFit is an existing method created by Nan and Wonka [11] to extract lightweight polygonal surfaces from point clouds, by intersecting all extracted planar instances, and combining them appropriately, so that a polygonal surface is formed without gaps or holes. This results in a 3D model of an input point cloud.

Candidate face generation

As input, the PolyFit method uses a point cloud that can be noisy and can also contain outliers. As a first step, planes are generated using the efficient RANSAC algorithm, which is a variation of the original RANSAC method proposed by Schnabel, Wahl, and Klein [14] . This results in a point cloud with points that are each assigned either to one or to zero (outliers) planes, and a set of plane equations describing those planes.

To tackle the presence of any undesirable planes, the following refinement step is done to merge the planes. Undesirable planes could be intersecting planes that have very small normal differences, or unsupported planes. These planes are merged based on being within a maximum angle between plane pairs and a minimum amount of shared supporting points, meaning points within the plane distance threshold of both planes.

After this, a number of planes has been found, and with pairwise intersecting, all planes are divided in segments (candidate faces) based on all intersections between the planes. In Figure 2.5, two planes can be seen that intersect. The result is an edge e between the planes, and four candidate faces (Figure 2.5a).

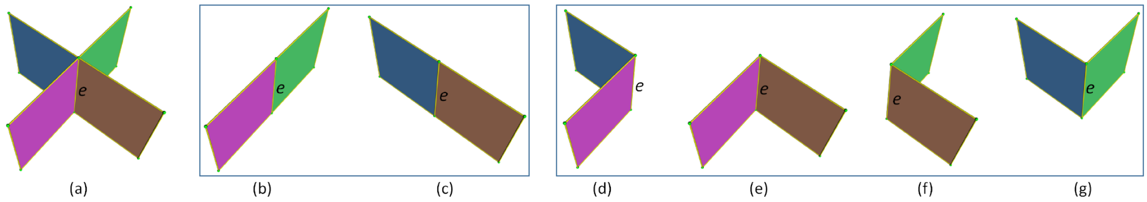


Figure 2.5: Plane intersection: result and selection options. (b) and (c) are non-sharp edges, and (d-g) are sharp edges [11].

Face selection

With the generation of all these plane segments or candidate faces, a selection can be made that optimally describes the geometry of the object. If the plane generation was sufficient, a model can be found using these faces with a complete and closed polygonal surface. Nan and Wonka [11] define three 'energy' terms which are optimised to choose the faces: Data-fitting, Model complexity and Point coverage. Based on these terms, faces are selected. Each edge that is found, is connected to four faces (two planes divided in two). In a closed polygonal surface, the number of faces connected to an edge should be two, so the optimisation process chooses always two or zero faces connected to an edge (see Figure 2.5).

Data-fitting is the term that is defined by how well the model fits through the points. This fitting term

E_f is mainly influenced by two characteristics, described by the equation in Equation 2.9.

$$E_f = 1 - \frac{1}{|P|}\sum_{i=1}^{N} x_i * support(f_i) \tag{2.8}$$

$$support(f) = \sum_{p,f|dist(p,f)<\epsilon} (1 - \frac{dist(p,f)}{\epsilon}) * conf(p) \tag{2.9}$$

For each face, the support is calculated as in Equation 2.9. This is a summation of an equation calculated for each point within distance ϵ of the candidate face f. The term on the left is a distance measure: it is the ratio of maximum distance ϵ and point p to the plane. If point p lies closer to the plane, the term will increase. On the right, there is a confidence term, which is an indication of how confident the model is that the local point cloud is of good quality. This is calculated by two geometric properties of the point cloud at three scales. The geometric properties are the planarity of a local fitted plane and the uniformity of the distribution of the points.

In this way, E_f will be 1 if the point cloud is locally in a perfect plane, uniformly distributed and the candidate face is fitted exactly through the point cloud.

The second term under consideration is **model complexity**. If the data-fitting term was the only defined decision term, it will choose faces that are all well fitted, but leaves gaps if the data has gaps. To balance this, the model complexity energy (E_m) will force the optimisation to avoid a lot of sharp corners, and therefore increases the larger planar regions. It is defined as the ratio of sharp edges in the model, where an edge is considered sharp if the edge has a corner, and planar (not sharp) if the connecting faces are from the same plane.

The last term is **Point coverage**, and this is a measure to calculate how well the points in the data cover a face. The 'energy' of this term E_c is defined as the ratio of uncovered regions in the final model. This is done by calculating the alpha shape area of each face and subtracting this from the total face area.

Optimisation

To get the desired result, a weighted sum of the above described energy terms is minimised. The weights can be adjusted to the desired values, which will give different results. There are some constraints to the optimisation, for example the already mentioned condition that each edge is connected to 0 or to 2 candidate faces. The selection options for each edge can be seen in Figure 2.5b-g.

3 Data Description

In this chapter, an overview is given of the data and the area covered by this data that is used in this thesis. First, the area is described in section 3.1, after that the image data (camera system, image specifications) is described in section 3.2. Lastly, the individual test cases are summarised in section 3.3.

3.1. Area

The study area is defined as the area of the city of Maassluis that is covered by aerial imagery (oblique and nadir). This is a small city located between Rotterdam and the North Sea, along the waterway Nieuwe Waterweg, connecting the port of Rotterdam to the North Sea.

There are some more cities that have available oblique imagery available, but Maassluis has been chosen because it is close to Delft, for any visits in-situ, it has enough variety, and the images are already checked and calibrated and are of suitable quality. Some other places (e.g. Capelle a/d IJssel) are used for testing and getting used to the data.

The area of Maassluis is a mostly flat area with varying terrain, with different style of buildings, nature, and water bodies. In the city Maassluis are residential areas, some high apartment buildings and industrial buildings, but also roads and a highway next to the city. The Nieuwe Waterweg, as mentioned above the outlet of the port of Rotterdam, is next to the city. Outside of the city there are agricultural fields and some forested areas. This results in a varied test area, that will be interesting to look at. The project area can be seen in Figure 3.1.

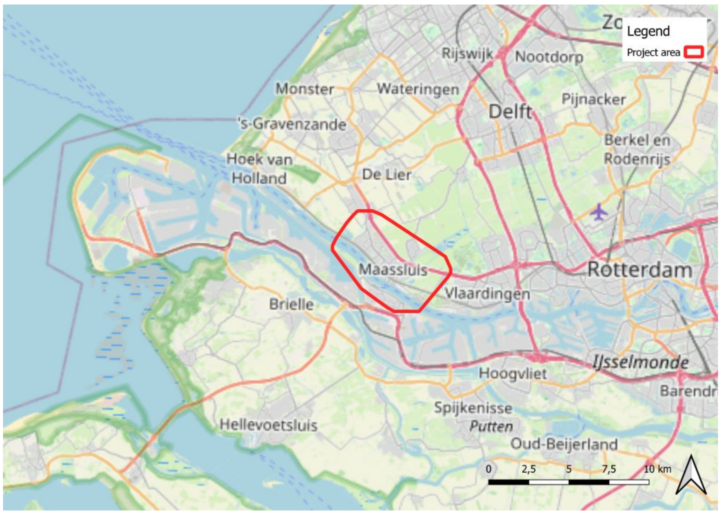


Figure 3.1: Project area (map from OpenStreetMap contributors [12])

3.2. Imagery data

The data used in this project will be mainly airborne imagery data acquired by aerial surveys, capturing images to four sides (Forward, Backward, Left and Right), and also a nadir image. For this there are 8 cameras on board of a plane, 4 pointing to each direction and 4 pointing down. The specifications of each camera can be different In this case the cameras pointing down have a different resolution and/or spectral band than the oblique cameras, and the cameras pointing left and right have a different principal point (Table 3.2). Apart from the images, there is also GNSS and IMU data available.

The imagery data and the adjusted set used in this study is captured by Slagboom en Peeters, a aerial data capturing company in the Netherlands. Apart from the images, Slagboom en Peeters also delivered a set of triangulated tie points which is used for the bundle adjustment after the triangulation of extracted and matched features in this study failed (as described in subsection 5.1.2).

3.2.1. Flight path specifications

The survey at Maassluis is flown at an height of around 600 meters, in strips of 200 meters apart. In those strips, the distance between image positions is around 80 meters. All strips are flown either from west to east or vice versa. This can be seen in Figure 3.2, where all + signs represent one image location with 5 image directions. The image on the cover of this thesis is also a representation of these image cones over a map of the area.

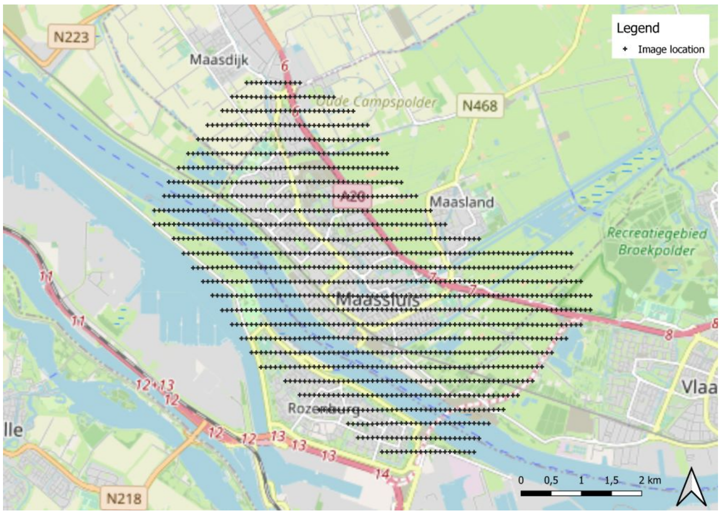


Figure 3.2: Image capturing locations, over the city Maassluis, the Netherlands (map from OpenStreetMap contributors [12])

Because the plane that is used to fly the route does not fly in a perfect straight line, it is not perfectly orientated and it does not take the pictures at exactly known locations, the exterior orientations and positions of the plane are captured by a GNSS and IMU installation on the plane. This is stored in XYZ coordinates that are in the two-dimensional Dutch coordinate system (RD) and the Dutch height system (Normaal Amsterdams Peil) for the XYZ coordinates, and the rotations about the X, Y and Z axis (rX, rY, rZ) in degrees, along with some standard uncertainties in these measurements.

GNSS/IMU	σ range
X, Y, Z	0.016-0.022 meter
rX, rY	0.003-0.004 degree
rZ	0.007-0.011 degree

Table 3.1: Uncertainty range in the GNSS and IMU observations (standard deviation)

3.2.2. Camera specifications

The images are taken with a UltraCam Osprey 4.1 camera, installed on the plane. An image of the camera can be seen in Figure 3.3. In the right picture, the 8 cameras are visible, with the oblique cameras in the hole in the middle of the set up pointing in opposite directions, and the 4 nadir cameras placed around the oblique cameras. The most important camera specifications are visible in Table 3.2. Important to note is the deviation of the principal point in the Left and Right camera, these are shifted 6.68 mm to the 'lower' short side.

(a) (b)

Figure 3.3: UltraCam Osprey 4.1

Camera	Format (px)	Pixel size (μm)	Principal point (px) x	y
2x PAN	14016 x 20544	3.760	7008	10272
RGB	8760 x 12840	6.016	4380	6420
NIR	8760 x 12840	6.016	4380	6420
Backward	10560 x 14144	3.760	5280	7072
Forward	10560 x 14144	3.760	5280	7072
Left	10560 x 14144	3.760	5280	8849
Right	10560 x 14144	3.760	5280	5295

Table 3.2: Camera specifications. PAN (Panchromatic band), RGB (Red, Green, Blue bands) and NIR (Near InfraRed bands) are pointing down. The PAN image is a stitched result of 2 cameras. Backward, Forward, Left and Right are oblique image cones, capturing in red, green and blue bands.

The angle with respect to vertical at which the oblique images are taken is about 45 degrees (measured with respect to the principal point location). The oblique images forward and backward are in landscape orientation, and the images to the left and right are in portrait orientation. The viewing range is 24.3 degrees on the long side, and 18.3 degrees on the short side of the image.

The camera focal point positions are not all in exactly the same position, as this is physically impossible. Therefore, some eccentricities are applicable on the oblique image cones, which are referenced to the Nadir image cone. This is visible in Table 3.3.

Camera	X [mm]	Y [mm]	Z [mm]	ϕ [degrees]	ω [degrees]	κ [degrees]
Nadir	0.000	0.000	0.000	0.000	0.000	0.000
Backward	-106.712	-21.692	-7.892	0.029	-44.993	-0.077
Forward	-106.878	-99.972	-129.667	0.029	44.968	0.030
Left	-35.468	-86.232	5.913	44.992	-0.078	0.014
Right	-115.159	-85.854	-144.906	-44.995	0.056	-0.009

Table 3.3: Oblique image cone locations in reference to the nadir cone.

3.2.3. Received data

The imagery data is captured and already processed by a third party, Slagboom en Peeters. This is an aerial imagery company, specialised in capturing and processing aerial images. In addition of the images, also a processed bundle adjustment is included in the data delivered. Feature detection, feature matching, and triangulation and bundle adjustment are already carried out, resulting in an adjusted data set consisting of a point cloud of the following things:

- File with locations of tie points and control points with XYZ
- File with tie point and control point locations in the images in pixels
- File with measured GNSS and IMU information of image locations

3.3. Individual test cases

In Maassluis, a number of different kinds of buildings and surroundings are chosen, those are described below, and shown in Figure 3.4. The locations have been carefully selected to reflect some variety of situations, starting somewhat simpler in Figure 3.4a and gradually increasing in complexity and noise (for building reconstruction) towards Figure 3.4f. All these pictures are taken from the available imagery data.

These buildings will be used in the second part of the method, and this selection is made to be able to test the method that is used on different types and complexities of buildings.

Warehouse: Figure 3.4a is a simple industrial building, with box-like walls and roof. This can serve as a simple test for the method, with low complexity and not much noise around the building in the form of vegetation or other objects.

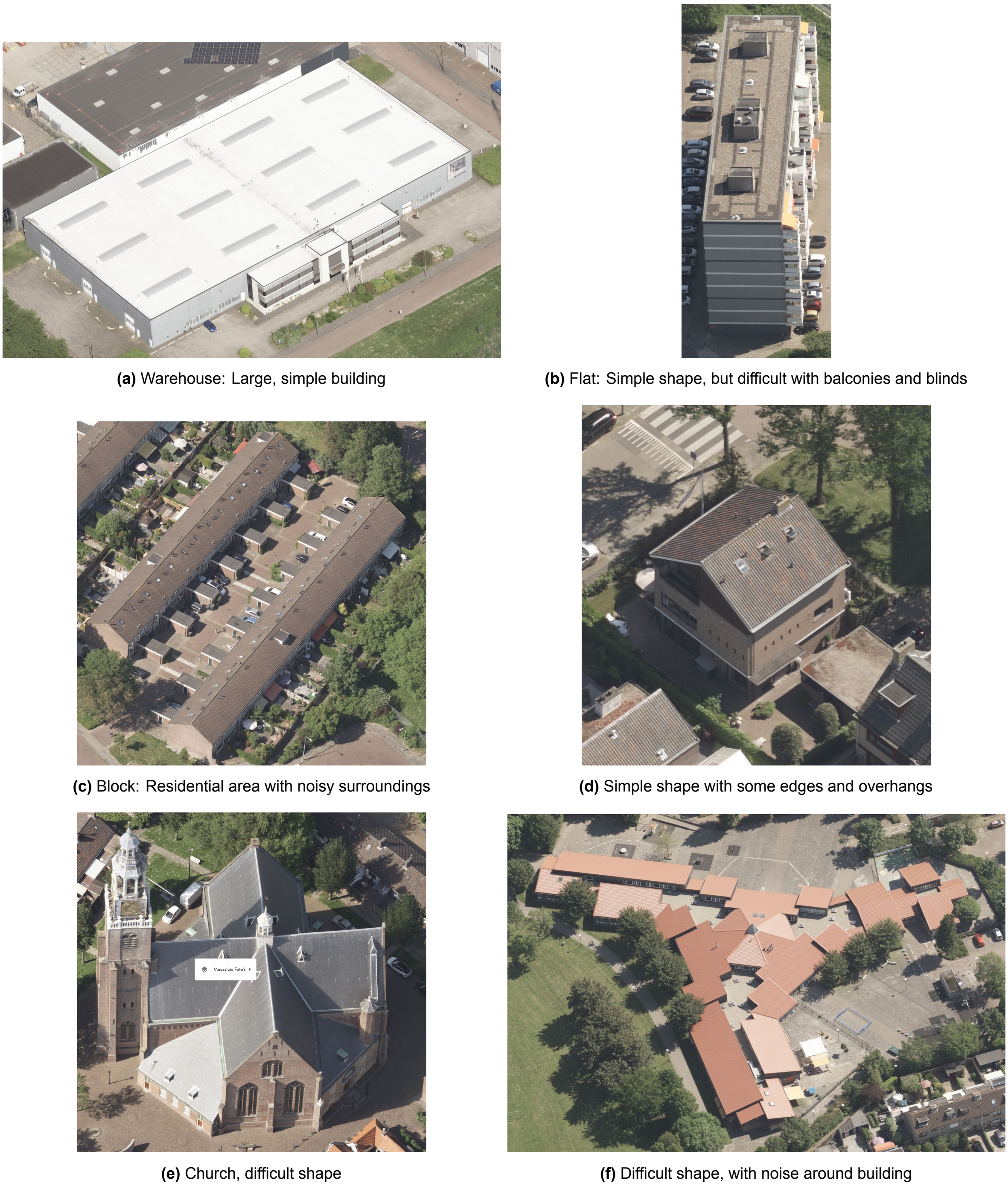
Flat: Figure 3.4b is a high apartment complex, and it has a very basic building shape. One side however, has a very noisy outline, with balconies and canopies. It will be interesting how this will be processed in the 3D modelling.

Block: Figure 3.4c shows a residential area, with a simple house structure in two terraced houses rows. Added complexity is the garage in each individual house, with some cars in between. This creates a noisy surrounding, that can be quite difficult to process. Also the gardens and the canopies in the gardens are noise.

House: Figure 3.4d is an image of a free-standing residential building. the building itself has some irregularities, with a small extended roof and other indentations. The building is positioned in the shadow of some large trees, which will influence the quality and visibility from that side in images, and that is the interesting challenge in this building.

Church: In Figure 3.4e, the church of Maassluis is a moderately complex building, with some noise in the planes forming the building in the form of indentations and turrets. Especially these turrets will be a challenge for the algorithm, and really test the robustness of the method.

School: Figure 3.4f is a school building, with a lot of different roof structures and building parts. There is also a lot of vegetation around the building, which, in addition to the high complexity of the building itself, will be an enormous challenge for the algorithm.



(a) Warehouse: Large, simple building

(b) Flat: Simple shape, but difficult with balconies and blinds

(c) Block: Residential area with noisy surroundings

(d) Simple shape with some edges and overhangs

(e) Church, difficult shape

(f) Difficult shape, with noise around building

Figure 3.4: Test cases

4

Methodology

In this chapter, the methodology used to research the quality of the photogrammetry and 3D reconstruction with oblique imagery is described and explained. First, the extraction and the matching of tie points in the images is explained in section 4.1. After that, the triangulation of the tie points and the adjustment of the observations is discussed in section 4.2. The last step to a dense point cloud of specific chosen buildings with dense matching is described in section 4.3. After this, the reconstruction of a 3D model is discussed, with the processing of the point cloud to extract planes with a RANSAC based method in section 4.4, while the output of this RANSAC method is not good enough to perform the 3D model generation so this will not be done (see chapter 6).

4.1. Feature Detection

For the feature detection in the image set, a SIFT algorithm is used, as described in subsection 2.2.5. This will find the needed object points in the images, to be able to match the images to each other. The SIFT algorithm implemented in COLMAP searches for scale-invariant features, which is even more important in oblique imagery, because the scale in different images can differ a lot. As SIFT detects on pixel-based level, the standard deviation of image coordinates of detected features is around one pixel, and therefore this is used in further initial input for the bundle adjustment performed here-after.

4.1.1. Feature Matching

Matching is then done on image pairs, comparing the detected SIFT features and checking which features are a good match. To decrease computation time and not having to match each possible image pair, the images are matched based on a spatial query. This is not as straightforward as it would be with only nadir images, because oblique images capture an image of an area in a different place than the camera position on ground level. Therefore, a method is proposed here to match images based on the location of the image plane on ground level, by displacing the camera location to directly above the image plane. This principle is shown in Figure 4.1.

The angle at which the oblique images look down in this set up is around 45^o, so the images should be displaced in the direction the cameras are taking pictures with the same distance as the height of the plane above the ground (Pythagoras). In this case, this is 600 meters.

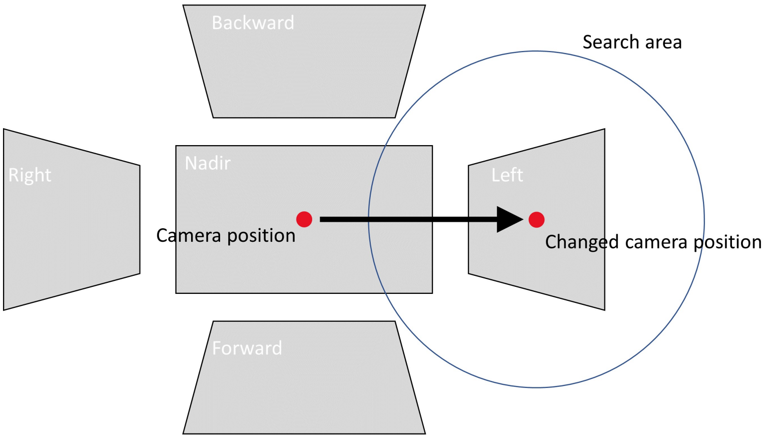


Figure 4.1: Virtual displacement of image location in image pair search

After this, image pairs are found using a nearest neighbour radius search. The feature matching algorithm will be given a custom list of images that have overlap, so just these images need to be matched, which speeds up the process significantly.

With an n amount of images, the total image pairs if exhaustively matched is of complexity O(n^2), because every image has to be matched to all other images, giving the total amount as

$$image\ pairs = \frac{n(n-1)}{2}, \tag{4.1}$$

with n = number of images.

If a search radius is used to find image pairs and images are only matched based on a maximum distance threshold, this complexity approaches O(n), if the distance threshold is significantly smaller than the extend of the image set. The total amount of matches is described as

$$image\ pairs = N * n, \tag{4.2}$$

with N = number of neighbours for an image.

If the images are spatially equally distributed (which is to be expected of such an image set), N will be constant. In this case, a typical value for n will be around 50 images. This way of matching results in a significant decrease in calculation time.

In contrast to an image set with only nadir images, the orientations of the images in an oblique image set are not all the same. This can cause difficulties in matching the images with different orientations, because features are recognised based on contrast gradients (subsection 2.2.5). If a feature is viewed from a different orientation and it has a different background, the contrast gradient is different and will not be matched, while it is the same object.

4.2. Bundle Adjustment

The observations and exterior and interior image parameters can be adjusted with a non-linear least squares adjustment as in section 4.2, adjusting all camera orientations and tie point locations. The goal of this step is to estimate how well all observations will fit, and in that way an estimation of the quality of the observations is derived. The adjustment provides a theoretical standard deviation for all adjusted observations, which will be used as an indication of how good for instance a tie point location is observed and how well the orientation of the camera is estimated. This is the quality metric used in the assessment of the quality of the photogrammtric output.

4.2.1. Triangulation

To get an estimation of the real world location of the tie points, a triangulation is done with the estimated camera positions from the GNSS/IMU data and the matched features in the images. A series of forward intersections (subsection 2.2.2) is done to place (triangulate) all matched features in the 3D environment. This is done by drawing a line through the focal point of the image, the point on the image plane and if three or more images can see a point, the intersection of those lines is the real world location of the point [9].

Exterior image orientations
COLMAP is able to perform a reconstruction of the image locations and tie point locations. However, this will take a long time when done without giving COLMAP the image exteriors, and in addition the reconstruction is done in an arbitrary coordinate system, which must then be scaled and transformed to the RD coordinate system. This is because COLMAP will place the images in an arbitrary location, as it has no information on image exteriors. These exterior image orientations must be combined first.

The images are oriented in two ways. Firstly, the plane has a location and orientation. This is described by an X, Y and Z coordinate measured by a GNSS system, and a roll, pitch and yaw described by an omega, phi and kappa angle. Secondly, the image cone is oriented with respect to the plane, giving it eccentricities for each oblique image on the XYZ and $\omega\phi\kappa$ from the nadir cone (plane orientation). In Figure 4.2, this is visualised on the left and on the right.

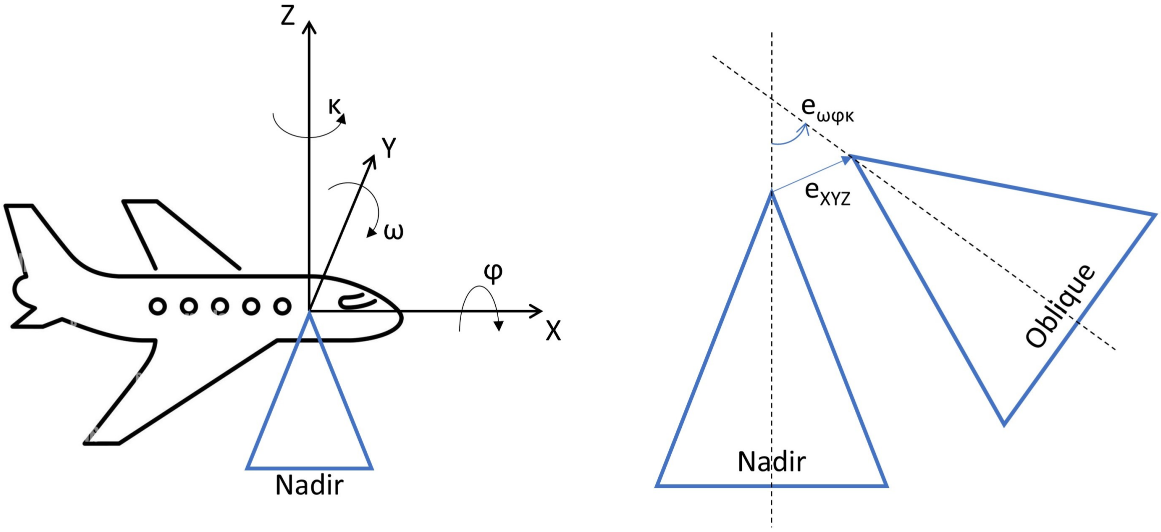


Figure 4.2: Orientations of the plane and oblique cones

First, the yaw, pitch and roll of the plane has to be combined with the eccentricity of the oblique image cones, because the orientation of the plane will influence the position and orientation of the optical centres of the oblique image cones. That is done using the rotation formulas described below. The equations in Equation 4.3 are the rotation matrices in one rotation axis. When the rotations of the plane and of the eccentricities relative to the plane are combined, they give the rotation matrix R for all 5 image directions, in each image location. From this rotation matrix, the image cone orientations ω, ϕ, κ can be determined and these are the image cone orientations relative to the RD system.

$$R_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos(\phi) & -sin(\phi) \\ 0 & sin(\phi) & cos(\phi) \end{bmatrix}, R_y(\omega) = \begin{bmatrix} cos(\omega) & 0 & sin(\omega) \\ 0 & 1 & 0 \\ -sin(\omega) & 0 & cos(\omega) \end{bmatrix}, R_z(\kappa) = \begin{bmatrix} cos(\kappa) & -sin(\kappa) & 0 \\ sin(\kappa) & cos(\kappa) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.3}$$

$$R = R_z(e_\kappa) * R_y(e_\omega) * R_x(e_\phi) * R_z(yaw) * R_y(pitch) * R_x(roll) \tag{4.4}$$

To determine the difference in the image cone positions dX, dY & dZ in RD XY and NAP height Z direction, the eccentricities e_X, e_Y & e_Z must be rotated with only the plane rotations, this is done with the equation below.

$$\begin{bmatrix} dX \\ dY \\ dZ \end{bmatrix} = \begin{bmatrix} e_X \\ e_Y \\ e_Z \end{bmatrix} * R_z(yaw) * R_y(pitch) * R_x(roll) \tag{4.5}$$

These are the displacements of the oblique image cones from the nadir image cone location in X, Y and Z direction.

Conversion to Quaternion representation
COLMAP uses quaternion rotation representation, and therefore the Euler representation of the exterior parameters of the images need to be converted to quaternion representation. Quaternion representation is much used in gaming engines, and does not suffer from the Gimbal Lock as Euler orientation does. This angle representation has 4 parameters, instead of the three in Euler representation. It is possible to calculate these four parameters from the rotation matrix found in the previous section.

4.2.2. Bundle adjustment input

As input, the adjustment needs estimated locations and orientations of the image, which are retrieved from an onboard IMU and GNSS system. Also the locations of the tie points in the images are used as adjustment input, and estimated 3D coordinates of the tie points are also required which is estimated in the triangulation. Another required input is the uncertainty in these observations. The adjustment is given three different input sets:

- Nadir
- Oblique
- Nadir + Oblique

These sets are chosen to be able to see the difference between the Nadir and the Oblique image set, and also analyse the result of the combined set. The adjustment is done with only the observations (image exteriors, tie points, etc) belonging to a specific set. A tie point belongs in a set if it is visible in images belonging in a specific set.

4.2.3. Bundle adjustment output

The expected output of the bundle adjustment is adjusted values for all observations that are provided to the adjustment. Next to that, the estimated standard deviations calculated based on the provided uncertainties are expected to be an output. A lot more is generated by the Bundle5 software, but is not of importance here. For each adjusted image set (as in subsection 4.2.2) there is an adjusted set of observations.

4.3. Dense Matching

From the bundle adjustment, a dense match of the images will be done to get a dense point cloud model. As the goal was to propagate uncertainty in the complete pipeline, a way needs to be found to do this also in the dense matching. Proposed is to use the number of observations of a tie point, and couple this to its theoretical standard deviation, as the number of observations is one of the biggest contributors to the accuracy of a point. This will be supported by the analysis of the bundle adjustment results. In the dense match result, the output that is needed is the number of images a dense matched point is visible in, to be able to connect this to the quality quantification.

4.3.1. Input

As input for the dense match, only the images are selected that see the specific building. With this image collection, a dense match is done with the COLMAP software as it is described in subsection 2.2.6 and subsection 2.3.2. The preferred way would have been to use the results of the bundle adjustment as exterior orientation input of the selected images, but the amount of work that would have gone in converting the bundle adjustment output to the COLMAP configuration is outside of the scope of this thesis, and not necessarily needed in the testing of the proposed method.

4.3.2. Output

From the files that COLMAP produces, the following variables are of interest and stored for further use: The coordinates of the points in the dense point cloud, the RGB values of the points and the number of images a point is visible in. This last value is used in the propagation of the quality of individual points.

4.4. Weighted RANSAC

The proposed RANSAC method uses the quality of the points, propagated through all previous steps, to estimate planar features in the obtained dense point cloud of a building. Normally, it randomly chooses three points to span a plane, after which a cost function determines how good this plane is. If the plane is deemed good enough, the plane equation is stored and the points belonging to this plane are removed from the point cloud. Where the RANSAC method in this thesis differs, is that the three points that the plane is spanned with are not chosen randomly, but weighted randomly, preferring points with a higher quality. This should give a result improvement. In the least squares evaluation of the plane the quality of the points also is used as a weight (weighted least squares), to increase their influence over the points with a lower quality.

4.4.1. Quality propagation

To propagate the quality from the bundle adjustment to the dense match, proposed is to use the combined theoretical standard deviations in XYZ directions as a weight as shown in Equation 4.6. Dense matched points receive the same weight as tie points with the same number of observations. There is a high correlation between the number of observations and the theoretical quality of tie points, and this characteristic is propagated to the dense matched points.

The used value for each point is the length of the vector spanned by the standard deviation in X, Y and Z direction. The standard deviation values that are used are the theoretical standard deviation values of the Nadir + Oblique adjustment, of which the results are described in section 5.2. This is shown in Equation 4.6.

$$weight = \sqrt{\sigma_X^2 + \sigma_Y^2 + \sigma_Z^2} \tag{4.6}$$

A polynomial is fitted to the standard deviation medians per observation number, and this fit is used as value for the dense matched points. The result of this is further discussed in the results chapter.

4.4.2. Weighted Random point sampling

The quality of a point is used as weight in the random sampling of three points to span a preliminary plane. The chance that a point is chosen, is directly related to the weight. This done by generating a random number between 0 and the total sum of weights ($\sum_{i=1}^{n} w_i$) of all points. Iterating through all the weights, if the random number is lower than w_i, that corresponding point is chosen. If not, w_i is subtracted from the random number and w_{i+1} is evaluated in the same way.

When a point is chosen, the weight of that point is set to zero to make sure it is not chosen twice, because three unique points are needed to be able to span a plane. The complete algorithm can be found in Algorithm 1.

Algorithm 1 Weighted Random points

```
$candidates \leftarrow P_0...P_n$
$weights \leftarrow w_0...w_n$
$NP \leftarrow number\ of\ needed\ points$
for $N \leftarrow 1, NP$ do
    $random \leftarrow 0, sum(weights)$
    for $i \leftarrow 0, n$ do
        if $random \leq w_i$ then
            $points_N \leftarrow P_i$
            $w_i \leftarrow 0$
            break
        else
            $random \leftarrow random - w_i$
        end if
    end for
end for
return $points$
```

A	w_1 = 10
B	w_2 = 5
C	w_3 = 20
D	w_4 = 5
E	w_5 = 10

Table 4.1: Example weighted random selection. Sum of weights is 50.

In the example in Table 4.1, a random number between 0 and 50 (24) is generated.

- Iteration 1: checks if 24 is smaller than w_1. No: 24 - 10 = 14
- Iteration 2: checks if 14 is smaller than w_2. No: 14 - 5 = 9
- Iteration 3: checks if 9 is smaller than w_3. Yes: point C is chosen

After this, the weight for point C is set to zero and a new sum of weights is calculated, and the process is repeated for a second weighted random point. Point C can not be chosen again because it has now weight 0.

4.4.3. Plane extraction

Span plane

Based on the random generated trio of points (P, Q, R), a plane is defined. A plane is defined by the equation $Ax + By + Cz + D = 0$, where [A, B, C] is the normal of the plane. The normal of a plane is perpendicular to the plane, and therefore it can be calculated as the cross product of vectors between PQ and PR. By filling in XYZ of a point, D is found.

Inlier check

In this case, a point (P_i) is marked as inlier of the plane if it meets two thresholds. The first threshold is distance d_i from point P_i to the plane (f). The second threshold is the angle difference $\Delta\phi$ between the point normal, calculated based on its 6 nearest neighbours, and the normal of the plane. The number of 6 nearest neighbours is chosen arbitrarily, it is enough neighbouring points to have a reliable normal and it is not the variable to be investigated here. d_i and $\Delta\phi$ are calculated with equations 4.7 and 4.8.

$$d_i = \frac{|A_f * x_{P_i} + B_f * y_{P_i} + C_f * z_{P_i} + D_f|}{\sqrt{A_f^2 + B_f^2 + C_f^2}} \tag{4.7}$$

$$\Delta\phi = \text{arccos}\, \frac{|A_f * N_{x_P} + B_f * N_{y_P} + C_f * N_{z_P}|}{\sqrt{A_f^2 + B_f^2 + C_f^2}} \tag{4.8}$$

d_i = Distance between point P_i and the plane
$\Delta\phi$ = angle between normal P_i and plane normal
A_f, B_f, C_f, D_f = Plane parameters of plane f (Ax + By + Cz + D = 0)
$x_{P_i}, y_{P_i}, z_{P_i}$ = Coordinates of point P_i
$N_{x_P}, N_{y_P}, N_{z_P}$ = Normals of point P

Fit plane
RANSAC uses the notion that in every iteration, a random set of points is chosen to span the geometric feature. After the amount of inliers is calculated, a check is done to see if there are more inliers than in any previous iteration. If yes, a new plane is fitted based on all inliers using a weighted least squares. In this way, the quality of a point is used again to fit the plane, which should improve the plane result. Again is checked for all points whether they are inliers, following the same procedure as before. The plane with the largest amount of inliers is kept and stored as the extracted plane.

4.5. Assessment metrics

The metrics on which the results will be assessed are described here, for the most important results.

4.5.1. Feature Matching

The feature matching results in image pairs matched to each other with a certain amount of tie points that could be matched. For the assessment of the spatial matching method, there will be looked at the difference between the number of matches for the spatial matching method with displaced image locations for oblique images and a brute force method matching every image pair. Also the number of matches between image pairs will be an interesting result to look at, because this will show how strong the connections between certain images is.

4.5.2. Bundle Adjustment

For the results of the bundle adjustment, an analysis of the theoretical standard deviations will be done. This is an theoretical indication of the quality of the variables. The metrics that will be assessed with are the results of the output of the bundle adjustment in the covariance matrix of the estimated variables, such as the standard deviations in X, Y and Z direction of the tie points, and the standard deviations in X, Y and Z of the image locations, and the standard deviation in the image orientations in ω, ϕ, κ.

4.5.3. RANSAC

To assess if the weighting method improves results of the RANSAC method, the results will be assessed on two different output statistics. The method generates a number of planes found, and classifies every point to one of the planes or as an outlier. The assessment will be done on the amount of planes found for each method, and the ratio of points classified as part of a plane. A third assessment criteria will be how 'fast' the planes are found in terms of number of iterations. This will consist of a comparison between the two characteristics mentioned before with 100 iterations, or 1000 iterations. The decrease in iterations means a method that needs less 'luck' will perform better and converges to a better result in less iterations.

Because there is a certain randomness involved in the RANSAC process, the results of an average of 100 runs will be used as result.

5 Results

In this chapter, all generated results will be presented. This is divided in four sections. In section 5.1, the results of the feature detection and matching are discussed, and how the matching compares to exhaustive matching. In subsection 5.1.2 also the triangulation results in COLMAP (subsection 2.3.2) are presented. The next section, section 5.2, the results of the bundle adjustment analysis are shown. After this, the results of the dense matching are discussed in section 5.3. In section 5.4, the result of the weighted RANSAC adaptation are shown and the geometric feature extraction method is analysed.

5.1. Feature Detection and Matching

5.1.1. Match matrix

For the results of the matching, the match matrix can be plotted to to visualise the amount of matches between image pairs. First, the matching methods will be compared using a subset of 145 images, after which the full match matrix is analysed.

Method comparison

In subsection 4.1.1 is explained how the required number of matches can be brought down significantly, and first a comparison of this is shown in Figure 5.1 on a subset of 145 images (instead of full 6890). Here, the exhaustive matching algorithm, matching all images to each other and the proposed matching matrix based on spatial matching are compared. On the axes, the individual images are plotted. From left to right and from up to down, the Backward, Nadir, Forward, Left and Right images are placed and the colour red indicates the amount of matches between the image pair.

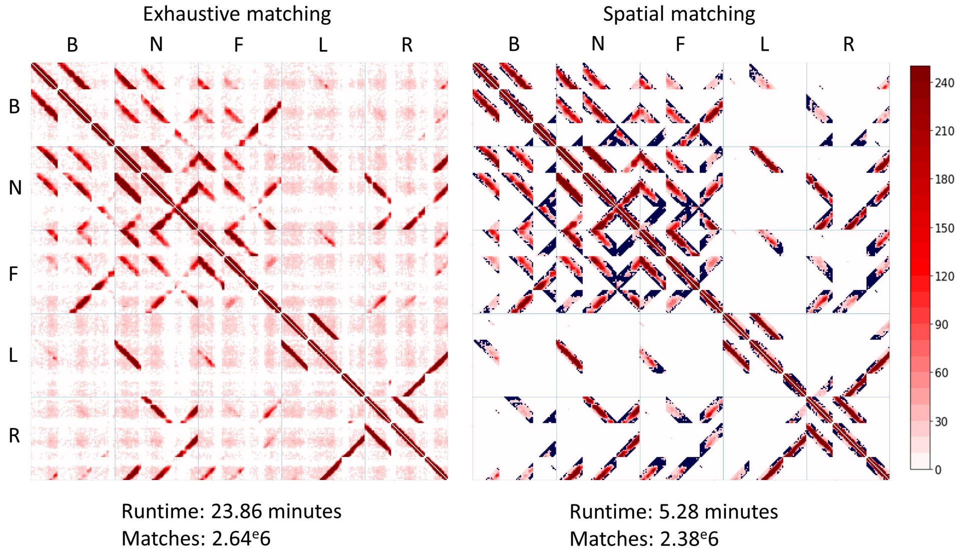


Figure 5.1: Exhaustive matching versus the spatial matching based on displaced image locations. In the right picture, the purple patches are image pairs that are matched, but have no matches.

In general, from these match matrices can be derived that images in the same flight line and in the same orientation have a relatively large amount of matches, because they have larger overlap. This is deduced from the diagonal features pointing from top left to bottom right. Another matching characteristic is the features in perpendicular direction, these are images matching between different flight line directions, 'close' to each other in opposite directions.

The result of the subset shows that the matches are concentrated in the expected places. The number of matches in the spatial matching method are 10 percent less, but this is not very significant and these matches also are incorrect matches between images that have no overlap at all. The running time meanwhile drops significant, which will be even more when the complete image set is evaluated.

Full match matrix

In Figure 5.2, the complete match matrix for the entire image set is visualised. As this is 6890x6890 images, a lot of detail is lost in the visualisation. Nevertheless, it gives a complete overview of which image orientations are connected to other orientations, and how well this connection is. A summery of the amount of matches between each orientation is shown in Table 5.1.

This result shows that the Right and Left images are connected very poorly to the Forward and Backward images. When looking at the numbers, the connection in these orientations is at least 10 times less than in other direction. Furthermore, between Right and Left, and between Forward and Backward, the connection is mainly from matches between different flight line directions, making the image orientations effectively the same. This is recognisable as the bottom-left to upper-right lines in for instance the Backward and Forward matching box, as explained earlier. A Forward image in one flight line is in the same direction as a Backward image in a flight line in the opposite direction. Therefore, matching between different oblique image orientations is not very effective. Here plays the connection to the Nadir images an important role, because the Nadir images have more matches to all oblique orientations, connecting those together.

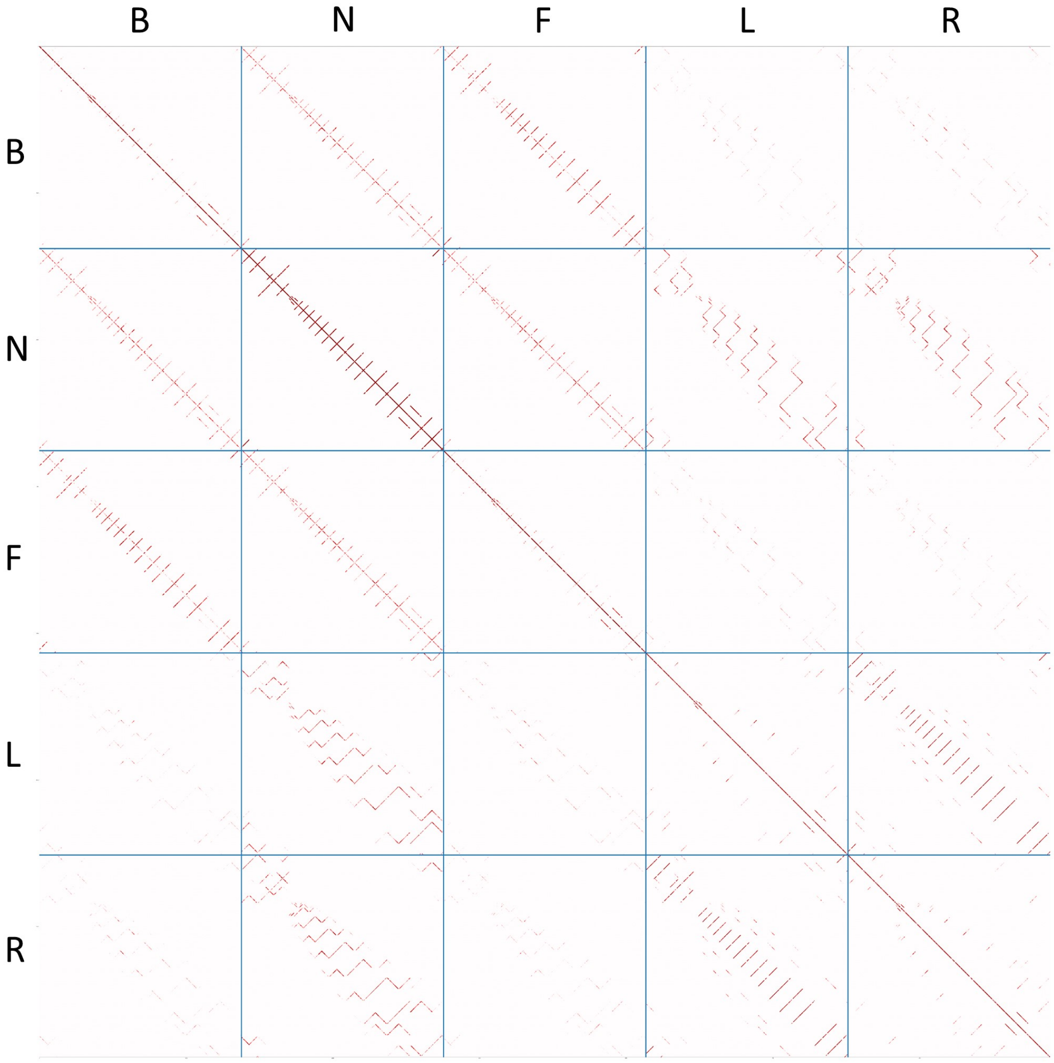


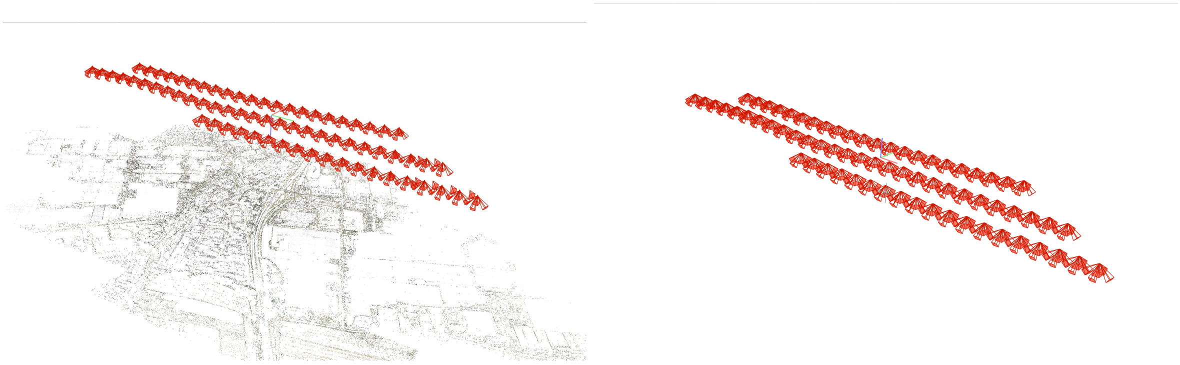
Figure 5.2: Spatial feature matching on the complete image set of Maassluis.

	B	N	F	L	R
B	5.13	1.30	1.93	0.19	0.22
N	-	11.96	1.32	1.79	1.85
F	-	-	4.99	0.22	0.20
L	-	-	-	4.21	2.62
R	-	-	-	-	4.25

Table 5.1: Total number of matches between two image orientations (x100000).

5.1.2. Triangulation result

After the images are matched, the matched features can be triangulated as tie points. This is done with help of COLMAP. In small amounts of images, the triangulation can be done without estimations of camera positions and orientations, where COLMAP then uses the Structure from Motion method to place all cameras. A result of this is shown in Figure 5.3a, on the same subset of 145 images as in the previous section. In red, the camera position and orientations are visible as image cones, and the triangulated tie points are visible as points.



(a) Triangulation using the SfM method in COLMAP. In red the camera positions on the fligth line and in grey the tie points. Not suitable for large amounts of images.

(b) Converted orientations from measured values, visualised in COLMAP with quaternion representation

Figure 5.3

So to be able to triangulate all points, COLMAP needs preset camera locations and orientations. With the GNSS and IMU measurements, and the eccentricities, this is calculated with the rotation matrices, and converted to the required quaternion representation. The result of this conversion is shown in Figure 5.3b. Compared to the result of the SfM method in Figure 5.3a, this is likely to be an accurate result, because all image cones are in the expected places and have the right orientation. However, when the guided triangulation is carried out, COLMAP is not able to triangulate the tie points. This result is in Figure 5.4, and it can be seen that very few points are triangulated and placed in the model, while much of the camera positions are adjusted to wrong positions, for example the image cone completely dropped under the flight line. Unfortunately, this is not solved, and this is why in further work the triangulated and adjusted tie point set from Slagboom en Peeters (subsection 3.2.3) is used.

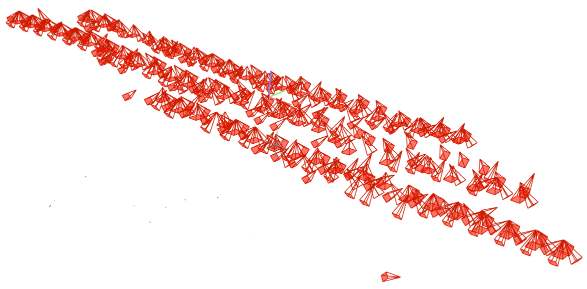


Figure 5.4: Failed triangulation attempt, COLMAP was not able to triangulate the tie points with preset image orientations. The image cones have more 'random' orientations, and do not agree with results in Figure 5.3. Also no triangulated tie points are visible.

5.2. Bundle Adjustment results

In the results of the bundle adjustment on all imagery, it is tried to get a result with the most constraint input uncertainty, as long as the bundle adjustment is still accepted. The bundle adjustment results in adjusted parameters for the tie points, control points and image coordinates and orientations. From the method explained in section 4.2 and in the literature in subsection 2.2.4, also the standard deviations are calculated and these results are visualised and interpreted in this section.

5.2.1. Tie points standard deviations

In this section, the results of the bundle adjustment will be analysed; More specific, the tie points and their standard deviations will be discussed. For all tie points, also a theoretical standard deviation is calculated as described in subsection 2.2.4 and section 4.2, and for the three different sets of images (Oblique, Nadir, and Nadir + Oblique), these are shown in the images in Figure 5.5.

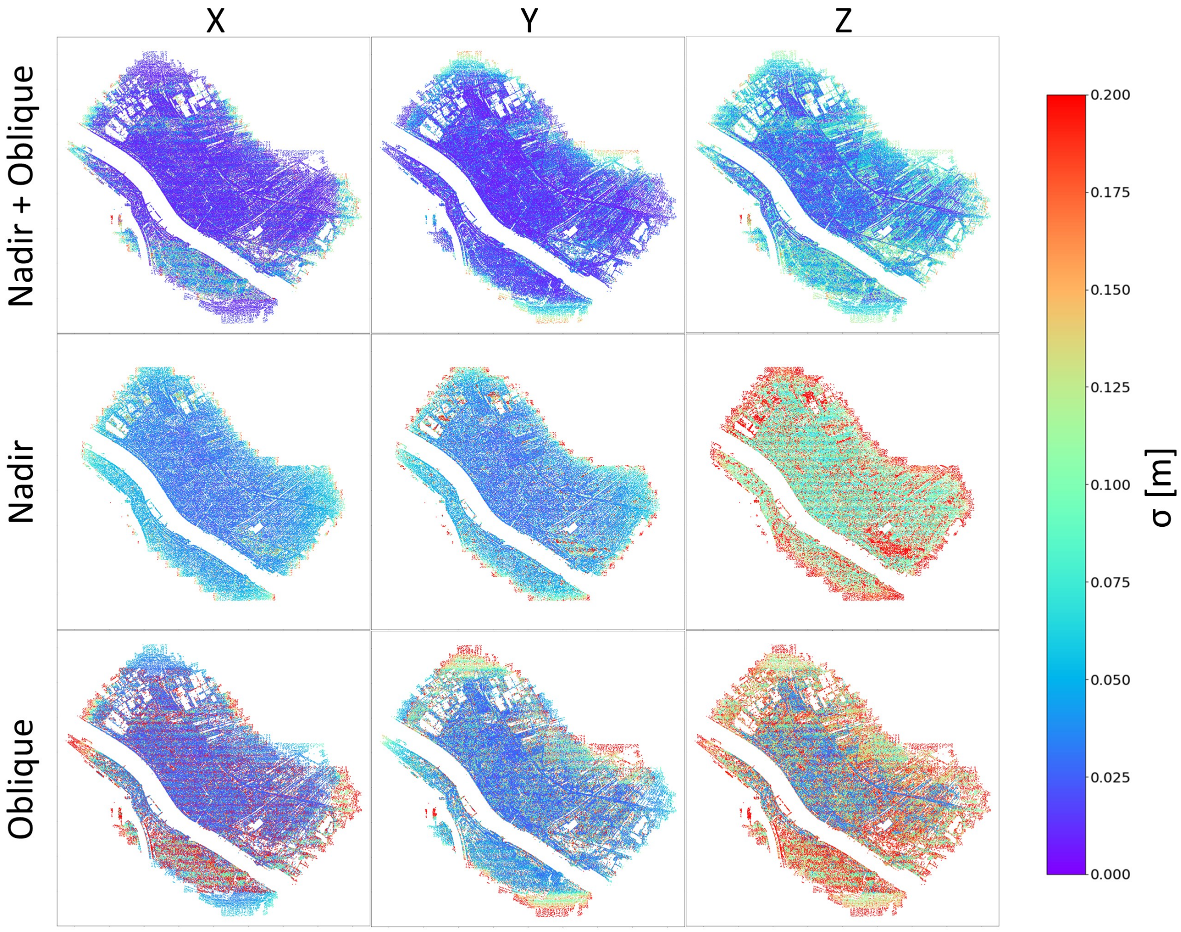


Figure 5.5: Standard deviations of the tie points in X, Y and Z directions, in three image sets: Nadir + Oblique images, only Nadir images, and only Oblique images.

From the results in Figure 5.5, there are multiple things that immediately draw attention. In all configurations, the standard deviation seems worse in Z direction, which is expected. Also, in most images there is a horizontal line pattern visible, which originates from the fact that the overlap in Y direction is worse than in the X direction. This results in higher standard deviations in the strips between the flight lines, because there are less observations of the tie points.

Another general observation can be made about forested areas. In the bottom-right of the area, there is an area with patches of forest, and in all configurations there are less tie points and higher standard deviations. This is most clearly visible in the nadir Z direction. The decrease in quality here can be due to the fact that features in trees are less recognisable, and less features are matched. Therefore less tie points are used as observations, and with fewer observations the bundle adjustment is less redundant and the quality will be worse.

Nadir

In the Nadir image set results, the difference between the X and Y direction and the Z direction is large, especially to the borders of the area. This difference is mainly because of the orientation of the Nadir images, which are all taken straight down. The quality in the viewing direction is worse than in the plane parallel to the image plane, because the intersecting view lines are more parallel in that direction.

Toward the edges, the number of observations for each tie point decrease, and therefore also the standard deviation increases.

For the Z direction, the horizontal striped pattern is very clear visible. This has a slightly different origin than earlier explained. Because the overlap between some strips is from further away, as the widest part of the image is in the overlap direction to other flight lines. This creates overlap regions with bigger intersecting observation angles, which decreases mainly the standard deviation in Z direction. This is shown in Figure 5.6. A, B, C and D are images in different flight lines, where point P is not in the overlap zone, while point Q is.

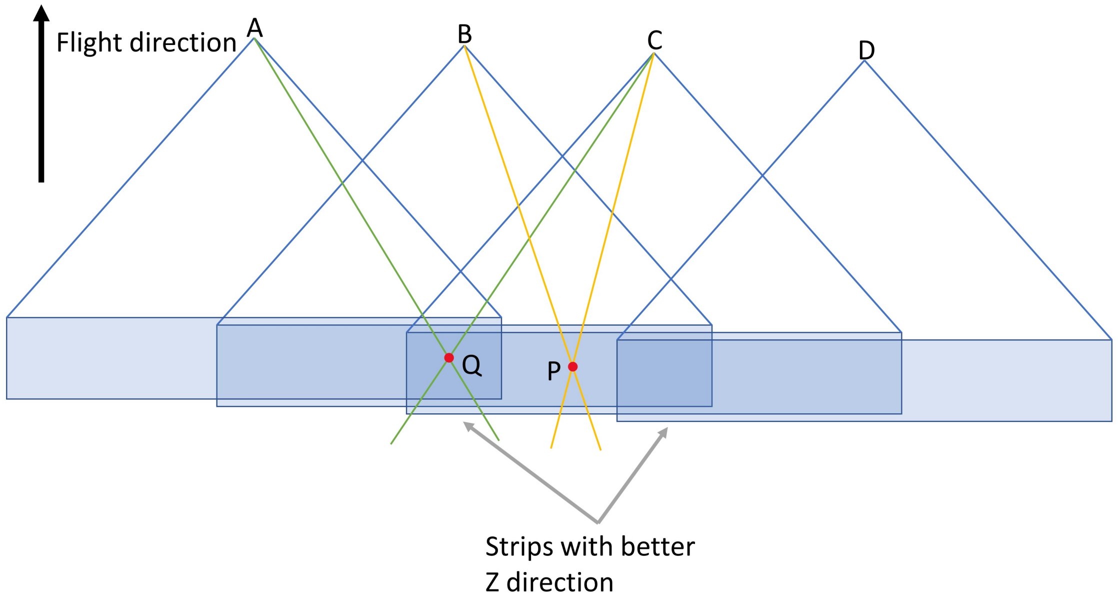


Figure 5.6: Origin of stripe pattern in Nadir Z direction

Oblique

In the oblique image set results, there is a large difference between the X and Y direction on the edges of the area. In general for photogrammetry, when images are taken from the same direction, the quality in a triangulated point is better in the plane parallel to the image plane, and worse in the view line direction. This is why in the X direction, the quality is poorer left and right and in Y direction it is poorer up and down. This is because in those areas, the main source of images is of one direction.

For the Z direction, high standard deviations are seen (red), more than in XY direction, which indicates that the quality is bad. However, in the middle there is an area where the standard deviation is actually very good compared to other points. Values are comparable even to the X and Y direction. This is due to the fact that those points are viewed from different angles, making the viewing lines almost perpendicular and therefore the quality in the adjustment very good. Another part of this is that oblique images looking inward in the area only have a maximal overlap more to the middle of the area, resulting in a smaller area with maximal image overlap. In the outer parts of the set, these different viewing orientations are less likely to occur, the overlap is less and therefore the standard deviations are higher.

Oblique + Nadir

In the combined configuration, the qualities of both image sets are combined. The combination of both image sets leads to the best results in terms of standard deviations, with most points well below 10cm standard deviation. This makes sense, because compared to both individual configurations, only information is added, so the result can only improve.

5.2.2. Tie point subset analysis

To remove any outliers because of less image availability on the edges of the area, a subset is taken from the complete tie point set, in the middle of the area. In this way, the image coverage is maximal in all points and a more thorough comparison can be done between Nadir, Oblique and Nadir + Oblique. The subset extend is shown in Figure 5.7. The subset avoids all areas without available tie points and the area is further away from the borders of the image set.

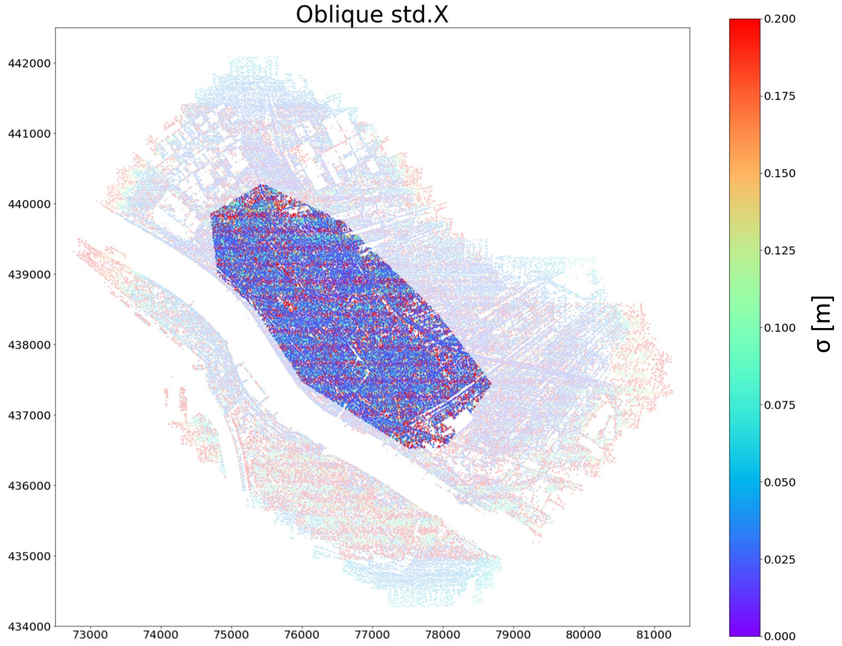


Figure 5.7: Subset taken from total set, to decrease influence outliers on the borders

In Figure 5.8, a histogram of the standard deviations in the subsets is shown and the median values are given in Table 5.2. Similar results appear in these histograms as visible in Figure 5.5, where it is clear that in the separate Nadir and Oblique sets, the results are similar to each other; However, in Z direction there is a clear difference in Z direction where there are better points in the Oblique set than in the Nadir set. In the combined set, the results of the median value improves with a factor 2-3 in all directions, which is significant.

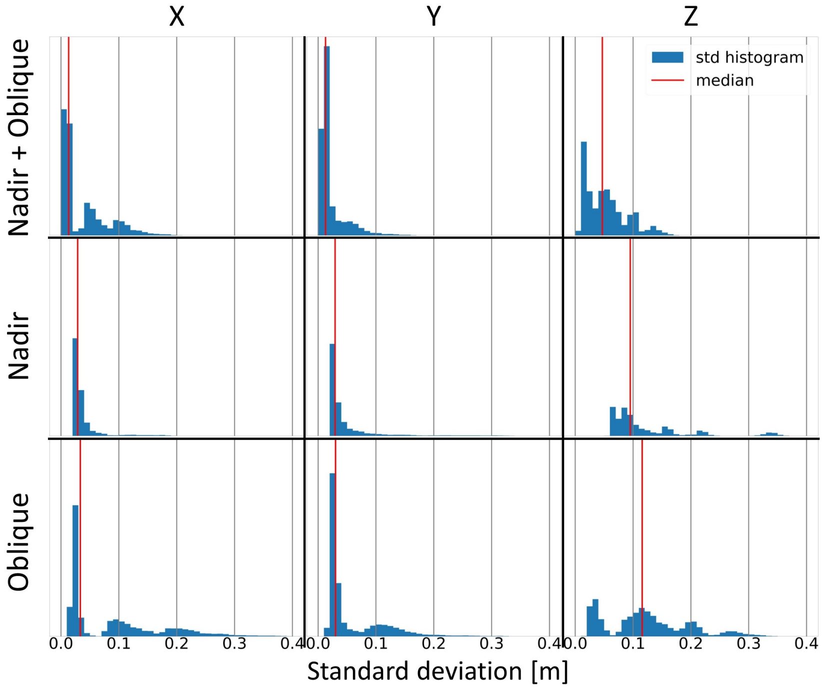


Figure 5.8: Tie point subset standard deviations in a histogram, with the median value in red. Median value in Table 5.2.

	X	Y	Z
Nadir + Oblique	1.31	1.27	4.68
Nadir	2.87	2.93	9.51
Oblique	3.33	3.01	11.57

Table 5.2: Standard deviation median values in cm of Figure 5.8

In Table 5.3, the number of shared tie points in the subset between the Nadir, Oblique and Nadir + Oblique image sets is shown, to prove that significant amounts of tie points are shared. As can be seen, approximately 2/3rd of all points in the Nadir set are also in the Oblique set.

	Nadir + Oblique	Nadir	Oblique
Nadir + Oblique	58940	25686	50855
Nadir		25686	16655
Oblique			50855

Table 5.3: Number of shared tie points between image sets, in the subset

To see if individual tie points improve in different adjustments, the standard deviation of tie points present in both sets is subtracted from each other. This is done as ’Oblique’ minus ’Nadir’, in Figure 5.9, and as ’Nadir + Oblique’ minus ’Oblique’ in Figure 5.10. This means if the result is negative, the tie point has a lower standard deviation and is considered better.

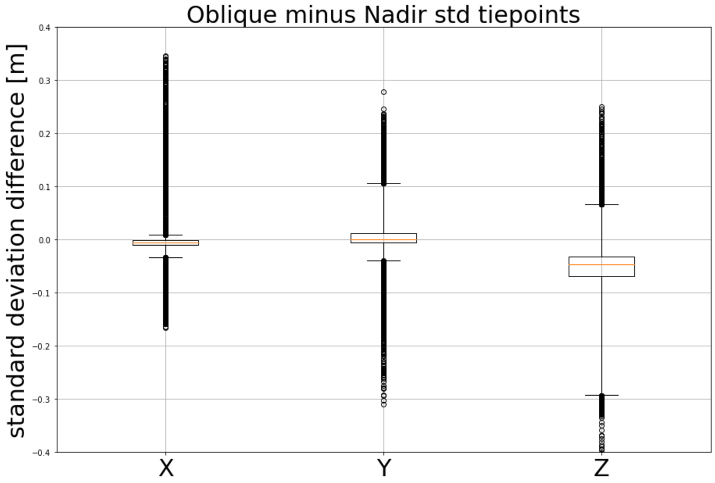


Figure 5.9: Difference in standard deviations between the Oblique and the Nadir image set. In these figures, X Y and Z direction are split in three box plots. 50 percent of the values are within the box, and the orange line is the median value. The whiskers are covering the 5-95 percentile range, and the rest of the values is plotted as outlier. If the value is negative, the standard deviation is lower in the Oblique set.

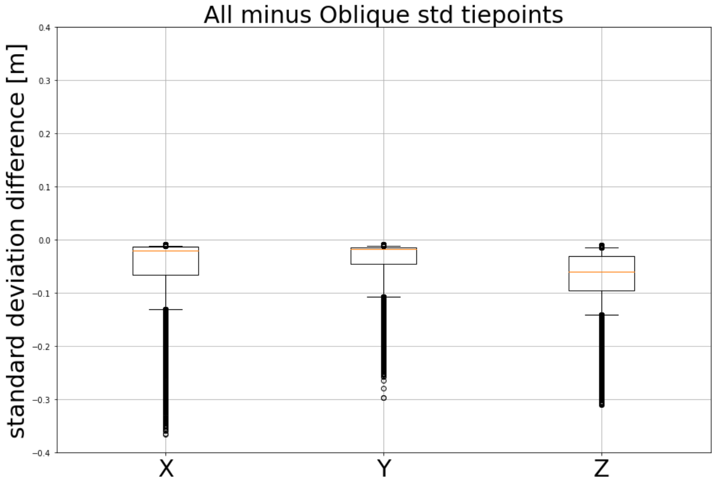


Figure 5.10: Difference in standard deviations between the Nadir + Oblique and the Oblique image set. In these figures, X Y and Z direction are split in three box plots. 50 percent of the values are within the box, and the orange line is the median value. The whiskers are covering the 5-95 percentile range, and the rest of the values is plotted as outlier. If the value is negative, the standard deviation is lower in the Nadir + Oblique set.

In the comparison between the Nadir and the Oblique adjustment in Figure 5.9, it can be seen that the X and Y direction have similar results; 50 percent of the tie points have a difference in standard deviation of less than 0.01 m. There are outliers in both directions, but overall there is no difference in which performs better. In height direction however this changes and the oblique set clearly performs better, with 50 percent of the tie points having a standard deviation improvement between 0.03-0.07m.

If then the Oblique set is compared to Nadir + Oblique in Figure 5.10, there is only improvements when the Nadir set is added. This is not unexpected, there is only information added to the adjustment, which should only improve results. Interesting is that the oblique set does perform better in X and Y direction compared to Z direction, which improves even more compared to Nadir.

5.2.3. Camera exteriors standard deviations

In a similar manner as the tie points, the camera exterior orientations can be visualised. These orientations are also adjusted and come with theoretical standard deviations in X, Y, Z and ω, ϕ, κ. In Figure 5.11, the standard deviations in X, Y, Z are visualised for the three different adjusted image sets. For each camera locations there are 5 image directions, each with their own adjusted exteriors.

Nadir

When the Nadir adjustment is considered, it is clear that the Z direction of the images has a lower standard deviation than X and Y, where Y is slightly better than X. This is in contrast with the results for the tie points, where the Z direction was worse than in horizontal direction.

Also very clear is the influence of the Nieuwe Waterweg, which is water, and in water no features can be found. No tie points means less observations, which results in less accurate results.

Oblique

In the Oblique set, it is also clear that the direction in which the cameras are oriented, the standard deviation is lower. This means that the distance of a camera to the tie points is a more constraint variable, which originates from the fact that this depends on the distance of tie points to each other observed in one image, which is a redundant observation.

Nadir + Oblique

When the image sets are combined, and the adjustment is ran with all images and tie points, the standard deviation improves in all directions and for most of the images, the standard deviation drops to under 3 cm.

		Nadir	Backward	Forward	Left	Right
Nadir + Oblique	X	2.54	2.29	2.27	2.90	2.87
	Y	2.19	2.80	2.79	1.92	1.88
	Z	0.83	3.11	3.08	3.09	3.08
Nadir	X	10.38				
	Y	8.85				
	Z	5.15				
Oblique	X		6.47	6.49	9.69	9.45
	Y		9.75	9.71	5.88	5.66
	Z		9.53	9.49	9.76	9.66

Table 5.4: Average standard deviations for image locations in cm.

A summary of average standard deviations is shown in Table 5.4. The reduction is significant when Nadir and Oblique are combined, and an important conclusion is that by adding oblique images, the quality of the exteriors of the cameras drastically improves. For the Nadir camera locations, this is an decrease of the standard deviation of factor 4-6. For the oblique, this amounts to an decrease of factor 2-4. This is probably less in oblique, because the oblique images are already connected from different orientations and therefore have already a relatively better standard deviation.

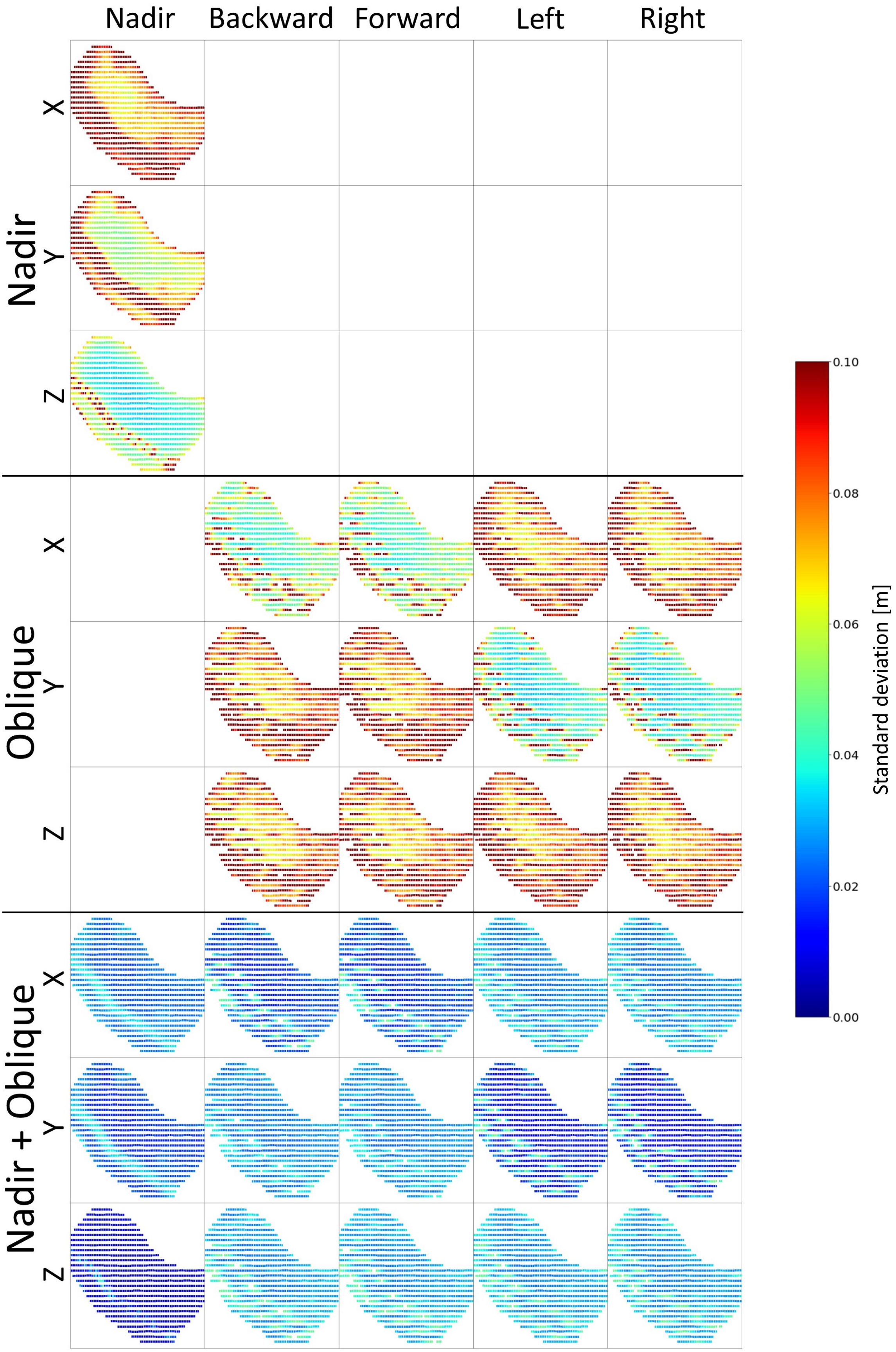


Figure 5.11: Standard deviations in image coordinates, in three different adjusted image sets. Nadir, Oblique and Nadir + Oblique. For each camera position, there is an image in all 5 directions (Nadir, Backward, Forward, Left, Right) with its own adjusted position and standard deviation. Naturally, in the Nadir set, only nadir images are present and in the Oblique set, only oblique images are present.

Image orientations
Not shown here are the standard deviations in the orientation angles. They show similar results, for the Nadir + Oblique combination much better results than in the separate adjustments. This is summarised in Table 5.5. The standard deviations for the separate sets are well under 0.2 milidegree, and for the total set it drops even to 0.03 milidegree. On a distance of 1000m, which is the average distance for oblique images taken at a height of 600m as in this case (chapter 3), a difference of 0.03 milidegree in an image orientation is a displacement of around 3 cm.

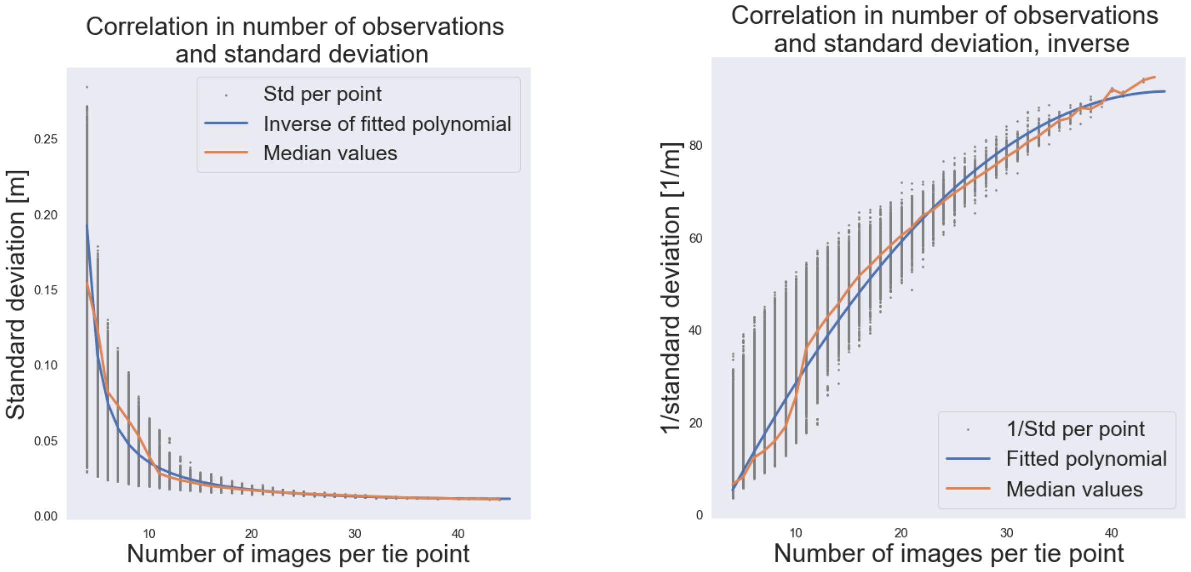
		Nadir	Backward	Forward	Left	Right
Nadir + Oblique	ω	0.0339	0.0448	0.0449	0.0338	0.0334
	ϕ	0.0396	0.0345	0.0346	0.0343	0.0339
	κ	0.0128	0.0338	0.0337	0.0324	0.0317
Nadir	ω	0.134				
	ϕ	0.160				
	κ	0.0402				
Oblique	ω		0.168	0.166	0.113	0.109
	ϕ		0.119	0.120	0.118	0.114
	κ		0.142	0.148	0.125	0.127

Table 5.5: Average standard deviations for image orientations in milidegree.

5.2.4. Quality propagation

The theoretical standard deviation resulting from the bundle adjustment is propagated to the dense match result. For this, the tie point standard deviation discussed in subsection 5.2.2 is used. Because this subset is in the middle of the area, and contains no large water bodies of other features that might significantly influence quality results, and all specific buildings are inside this area it is a good representation of average data.

In Figure 5.12a, the tie point standard deviations are plotted as function of how many image observations they have, with a line trough the median for each observation number. Through these median standard deviations, a polynomial function is plotted to map the number of observations to a quality metric. This is used for the dense matched result in section 5.4 as weight. To create a weight that is higher with a lower standard deviation, the result is inverted, visible in Figure 5.12b.



(a)

(b)

Figure 5.12: Standard deviations of the tie points in X, Y and Z directions, in three image sets: only oblique images (a, b, c), only nadir images (d, e, f), and both (g, h, i). Scale is from 0 to 20cm.

5.3. Dense matching result

For the dense matching result, the results as from the COLMAP dense-matcher are shown in this section. In aerial image sets the camera system is usually of high quality, and the resolution of the images is very high, as described in subsection 3.2.2. This results in that the images had to be reduced in size, leaving them with a maximum resolution of 3000 pixels. This is needed even after the set of images is reduced to only the images that have the selected building in view and the set of images comprised of circa 60 images. In Figure 5.13, the quality drop can be seen.

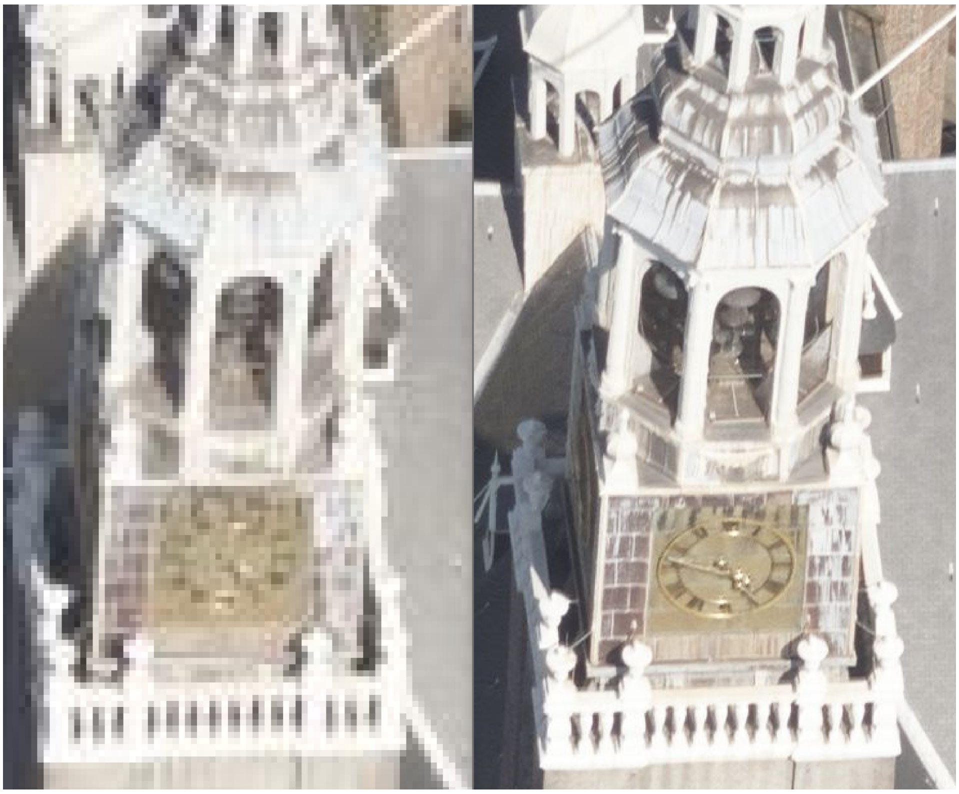


Figure 5.13: Part of an oblique image, after and before a reduction to maximum 3000 pixels. A lot of detail is lost in this reduction.

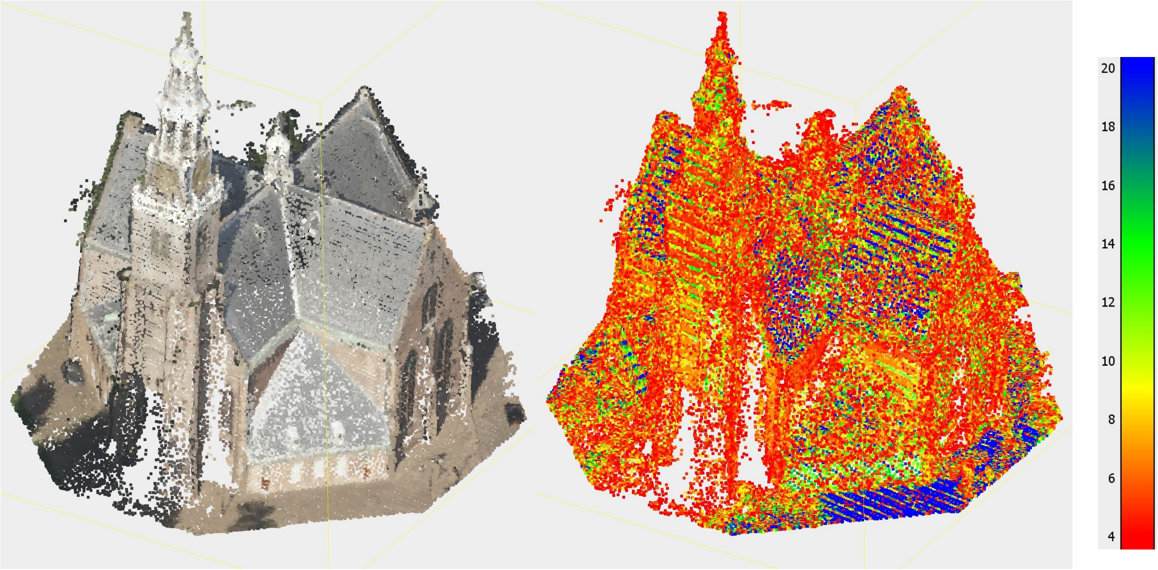
Because the COLMAP reconstruction is not referenced to a world coordinate, the dense match and therefore the dense point clouds of the different buildings are not necessarily in equal scale. This is checked with measuring the buildings in the BGT and in the dense point cloud, of which results are shown in Table 5.6. With this in mind, any distance threshold set for RANSAC is comparable to each building, because the scales do not differ significantly. This is due to the fact that COLMAP uses pixel sizes to scale the dense match, and the images are all similar in that way.

Building	meters/units	scaling factor
Warehouse	95/0.6	158
Flat	55.5/0.35	159
Block	77.5/0.46	164
House	17.5/0.11	159
Church	40.0/0.25	160
School	92.5/0.58	159

Table 5.6: Approximate scaling factors of the dense point cloud to meters

An example of the dense matching output is shown in Figure 5.14, where the RGB coloured point cloud is shown and the point cloud coloured with the number of images a point is visible in. This is extracted from the dense match algorithm in COLMAP, and is used to propagate the quality of the bundle adjustment.

As can be seen, the number of observations is higher in flat areas, particularly those that are visible from straight above, because there the combination of Nadir and oblique images provide the most potential observations. But also in some walls, the number of observations is higher. Where the dense match is less recognisable, the number of observations also decreases.



(a) RGB colors

(b) Number of observations per point

Figure 5.14: Dense matching result of the Church building

5.4. Weighted RANSAC results

For the result of the RANSAC method, the output of the weighted RANSAC is compared to a run with the same parameters but with unweighted RANSAC. This is done by running the same algorithm, but with the weights all set to 1. Also a run is done with filtering of points that are observed 4 times or less, because these are considered as too much of an outlier. In Table 5.7, the parameters used in RANSAC as described in section 4.4 are defined.

Parameters	
d	0.002 (0.32 m)
θ	10^o
Minimum number of points	0.3404%
Iterations	100 or 1000

Table 5.7: RANSAC parameters. The distance parameter is 0.002 in the COLMAP reference system, which translates roughly to 0.32 meters. The normal of a point can differ maximum of θ = 10^o of the candidate plane. The minimum number of points is a percentage of the point cloud size, to keep it normalised to building sizes (this means that with less points in a point cloud, the absolute value of this parameter also decreases). The iteration parameter is changed between the two given values.

5.4.1. Summary of the results

In Table 5.8 and Table 5.9, respectively the amount of planes found for each building in each configuration and the percentage of total point cloud that is classified in a plane is summarised. On the left side of the table are the runs with 100 iterations and on the right, the runs with 1000 iterations. Three weight methods are distinguished, the weighted method, the unweighted method, and the weighted method with points observed in less than 5 images removed.

Iterations	100			1000		
Weighted	Yes	No	Yes - obs>4	Yes	No	Yes - obs>4
Warehouse	16.6	16.9	16.2	25.2	25.7	25.0
Flat	14.9	14.9	15.4	24.7	25.2	24.7
Block	12.2	13.2	13.2	17.1	16.6	20.5
House	19.7	18.4	22.5	29.1	28.2	31.0
Church	10.8	11.0	11.8	25.0	25.2	26.5
School	24.8	25.3	26.1	26.5	27.1	27.5
Average	16.5	16.6	17.5	24.6	24.7	25.9

Table 5.8: Amount of planes found with different methods and iteration numbers, for each building.

Iterations	100			1000		
Weighted	Yes	No	Yes - obs>4	Yes	No	Yes - obs>4
Warehouse	58.2%	58.2%	59.6%	65.4%	65.4%	66.8%
Flat	43.4%	43.7%	47.1%	50.4%	50.2%	53.4%
Block	36.9%	36.2%	40.0%	40.4%	40.0%	45.0%
House	32.6%	31.3%	38.1%	38.8%	38.2%	44.1%
Church	23.2%	22.2%	25.3%	36.1%	36.0%	39.2%
School	42.9%	42.7%	46.9%	44.9%	44.9%	49.0%
Average	39.5%	39.0%	42.8%	46.0%	45.8%	49.6%

Table 5.9: Percentage of all points that is classified in a plane with different methods and iteration numbers, for each building.

In Table 5.8, no significant differences are apparent between the weighted and unweighted methods in terms of the number of planes extracted from the point cloud. The discrepancies, if any, are minute and decimal in nature, with the weighted method occasionally outperforming the unweighted one slightly, and vice versa. Upon excluding the points observed fewer than five times, a marginal increase in the number of extracted planes can be noted, specifically in the Flat, the Block, the Church and the School buildings.

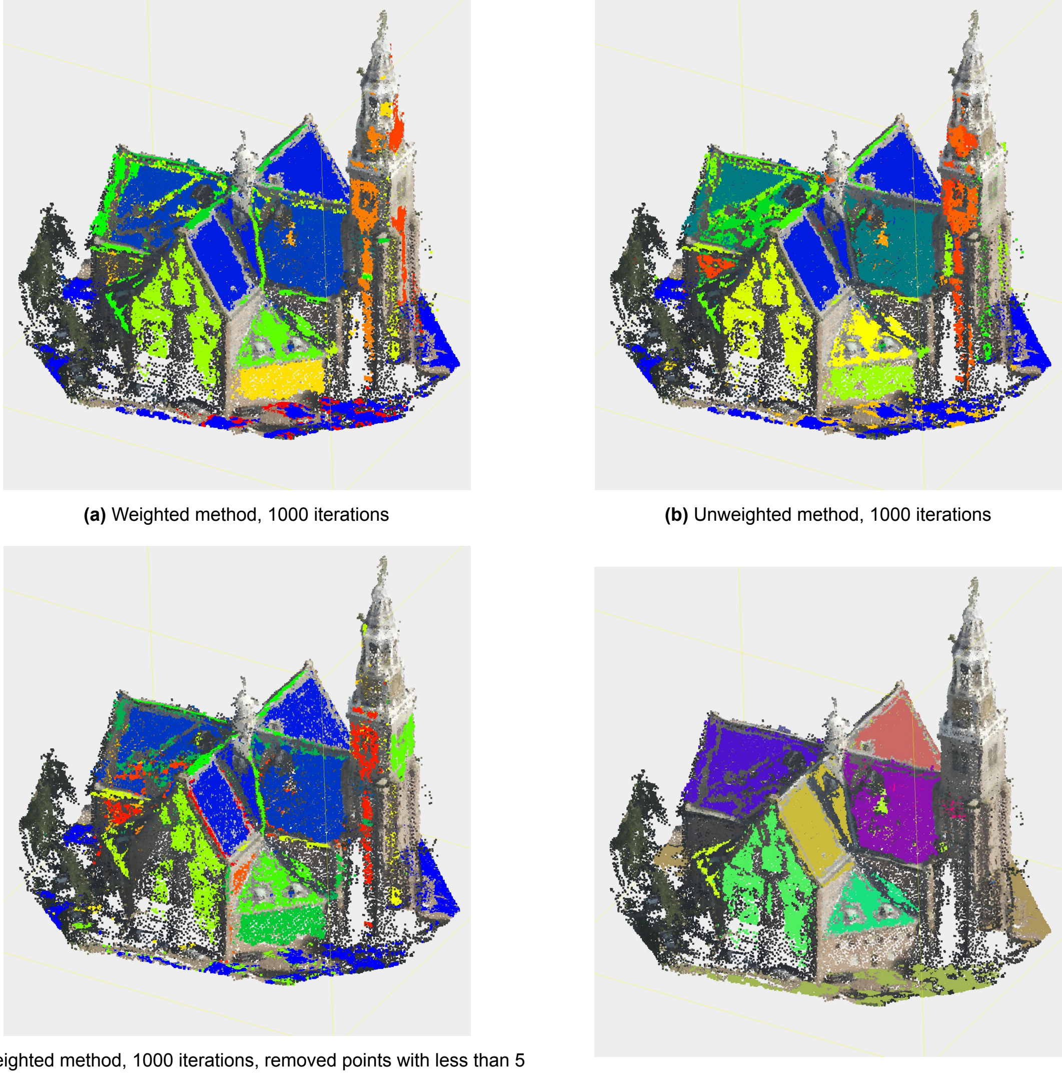
The second table, which reflects the percentage of points classified in planes, does not reveal significant variation between the weighted and unweighted methods either. Interestingly, the method that excludes the least observed points consistently exhibits a higher percentage of points classified in planes across all building types. This is likely due to the fact that points with fewer observations tend to be located in noisier parts of the point cloud, such as trees.

The removal of the points with lowest quality shows an improvement on the amount of planes that is extracted, and also classifies more points as inlier of planes. This is due to the fact that these points are more likely to be an outlier, and therefore not belong in a plane. Secondly, for instance a tree does not have a consistent and recognisable appearance and therefore also has points with less observations, while not a lot of points in a tree will be classified as belonging to a plane. This is also a reason the removal of these points improves the percentage of inliers overall.

5.4.2. Visual inspection of the result

To inspect an average result more closely, in Figure 5.15, an average result of all methods is shown. These are all results with 1000 iterations. In Figure 5.15a, the weighted method is shown, in Figure 5.15b the unweighted method, Figure 5.15c shows the weighted method with worst point removal and in Figure 5.15d as reference a RANSAC result with same parameters done in the open-source software CloudCompare.

It is hard to conclude from these visualisations, as the results are not very different. Differences in found planes are more likely due to the randomness of the RANSAC process, than to the difference in method because the spread in the RANSAC results is quite large. In Figure 5.16, the spread in the 100 runs of the 1000 iterations of the Church building is shown. As can be seen, the spread is significantly larger than the differences between the average results of the different methods. Also the quality of the dense match plays a role here, as the reference model from CloudCompare also has difficulties in recognising planes, especially the smaller planes. Only Church results are shown here, but this is similar for all buildings.



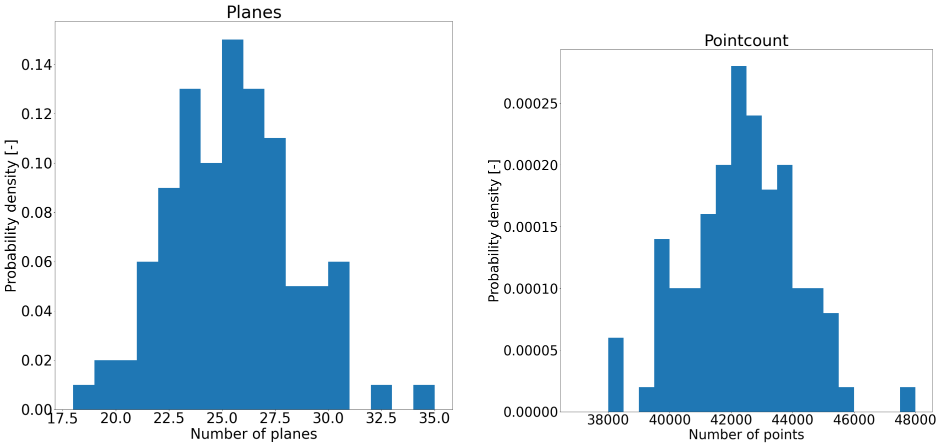
(a) Weighted method, 1000 iterations

(b) Unweighted method, 1000 iterations

(c) Weighted method, 1000 iterations, removed points with less than 5 observations

(d) CloudCompare result with similar settings

Figure 5.15: An average result of RANSAC with the Church building. The different colours represent different extracted planes, and if a point is not classified in a plane it has its RGB value from the dense match.



(a) **(b)**

Figure 5.16: The result of the weighted method with 1000 iterations of the Church building, number of planes and the number of inlaying points generated over a 100 runs, which are shown here in a histogram.

6 Discussion

In this chapter, results from the previous chapter are discussed and further interpreted to be able to answer the research questions in the conclusion. First, in section 6.1, the results of the bundle adjustment are discussed. After that, the dense matching is discussed in section 6.2 and finally, the 3D feature extraction is analysed and results are further explained in section 6.3.

6.1. Quality in Photogrammetry

In the results of the bundle adjustment, three main reasons that influence the theoretical quality can be identified.

- Availability of tie points
- Viewing direction
- Intersection angle of observation lines

When there are less tie points available in any form (image set borders, water bodies, forested areas) the quality overall decreases. This is the most straightforward factor, because if you have less observations, the redundancy in the solution is lower and standard deviations rise. The second large influence on the theoretical quality is the viewing direction, and the matching between those different directions. For the tie points, the quality in viewing direction is different than in the direction perpendicular to the viewing direction, so in the image plane. This is because it is estimated with viewing lines, and a small deviation in a viewing line angle results in a small deviation in image plane location, but can be large in distance direction. The intersection angle of viewing lines of an observed tie point is the third large influential parameter. It is similar to the previous one, but this is for instance visible in the standard deviation in the Z direction in the Nadir adjustment. In this result, there are horizontal stripes visible, that are due tho the overlap. Because the images from two flight lines further away overlap, the intersection angle of the viewing lines is larger and the standard deviation in Z direction improves in the area that has this overlap between far away nadir images. This has the same technical cause as the previous. In the other direction, this is not visible because the overlap is much higher and the images are smaller in that direction, creating no overlap between far away images.

6.1.1. Oblique versus Nadir

Oblique images where expected to increase the quality in certain aspects of the image set. Where Nadir has proven good results in horizontal directions, already known was the worse quality in height direction. This both shows also in the results of the adjustment of only Nadir images in this thesis. Oblique images on the other hand show more outliers, but also more points that have increased theoretical quality. And as expected, the quality for tie points in height direction indeed is significantly better than for Nadir images. The reason for the outliers could be that the distance to the observed feature is larger and ground sampling distance becomes larger, and the observation less accurate.

In the camera exteriors, the opposite is happening. Height direction is better in the Nadir adjustment, while horizontal directions are better in the Oblique adjustment. This is because the height in Nadir images is related to scale, and with a lot of observations in an image with good horizontal direction, the scale is good and therefore the distance to the tie points is of higher quality. In oblique, this is also more significant in the viewing direction for the same reasons, and therefore this results in better results in the horizontal directions.

However, if the sets are adjusted together, they combine the advantages of each individual set and improve the quality even more. Also in the camera exteriors, the improvement of the standard deviation is significant. This is achieved because the Nadir images tie the different image orientations together, and create a more redundant system.

6.1.2. Camera exteriors

In this bundle adjustment, also the cameras are adjusted as separate cameras that can 'move' with respect to one another. In reality however these cameras are attached to each other, and they have no (or very little) room to move. This also means that they have similar exteriors, and similar standard deviations, and should be adjusted as one unknown. This will influence the quality of the observations. In the results, already is very clear that tying Nadir and Oblique images together improve results significantly, and using this connection between orientations it might improve even more.

6.2. Dense matching

For the purposes needed in this thesis, the dense matching process was a challenging step. Because aerial images are of very high resolution, dense matching on these image sets is a challenge. It can be questioned if dense matching is the way to go for reconstruction purposes of objects, because in the dense matching process, a lot of information can be lost or generated as noise. There are other ways to extract information from images, directly recognising features in images such as corners, edges or other techniques. This is a more direct way to process images, and in that way less steps that can cause noise or errors are needed.

Because of the high resolution, reduction of images of the dense matching is needed an may influence the dense match result quite a bit, because the image in this case gets reduced to 1/6th of its pixel size. This influences the results of the dense match of course, but also the quality of a point may not be directly transferable from the tie points to the dense points. This is one of the larger assumptions in this thesis.

6.3. 3D model reconstruction/3D feature extraction

One of the first goals of this thesis was to reconstruct a 3D model, using the polygon fitting algorithm proposed by Nan and Wonka [11] and incorporate the quality of the points in the 3D reconstruction. A comparison between reconstruction with or without the quality taken into account should provide an answer to the question if quality could be of importance in these 3D reconstructions. However, due to the results of the dense match this was harder than expected, and chosen was to completely left this out of the methodology. Instead was chosen to compare results of a weighted or unweighted RANSAC method.

Two main characteristics of the results where used as an assessment criteria: the amount of planes generated, and the ratio of points classified as inliers or outliers. In these criteria, no significant differences are found between a weighted RANSAC and an unweighted RANSAC. The parameters which are used in RANSAC where kept strict, to be able to identify better if there was a difference in the methods. The result of the RANSAC has a high variability, questioning the quality of the implementation of the RANSAC method in general. To eliminate this variability, the runs where done 100 times and an average was used to analyse the results.

Removing points with a lower quality entirely however seemed to improve the results. While this looks as a significant improvement, it is probably not due to the weighing of the points directly. The quality is taken into account, but only as a threshold to remove certain points, and the weighing of the points in the method is not the largest contributor to the improvement.

7 Conclusions and recommendations

7.1. Conclusions

The main research question in the beginning of this thesis was: ”What is the quality of oblique aerial imagery compared to nadir imagery for photogrammetry purposes, and how can this quality be incorporated in a 3D reconstruction of buildings?” To answer this question, sub-questions where formulated and these will now be answered.

How can the theoretical quality of a point coordinate in oblique images be found?

Using observed features in images, these images are matched together. A bundle adjustment is done to adjust all unknown parameters, and using the covariance matrix of the observations and the system of equations, the covariance matrix of the unknowns can be calculated. This involves a very computationally intensive inversion, because the system of equations and the number of observations can become very large in aerial photogrammetry.

What is the difference in quality between the oblique image observations and nadir image observations?

There is a large difference in quality between Nadir and Oblique image adjustments, which is mainly due to the different orientation of the images. This gives higher accuracy in different directions, and depending on the use or need a choice can be made. For this image set, the standard deviations of the tie points in horizontal direction are in the order of magnitude of around 3 cm in representative areas, where in the vertical direction it is around 10 cm. However, if possible, a combination is always preferred, because the combination is better than both sets separately and it improves to 1 cm in horizontal and 5 in vertical direction.

What are the characteristics of the observation parameters and surroundings that have influence on the quality of the results?

For the bundle adjustment, there are three main characteristics influencing the results. First, tie point availability is important; areas that have less tie points available show bad results. Because more observations create a more redundant system of equations, the theoretical quality will be improved by more observations. Secondly, the viewing direction influences the results. For tie points, the quality is better parallel to the image plane, and for similar reasons the quality for camera exteriors is better in perpendicular direction with respect to the image plane. And lastly, the availability of observation angle differences improves the result as well. When the intersection angle of the observation lines is higher, the quality improves.

For 3D geometric feature extraction, the improvement is not visible for the method. If the points with lowest quality are removed, an improvement is seen, but this is less due to the weighing of the points in the RANSAC method and it is more some sort of outlier removal. A big influence on the RANSAC method in general is the dense match result, which is losing much information in the down-sampling of the high resolution aerial images.

How can the quality of points be weighted in a plane fitting algorithm?

Using the number of observations of a tie point, the number of observations of each point in a dense matched point cloud is assigned a quality value. Normally, RANSAC chooses points randomly to generate geometric features, but in this thesis RANSAC is adapted and chooses points now weighted randomly. Also candidate features are fitted with a weighted least squares on inliers, giving better points more influence.

Does the RANSAC method improve performance by incorporating the quality as a weighting metric?

The implemented RANSAC method to generate geometric features is assessed on the amount of planes it can extract from the point cloud, and the percentage of points that is classified as inlier of a plane. This is a good metric, if the parameters are kept very strict. A better performing method will find more planes and classify more points as inlier of a plane under more tight parameters, while needing less iterations in the process. The weighted method performed on 100 averaged runs not better on extracting planes nor percentage of inliers, and also in showed no difference in performance with fewer iterations. When the points where filtered based on a threshold of the quality metric, an improvement was visible but this is not necessarily due to the weighting method but more to the removal of outliers.

As final conclusion and answer to the main research question: Oblique images in itself have, aside to an improvement of quality in a different direction no special characteristics, but combining them with Nadir images the improvement in quality is significant, up to a factor of three, because the Nadir images act as a connection between the Obliques. Unfortunately, using this quality in the extraction of geometric features by means of a weighted RANSAC does not increase performance significantly.

7.2. Recommendations

A few recommendations for further research or use of applications is mentioned below.

One of the biggest recommendations would be to connect the camera exteriors of the different camera orientations together in the adjustment, and see if it is possible to adjust the observations in this way. This is a more realistic scenario, as this is the actual situation in which the observations are done. Another source of improvement could also be to try to find a method to extract features that can be matched to images under a larger angle (for instance Backward and Left orientations).

Another recommendation could be to find a way to dense match the images with higher resolution. This should help the RANSAC method a lot in general, and it can be interesting to see if the proposed weighting method works better then. Also the 3D model reconstruction will work better if the dense matched point cloud is more consistent and detailed. With more overlap and the inclusion of oblique images, these image sets will have a lot of overlapping images, and the dense match algorithms will contain too much computational storage space compared to images from one side with less images with overlap.

Furthermore, recommended is to implement this method with a slightly more sophisticated RANSAC method. In this thesis, a simple version of RANSAC is modified, but for example the detected planes are not separated if in one planar feature two sets of points clearly are two different point clouds and therefore probably belong to different planes. This could make a significant improvement in the RANSAC method, and

As a last remark, due to the high demands of dense matching, it might be more profitable to find other ways to reconstruct 3D models altogether. This is also more direct use of image information.

References

[1] Actueel Hoogtebestand Nederland. *AHN*. 2022. URL: `https://www.ahn.nl/` (visited on 07/26/2022).

[2] Martin A. Fischler and Robert C. Bolles. "Random sample consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography". In: *Communications of the ACM* 24 (6 1981). ISSN: 15577317. DOI: `10.1145/358669.358692`.

[3] Wolfgang Förstner and Bernhard P Wrobel. *Photogrammetric Computer Vision*. Vol. 64. 1970.

[4] Yasutaka Furukawa and J. Ponce. "Accurate, Dense, and Robust Multiview Stereopsis". In: *IEEE transactions on pattern analysis and machine intelligence* 32 (Aug. 2010), pp. 1362–76. DOI: `10.1109/TPAMI.2009.161`.

[5] Geogap. *BGT viewer*. 2022. URL: `https://bgtviewer.nl/` (visited on 07/26/2022).

[6] Geonovum. *Basisregistratie Grootschalige Topografie Gegegevenscatalogus BGT 1.2*. 2020. URL: `https://docs.geostandaarden.nl/imgeo/catalogus/bgt/` (visited on 07/26/2022).

[7] S. I. Granshaw. "BUNDLE ADJUSTMENT METHODS IN ENGINEERING PHOTOGRAMMETRY". In: *The Photogrammetric Record* 10 (56 1980). ISSN: 14779730. DOI: `10.1111/j.1477-9730.1980.tb00020.x`.

[8] Abdallah M. Khalil. "Two-dimensional displacement measurement using static close range photogrammetry and a single fixed camera". In: *Alexandria Engineering Journal* 50 (3 2011). ISSN: 11100168. DOI: `10.1016/j.aej.2011.07.003`.

[9] Karl Kraus. *Photogrammetry Geometry from Images and Laser Scans*. 2011.

[10] David G Lowe. "Distinctive Image Features from Scale-invariant Keypoints". In: *International Journal of Computer Vision* 60 (2 2004). ISSN: 0920-5691. DOI: `10.1023/B:VISI.0000029664.99615.94`.

[11] Liangliang Nan and Peter Wonka. "PolyFit: Polygonal Surface Reconstruction from Point Clouds". In: vol. 2017-October. 2017. DOI: `10.1109/ICCV.2017.258`.

[12] OpenStreetMap contributors. *Planet dump retrieved from https://planet.osm.org*. `https://www.openstreetmap.org`. 2017.

[13] Ravi Peters et al. *Automated 3D reconstruction of LoD2 and LoD1 models for all 10 million buildings of the Netherlands*. English. 2022. DOI: `10.14358/PERS.21-00032R2`.

[14] R. Schnabel, R. Wahl, and R. Klein. "Efficient RANSAC for point-cloud shape detection". In: *Computer Graphics Forum* 26 (2 2007). ISSN: 14678659. DOI: `10.1111/j.1467-8659.2007.01016.x`.

[15] Johannes Lutz Schönberger and Jan-Michael Frahm. "Structure-from-Motion Revisited". In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016.

[16] P. Teunissen. *Adjustment Theory: an introduction*. Dec. 2000. ISBN: 9040719748.

[17] P.J.G. Teunissen et al. *Probability and Observation Theory: an Introduction*. TU Delft, 2006. URL: `https://books.google.nl/books?id=ba-SnQEACAAJ`.

[18] Bill Triggs et al. "Bundle adjustment – a modern synthesis". In: vol. 1883. 2000. DOI: `10.1007/3-540-44480-7_21`.