# Oblique Aerial Adjustment Through the Creation of Synthetic Test Cases Graduation Plan

Walter Kahn

Student #5658543

1st supervisor: Edward Verbree
2nd supervisor: Martijn Meijers
External supervisor: Annemieke Verbraeck

Delft University of Technology
January 15, 2025

# Acronyms

**BGT** Basisregristratie Grootschalige Topografie

**DISK** DIScrete Keypoints

**GBKN** Grootschalige Basiskaart van Nederland

**GCP** Ground Control Points

**GNSS** Global Navigational Satellite System

**GSD** Ground Sampling Distance

**HDR** High Dynamic Range

**HTW** Handboek Technische Werkzaamheden

**IMU** Inertial Motion Unit

**LiDAR** LIght Detection And Ranging of Laser Imaging Detection And Ranging

**LightGlue** Local Feature Matching at Light Speed

**LSA** Least Square Adjustment

**MLS** Mobile Laser Scanning

**MoSCoW** Must have Should have Could have Won't have

**obj** Object File

**SFM** Structure From Motion

**SIFT** Scale-Invariant Feature Transformation

**TLS** Terrestrial Laser Scanning

# 1 Introduction

Adjustment theory is a statistical process to detect and remove stochastic errors from measurements. This is being applied to aerial nadir imagery to help in the reduction of outside surveying for the mapping of the Basisregristratie Grootschalige Topografie (BGT). Unlike nadir images, currently obliques are not adjusted due to difficulties that stem in the trustworthiness of the control points in the image because of a varying pixel scale when compared to top down nadir as well as occlusion that occurs due to different camera positions. However, oblique images do provide extra data opportunities that would otherwise go lost for municipalities that want to map the BGT or maybe even create a digital twin. For this it is essential to keep the structure of the image the same during the adjustment process and thus not to morph it or overlay on any surface. Due to the strict precision that the Kadaster holds to measurements in the BGT a raw unadjusted oblique image simply does not meet the standards laid out in the Handboek Technische Werkzaamheden (HTW) for the creation of the BGT. This leads us to the goal of the thesis which is to create a series of mathematically defined synthetic test cases that show the impact of key parameters as they relate to adjustment theory. A possible result of this could be that the key points are too inaccurate or that obscuring of objects in different angles leads to a bad fit. The lessons learned from such test cases could then be applied to real data to see whether they hold up in real data scenarios.

## 1.1 Problem Statement

As stated before, the images need to be properly positioned and not morphed for them to be measurable. By positioning, it is meant that parameters such as the scale, rotation, and *xyz* placement of the camera sensor center must be determined in world coordinates. In a perfect environment, the Global Navigational Satellite System (GNSS) and Inertial Motion Unit (IMU) present in an airplane would perfectly capture the location of the plane at the time of recording. However, due to the stochastic nature of measurements, which also affects GNSS, these parameters are never perfect. As a result, the set of images does not align at key points, meaning they cannot be used for measurement in a non-adjusted manner. This issue can be addressed through Least Square Adjustment (LSA), which, in its most basic form, takes a set of stochastic observations for each image in the form of keypoints and calculates the best fit for the camera parameters. However, even after adjustment, the positions of these observations will still not be perfect. Given the bounds of the required accuracy needed for the data's purpose, they can be considered sufficient for the task at hand. For example, the BGT requires a measured accuracy of 5 cm [12].

To further complicate matters, images are not merely dependent on the GNSS/IMU combination present during capture but also on the internal camera model, which has yearly calibrated values but is also subject to stochastic errors. These errors can also be adjusted using bundle adjustment. However, this requires a higher number of observations to solve the adjustment.

This process of adjustment has been successfully applied to nadir images (Meaning images that are taken straight down) for years. This is achieved by identifying keypoints that exist between images and using them to reconstruct the 3D positions of the cameras through a process called Structure From Motion (SFM). In the case of nadir images, these keypoints are detected at sub-pixel accuracy using Scale-Invariant Feature Transformation (SIFT), which employs a variety of classical image processing methods to identify points of interest visible in multiple images, allowing for their 3D reconstruction and the connection with terrestrially captured ground points called Ground Control Points (GCP) which allow for the correct scale to be applied to the images which is instrumental in the measuring process.

Due to the intrinsic differences in oblique images (a set of four images taken between 40 en 50 degrees in the four cardinal directions of the airplanes), the use of SIFT has been shown to be ineffective in case of oblique images, as SIFT is orientation invariant.SIFT does however suffer from occlusion when looking at the same point from a different angle. To address this, recent advancements in machine learning techniques, such as DIScrete Keypoints (DISK) and Local Feature Matching at Light Speed (LightGlue), have created opportunities to better connect images. This results in higher accuracy adjustments for oblique images compared to using SIFT at the cost of sub-pixel keypoint selection, making the adjusted oblique image set more suitable for measurement purposes as described in the introduction.

However, the problem arises that, due to the relative novelty of oblique adjustment using DISK and LightGlue, a definitive quality metric is not yet known and can not ve assigned to such adjusted sets. As such, this thesis explores a method through which such quality can be described via the creation of synthetic tests. These synthetic tests are based on 3D graphical engines that employ readily available 3D geo data to take rendered images of city scapes. These synthetic tests would include the creation of a “perfect” or ground-truth data set which simulates ideal reconstruction through the non gaussian effected observations and compare them to purposefully altered situation that simulate common occurrences. Another disadvantage of using real life data is the impossibility to isolate such occurrences which can be prevented by the creation of synthetic test cases that only show isolated occurrences.

## 1.2 Research Goal and Question

Based on the problem statement a possible goal of the research could be to alleviate the problem that currently faces image adjustment which is the inability to test which parameters impact the adjustment divided for nadir and oblique images. The result of this would be the improvement in the trustworthiness and allow better measurement accuracy. Another goal that can be defined is the creation of a set of test cases upon which both keypoint detection algorithms but most importantly adjustment algorithms can be tested. As it would create small image sets that can be used as test cases with predictable input and experiment with edge cases. A list of goals as it relates to the rest of the research is given in Chapter 4.

**The main research question is as follows**
*What parameters are key in the reconstruction accuracy of adjustment theory applied to rgb aerial images using synthetic test cases?*

The aim of the study is to research the impact of the key parameters present in adjustment theory, specifically focused on their use in bundle adjustment[29]. Normally, a user would not be able to test these different effects on a consistent scale due to the costs associated with aerial image collection. For this, the creation of a synthetic test datasets with the use of 3D computer graphics rendering software might offer a solution by combining 3D computer graphics with readily available geo data. To achieve this proposition, as outlined by the main research question, a number of sub-questions need to be formulated. Later in this chapter, the scope is further defined through the creation of a Must have Should have Could have Won’t have (MoSCoW).

1. What is the mathematical pipeline for aerial adjustment theory?
2. What hyperparameters are present in aerial adjustment, and what are their effects when changed in synthetic tests?
3. What are the main differences between nadir and oblique image adjustment? How do the different parameters of oblique imagery influence the errors present in the final adjusted images.
4. How does the choice of feature extraction algorithm influence the reconstruction accuracy?
5. How do external factors affect adjustment?

## 1.3 Scope

The final Section of the introduction Chapter is the scope which explains the goals and limits the research based on the main question.

**Area**
The research will focus on the Dutch city of Rotterdam, which has a well-developed 3D model and aerial image data. The 3D city model is accessible [20]. The image data is however not publicly accessible but has been made available by MiraMap and the municipality of Rotterdam for the completion of this research. More about the types of data in Chapter 3 Methodology.

**Software**
The research also requires a set of software tools, some of which already exist; in these cases, off-the-shelf versions will be used. If software, such as the 3D rendering tool, is deemed insufficient for the research purpose, a custom version needs to be developed. More about the software, its usage, and its possible creation is detailed in Chapter 3 Methodology.

**MoSCoW (Must have, Should have, Could have, and Won't have)**

1. Must have
   - Provide a general overview of the aerial image collection pipeline using literature.
   - Give a detailed description of the data collection, keypoint extraction/matching, adjustment, and result validation steps in the pipeline.
   - Develop a 3D city renderer capable of generating synthetic oblique images by loading 3D building models in Object File (obj) format, adjusting the camera model, a DTM, and a BGT base layer.
   - Create a "perfect" set of images/positions to verify results.
   - Provide an in-depth overview of the mathematical adjustment model and its dependent parameters.
   - Create a set of synthetic tests differing from the "perfect" set to explore the independent effects of various parameters.
2. Should have
   - Use image processing techniques to introduce noise into the images to simulate data quality issues beyond intrinsic parameters.
   - Create synthetic test data for all parameters to explore their independent effects.
   - Use the actual flight plan to accurately recreate real-life data.
   - Allow for loading actual camera systems to recreate real-life data.
   - Enhance the 3D city renderer with advanced rendering techniques such as High Dynamic Range (HDR) and the inclusion of the sun's position and shadows.
3. Could have
   - Perform synthetic tests with two parameters simultaneously to analyze dependencies.
   - Verify results using actual data collected by the municipality of Rotterdam for the year 2024.
   - Conduct similar tests in a different area.
   - Improve the realism of the 3D city renderer by implementing keypoint extraction challenges, such as water, clouds, trees, and street furniture.
4. Won't have
   - Develop an adjustment algorithm, instead using the adjustment core of GeoDelta.
   - Develop a keypoint detection algorithm, instead using SIFT and DISK.
   - Develop a keypoint matching algorithm, instead using SIFT and LightGlue.
   - Use the 3D rendering software to create mobile panoramic images for similar adjustment tests.
   - Implement time-shift vectors as outlined in ValidDefo.

The goal of the MoSCoW is to outline the scope of the research proposed in this project. A critical aspect is the inclusion of items that will not be addressed during the research to define the limits of the thesis required to conclude an MSc in Geomatics. The list also serves as a starting point for future research by highlighting areas that could further improve the methodology and outcomes, albeit beyond the current time constraints.

The first excluded element is the development of an aerial adjustment algorithm. This is deemed unnecessary for this research as it would introduce numerous variables and optimization challenges, leading to potentially non-representative results. For this study, the GeoDelta adjustment core was chosen. This core is loosely based on [32] ,[33], [34], [11], [37], and[29], incorporating the Delft method of adjustment. An open-source alternative, such as https://github.com/applied-geodesy/bundle-adjustment, commonly used in software like JAG3D (classified as the German approach), could also be considered.

The second and third exclusions involve the development or optimization of keypoint detection and matching algorithms. Techniques like SIFT have existed for a considerable time, with many variants created to overcome its limitations, as discussed in the next chapter. Neural network-based successors, such as DISK [35] and LightGlue [15], have also emerged, spawning their own variants to address specific shortcomings. Preliminary testing at GeoDelta has shown the combination of these existing methods to be satisfactory for detection and matching keypoints. Thus, an off-the-shelf approach was deemed sufficient for this study. Time might be spent on optimizing them but no new method will be developed.

The fourth excluded aspect is the adjustment of terrestrial-based panoramic images collected for the coloring of Mobile Laser Scanning (MLS) or Terrestrial Laser Scanning (TLS). While these methods also utilize DISK and LightGlue, they require modifications to the bundle adjustment algorithm. Additionally, this approach demands higher surface-level detail, as opposed to the lower detail used in aerial imagery due to distance of image. Given the increased time and complexity, this research focuses solely on aerial imagery.
The fifth and final limitation is the exclusion of shift vectors as described in [36]. While promising, this method does not align well with the research goal, which is to validate current oblique adjustment methods using synthetic tests. As shift vectors are not widely used in practice, their inclusion is deemed outside the scope of this study.

## 1.4 Summary

**Summary**
This research addresses the challenges of applying adjustment theory to oblique aerial imagery for improving the mapping accuracy of the Dutch BGT. While nadir images (taken directly downward) have been successfully adjusted using bundle adjustment techniques, oblique images face difficulties due to varying pixel scales, occlusions, and the novelty of applicable algorithms. These issues limit their use despite their potential to provide valuable supplementary data for applications like creating digital twins or detailed municipal maps.

The research proposes the creation of synthetic test cases using 3D computer graphics and geo-data to study the impact of key parameters on adjustment theory. These tests will allow controlled experimentation with isolated variables, overcoming the limitations of real-world data, which is often subject to uncontrollable external factors. The study focuses on exploring the mathematical pipeline of aerial adjustment theory, identifying influential parameters, and testing their effects on reconstruction accuracy in both nadir and oblique images.

**Research Goal**
To evaluate the impact of key parameters in adjustment theory on the reconstruction accuracy of oblique aerial images using synthetic test cases, thereby improving the precision and trustworthiness of oblique image adjustments for high-accuracy mapping applications.

**Cause**
Oblique aerial images provide additional valuable data for mapping and urban planning. However, due to occlusion, varying pixel scales, and inaccuracies in camera parameter estimation, they cannot meet the high precision requirements set for applications like the BGT.

**Problem**
Existing adjustment techniques are insufficiently tested for oblique images due to the lack of controlled environments and suitable quality metrics. This hampers their application in scenarios requiring high accuracy, such as municipal mapping and creating digital twins.

# 2 Related Work

The second chapter of the P2 graduation plan is the related work based on the literature review that occurred during the second quarter of the 20242025 educational year. The literature lays the foundation for the rest of the research. It looks into the aerial collection pipeline specified to the key points, adjustment, and image parameters.

## 2.1 Aerial Oblique and Nadir Images

The collection, as mentioned before, works through the process of tendering; roughly five companies are involved in such processes in the Netherlands. The collection is done on a yearly basis, mostly during the spring, to allow for ample daylight and a reduction in the occlusion caused by vegetation. These collection campaigns are often also used for the collection of point clouds and infrared imagery. These extra collection methods do not get adjusted using bundle adjustment and, as such, fall outside the scope of the research. The data collection companies use aerial vehicles such as planes due to the size and weight of the camera, as well as the distance and height at which the plane needs to fly. The collection occurs for a variety of airborne methods such as LIght Detection And Ranging of Laser Imaging Detection And Ranging (LiDAR), infrared, and oblique and nadir images. The collection occurs through either a plane or a drone. In the case of the research at hand, only oblique and nadir images from a plane are considered. A nadir image is an image that is taken in a straight-down manner, meaning that the image plane is horizontal with the surface plane. These nadir images have a variety of use cases such as mapping of building footprints [22] and [38]. Another possible use case is in the detection of vegetation [27]. These concepts are condensed in the Netherlands in the form of the maintenance of the various base registrations, of which nadir is most prevalently used. This usage is somewhat described in [11], where the maintenance of Grootschalige Basiskaart van Nederland (GBKN) is described. This is an older version at which point it was not yet referred to as a base registration but works similarly to the BGT. Nadir images are used to perform change detection on a city scale [11].

In the case of normal oblique images, which are taken at a 35 to 55-degree angle, a different set of use cases can be defined. The main advantage of oblique images comes from the angle at which the image is taken, allowing the viewer to see the sides of buildings. Where nadir offers an almost 2D view and can thus be rarely used for 3D, an oblique image allows for preservation activities as described in [9]. It also finds its uses in the creation and extraction of 3D city models as described in [8]. Even though the BGT is not a 3D model as of writing, certain municipalities have gone ahead and updated their BGT to also include z-coordinates for certain objects. In the case of this research, oblique images are most interesting for the fact that their adjustment is difficult to ascertain the correctness of. An example where it has even been applied to land management systems such as BGT was given in [14], which describes aerial oblique adjustment as being measurable. However, when compared to nadir images, it shows an accuracy of 90 cm. Modern results show better outcomes [8].

## 2.2 Image Parameters

Bundle adjustment uses observations to create enough degrees of freedom to solve a linear system. More about the linear system is discussed in Section 2.6. In most cases, an aerial capture system consists of 5 cameras: one in the nadir position pointing straight down and four that point at a roughly 45-degree angle, thus capturing a larger surface area as described in [10]. This surface area can be described using the Ground Sampling Distance (GSD), which roughly describes the surface area that a single pixel represents. Figure 1 gives an overview of a simple camera model. This surface area differs based on the incident angle of the zenith and camera angle, with the smallest GSD being in the so-called oblique center, which does not necessarily correspond with the image center. The paper by [10], which gives an overview of the image parameters, also mentions the existence of high obliques, which are images taken at more than 55°. This results in the image also capturing part of the sky. The usage for this is mostly in drones, which fly lower to increase the amount of overlap while decreasing the number of images that need to be taken. It lists a number of effects that can be added to the camera model matrix to simulate various occurrences.

A camera matrix, shown in Equation 1, is a linear model that maps 3D image coordinates to 2D and can be inverted to calculate 2D to 3D.

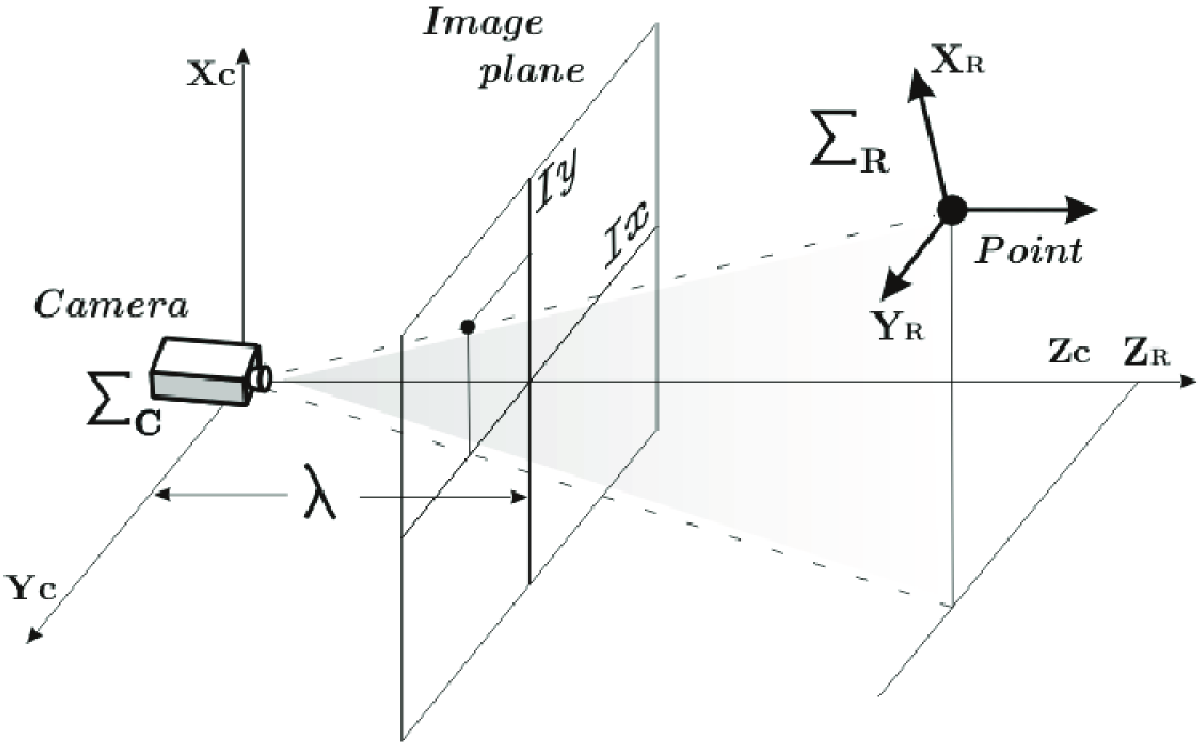



Figure 1: Simple Camera Model

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} W/f & 0 & c_x + ppa_x \\ 0 & H/f & c_y + ppa_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{1}$$

This includes the camera rotation, which is marked using the $\omega, \phi$, and $\kappa$ angles [11], and uses normal rotation matrices. The global camera position is in the form of $t_1, t_2$, and $t_3$. The width $W$ and height $H$ of the image are in pixels. The $u$ and $v$ relate to the image plane coordinates. These definitions and formulas are from [31]. Equation2 and Equation3 are formulas for the non-matrix version of Equation1.

$$u = \frac{f_x(r_{11}X + r_{12}Y + r_{13}Z + t_1)}{r_{31}X + r_{32}Y + r_{33}Z + t_3} + W/2 + ppa_x \tag{2}$$

$$v = \frac{f_x(r_{21}X + r_{22}Y + r_{23}Z + t_2)}{r_{31}X + r_{32}Y + r_{33}Z + t_3} + H/2 + ppa_y \tag{3}$$

These are the base case variables upon which a set of test cases can be built, more of which is discussed in Chapter 3 Methodology. Image parameters can be summarized as a camera model. [13] gives an overview of how to achieve this in a 3D computer program.

### 2.3 Feature Extraction Methods

The following section describes three feature extraction methods that are applied in the main research.
**SIFT**

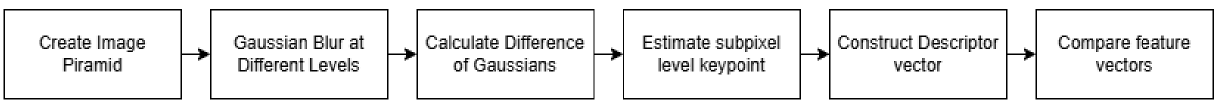



Figure 2: SIFT workflow

SIFT, also known as Scale-Invariant Feature Transformation, is an object recognition algorithm from 1999 [17] that is commonly applied to aerial images for the detection and matching of keypoints. It works by applying a variety of image processing techniques. A rough overview of the process is given in Figure2. The process works by taking an even-sized digital image and calculating the down-scaled pyramids for the image, resulting in a number of layers calculated using Algorithm1.

**Algorithm 1** Calculate the Number of Image Pyramid Layers

**Require:** $size \geq 1$
**Ensure:** $layers$ is the total number of pyramid layers

$layers \leftarrow 0$
**while** $size > 1$ **do**
  $size \leftarrow \lfloor size/2 \rfloor$ ▷ Divide size by 2
  **if** $size \bmod 2 \neq 0$ **then** ▷ Check if size is odd
    $size \leftarrow size - 1$ ▷ Adjust odd size to even
  **end if**
  $layers \leftarrow layers + 1$ ▷ Increment layer count
**end whilereturn** $layers$

Each pyramid layer is then blurred using an approximated Gaussian kernel with different levels of sigma that can be changed as a hyperparameter. Then, the gradient of the images is calculated, resulting in highlighting points that are uniquely defined in their local area. In the case of a good keypoint, the point will appear at different pyramid and sigma blur levels. The exact position of the keypoint is then determined based on the weight of the pyramid and how resistant the point is to the levels of blur, resulting in a sub-pixel keypoint. This means that the point is known in the image plane to a decimal level. For these detected keypoints, a descriptor vector is calculated, which consists of a 128-dimensional vector that encodes the gradients in a local area. These descriptor vectors are relatively standardized, as will be seen in the convolution neural network approach. This process can then be repeated for the whole image set, resulting in a number of keypoint candidates and their descriptor vectors. These vectors can then be matched using a dot product, resulting in a set of connections [30] and [2].

The disadvantages of this method are highlighted in [39]. A large number of variants have been created to overcome these issues. The main problem, as it relates to oblique aerial adjustment, which was briefly mentioned in the introduction, is the rotation at which keypoints are detected when comparing nadir and oblique images. It would seem that SIFT is capable of detecting points with a single rotated axis. This is how different flight lines are still capable of being connected. However, when a second and third axis of rotation are introduced, such as in an oblique image, SIFT is no longer capable of matching keypoints reliably. Other disadvantages are listed in [21]. Most of these disadvantages have been addressed through the introduction of neural network approaches.

## 2.4 DISK

Due to recent advancements in neural networks, a new method was proposed in 2020 called DISK, also known as DIScrete Keypoints [35]. This method combines image processing techniques and a U-net architecture for the detection of keypoints. A rough overview is given in Figure3. The results are similar to SIFT, given that it produces a 128-dimensional vector that works as a descriptor. The process starts by making the image square with a resolution that can be divided by 8. Padding can be used to achieve this. The resulting image is then used to calculate highlights by computing the grayscale, normalizing it, and finally applying a singular Gaussian filter to generate highlights. The image is then divided into a grid consisting of 8x8 cells, where each cell identifies its most prevalent keypoint. These keypoints are then passed through a neural detection net, which uses depth-masked training data to create a more comprehensive description of the keypoint that can be viewed from a variety of angles. Each keypoint also gets a confidence score, which can then be used in the matching.

Similar to SIFT, a number of disadvantages have been identified, which has resulted in a list of variants, as discussed in [7]. One of these variants allows for a variety of different light levels to be present in the image capture, showing relatively good results. Another possible expansion is described in [16], which uses the capture date of the image to base the confidence level for connecting images taken over a larger

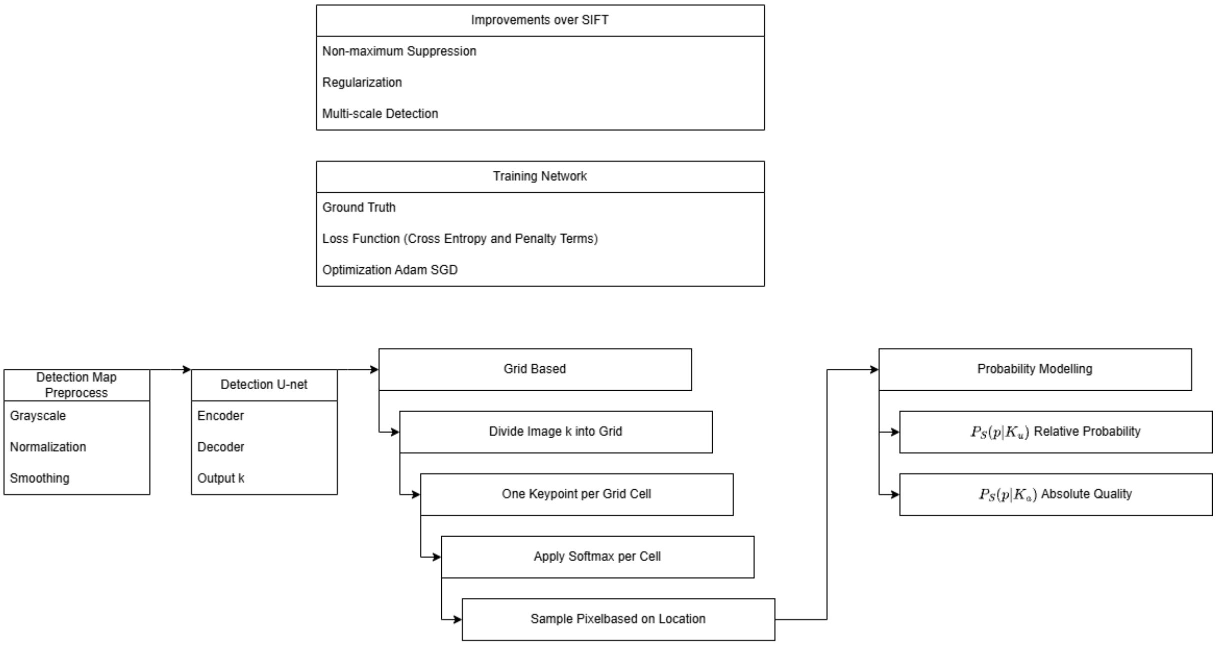



Figure 3: DISK workflow

time scale. Another version is [4], which removes the normalization, instead using the RGB-D range of pixels to better describe each keypoint. Each of these methods only detects keypoints, which, as mentioned before, are assigned an image position that, in the case of neural networks, is generally at the pixel level, along with a descriptor vector that can then be used for matching, as described in the next section.

## 2.5 LightGlue

LightGlue, first introduced in [15], is a derivative of SuperGlue, which was introduced in [25]. It aims to speed up the matching process by introducing a level of confidence when assigning pairs. It works based on the 128-dimensional descriptor vector as well as the location of the image to create a graph. This graph can be used with other point graphs from other images and matched according to their structure. The creation of the graph includes a hyperparameter called depth, which is stated to increase accuracy at the cost of performance. An overview of the workflow is given in Figure4.

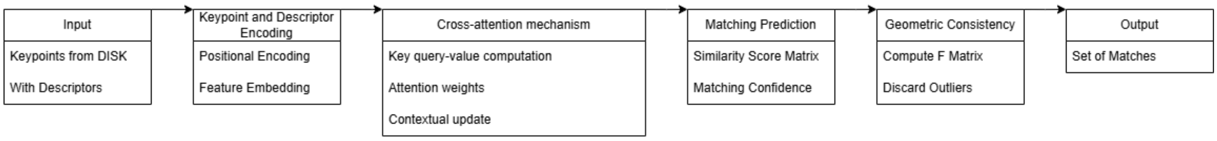



Figure 4: LightGlue workflow

LightGlue has been shown to work well with affine transformations (translation, rotation, shearing, etc.) of the image space, which addresses the inherent problems present in SIFT. LightGlue's graph construction works based on a pre-trained model that uses depth of field as an additional element. It resolves the issues that SIFT encounters, as LightGlue incorporates spatial awareness through the point graph it constructs.

## 2.6 Bundle Adjustment Using the Jacobian Matrix

The goal of bundle adjustment is to minimize the reprojection error, represented by the following formula [29]:

$$\min_{g,c,x} ||b - f(R(g), c, x)||^2 \tag{4}$$

This formula uses the linear model explained in Section 2.2, based on the rotation $R$, image center $c$, and camera position $x$. It projects the point using the function $f$ (as defined in Equation1) and compares it to the image plane observation point, denoted as $b$. This process is repeated for a large number of keypoints (at least 18 and up to a maximum of 81) per image, as specified in [11]. This non-linear minimization is performed by putting Equation4 into a Jacobian matrix. This matrix encodes both the camera and points in a non-square matrix. A set of rules are outlined in [6], including the minimal requirements for creating a bundle adjustment system. It also states that even though it is a non-linear system, due to the absence of a regularizer while searching for the best-fitting solution, the result will always be the statistically optimal solution based on the given parameters. The final thesis will include a step-by-step explanation of bundle adjustment as it pertains to the creation of the tests.

Chapter 1 (Introduction) discusses that an off-the-shelf bundle adjustment core will be used for the completion of this research. In this case, the GeoDelta core will be used. This is a Csharp-based implementation that has been tested on various applications and is optimized for GPU acceleration. Although this core is not publicly available, a variety of open-source versions exist. These include:

- **Google Ceres Solver:** A C++ implementation [1], which is a linear solver that also allows for the minimization of a bundle adjustment system.
- **German Bundle Adjustment (JAG3D):** An implementation from [3], slightly different from the Dutch implementation due to the inclusion of shift vectors whenever possible [18].
- **PyTorch 3D:** A Python-based extension built on PyTorch, which facilitates bundle adjustment tasks [24].

These open-source alternatives provide flexibility and accessibility for various bundle adjustment requirements.

# 3 Methodology

The third chapter of the P2 document describes the chosen methodology to address the main research question based on the findings of the literature review conducted in Chapter 2. This chapter leverages the aerial image pipeline and expands upon the selected elements to enhance their applicability. Additionally, an overview of the datasets and tools required to implement the methodology is provided.

## 3.1 Approach

The general approach can be described in five steps shown in Figure 5. The first step is the creation of a 3D city mesh rendering tool that allows for the creation of image test sets. Each test set will consist of roughly 100 camera positions with a as of now not yet determined resolution. This amount was chosen for its maximization in the possible tests that can be run whilst also minimizing the computational time. With the images completed for both nadir and oblique camera angles the 3D reconstruction discussed as SFM in Chapter 2 Related work can begin. For this COLMAP which is open source software that helps with the initial guess of of the camera positions is used in the keypoint detection and matching. An initial test could be based on hand picked features. While later tests for nadir images this is done using both SIFT and DISK/LightGlue and for the oblique images it is only done using DISK/LightGlue for reason mentioned before. These keypoints that are now known both on the image plane and their estimated 3D positions is used in the adjustment to calculate a best fit. The adjustment process and its resulting adjusted positions can be used to search for errors in the data. In the case of synthetic tests the expectancy is that the results reflect the input of the test case. Finally a w-test and f-test is calculated which is a rough estimation of its success. Further success can be drawn from its adherence to GCP which are collected terrestrially either through GNSS or total station. The more accurate a GCP can be be measured the more likely it is the adjustment was correctly thus resulting in a higher measurability in accordance with the rules of for example the BGT.

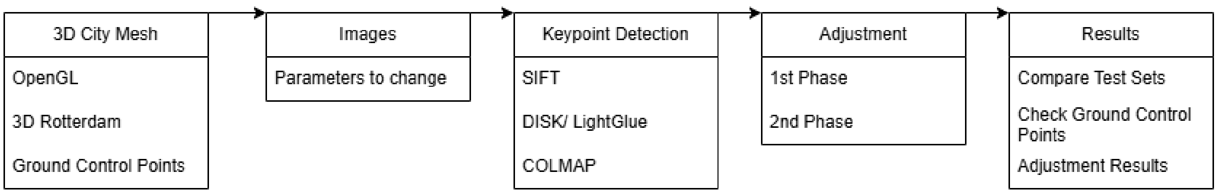



Figure 5: Research workflow

As mentioned before the research begins by establishing a 3D rendering environment that allows users to configure and modify various parameters to generate synthetic image datasets. This setup facilitates a controlled environment for simulating diverse camera configurations and conditions, providing the flexibility needed to evaluate various scenarios. The initial set of synthetic images will be generated based on the following camera parameters:

- X (position along the X-axis)
- Y (position along the Y-axis)
- Z (position along the Z-axis)
- Omega (rotation around the X-axis)
- Phi (rotation around the Y-axis)
- Kappa (rotation around the Z-axis)
- Focal distance (distance between the lens and the image plane)
- $PPA_X$ (principal point offset along the X-axis)
- $PPA_Y$ (principal point offset along the Y-axis)
- Image width (output image width in pixels)
- Image height (output image height in pixels)

- Pixel size (physical size of each pixel)

Chapter 5 Schedule lines out that the P3 phase will be used for the creation of test cases based on the possiblities of the 3D rendering tool. Examples of one of these test cases could involve introducing a small offset in the principal point $PPA_X$ to simulate a suboptimal camera calibration. Another test case might involve varying the focal distance to emulate real-world variations in camera optics. Further testing is outlined in the MoSCoW analysis in Chapter 1, with additional scenarios designed to simulate external factors that could impact camera performance. These external factors include:

- Variations in brightness and darkness
- Weather conditions, such as clouds, snow, or water reflections
- Vegetation, such as trees obstructing visibility
- Terrain features, such as hillsides and uneven surfaces

Once the test cases are defined, the second phase of the research involves applying keypoint extraction and matching algorithms such as SIFT, DISK, and LightGlue to extract features from the synthetic images. These features will be used to create a 3D reconstruction of the camera's positional and orientational parameters. Bundle adjustment techniques will then be employed to refine these initial estimates and improve the accuracy of the reconstructed parameters.

The final step involves comparing the refined data against real-world benchmarks, such as those derived from the flight plans and calibrated camera models used for the city of Rotterdam. The comparison will assess the effectiveness of the proposed methodology in generating accurate data for BGT (Basic Registration of Topography) mapping. Validation will draw on methodologies used in the HTW [11] tests.

## 3.2 Datasets and Tools

The city of Rotterdam, located in the Netherlands, serves as the initial study area for this research. This area was selected due to the availability of extensive geospatial data, making it an ideal testbed for validating the proposed methodology. For instance, synthetic datasets will be based on the 2024 flight plan created by MiraMap. This ensures that the synthetic data closely replicates real-world conditions, enabling meaningful comparisons. Additionally, real-world datasets, including calibrated camera models and image data, have been made available by the Municipality of Rotterdam and MiraMap, providing critical benchmarks for validation. The 3D rendering tool will leverage the Rotterdam 3D dataset to recreate detailed representations of the city's buildings.

To effectively implement the methodology, various tools and software packages are required. A detailed overview of these tools, their roles, and their descriptions is provided in Table 1. Notably, some software packages, such as those developed by GeoDelta, are proprietary and not freely available. To ensure reproducibility, open-source alternatives implementing similar algorithms are referenced wherever possible.

The rendering tool will be developed in C++ using libraries like OpenGL for 3D graphics and dear ImGui for graphical user interface development. The thesis document will be formatted using LaTeX for professional and structured presentation.

By combining these datasets and tools, the methodology ensures robust and reproducible results while maintaining high accuracy in simulating and analyzing real-world conditions.

| Tools | Description |
|---|---|
| Phoxy2 | A collection of photogrammetric software developed by GeoDelta. |
| Omnibase | A web-based viewer for visualizing geodetic data, developed by GeoDelta. |
| COLMAP | An open-source Structure-from-Motion (SfM) tool for estimating image positions. |
| QGIS | An open-source GIS viewer used to visualize 2D geospatial data. |
| LaTeX | A markup language used for creating structured, professional documents. |
| Microsoft Visual Studio | An integrated development environment (IDE) for creating the rendering tool. |
| C++ | A programming language chosen for its speed and extensive library support. |
| OpenGL | A framework for creating 3D graphics and GPU-accelerated rendering. |
| dear ImGui | An open-source GUI library compatible with OpenGL and C++. |

Table 1: Overview of tools used in the research [19], [28], [23] and [26]

# 4 Rich Picture

The creation of a Rich Picture is to enhance the understanding of the research at hand, showing different factors such as the student and the supervisors and showing the responsibilities of the different parties as well as the overall goals and the general pipeline/methodology of the research.


Figure 6: Rich Picture Related to the Completion and Process of the Research

# 5 Schedule

The thesis, as outlined in the previous four chapters, is scheduled to take place during the 20242025 academic year, spanning the second, third, and fourth quarters. This chapter provides an overview of the key milestones, deadlines, and a general timeline for the thesis process.

## 5.1 Deadlines

Due to the recent changes in the BK graduation rules as laid out in the graduation manual [5], a new graduation system (called Asystem) has been introduced. Since this is the first year this system has been implemented, the old system (Psystem) is also still in place. The following table, which consists of the deadlines and milestones, includes both the old and the new systems. Certain dates are not yet set; as such, rough time periods are given. These time frames are used as phases in the second section of the chapter called Time Planning.

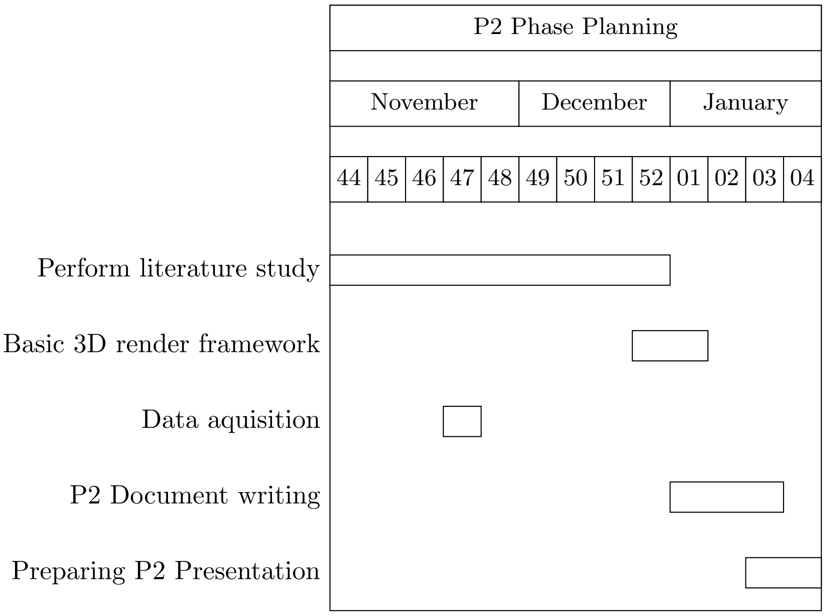
| Date | Old Name | New Name | Description |
|---|---|---|---|
| 15th of November 2024 | P1 project description | - | Not named in the new system but functions as a starting point |
| 15th of January 2025 | P2 report | A1 report | The report needs to be uploaded one week before the P2 presentation. |
| 22nd of January 2025 | P2 presentation | A1 presentation | P2 Presentation moment |
| 7th to 13th of April | P3 meeting | A2 presentation | Also referred to as the Midterm phase on MyCase. Purely a formative moment of which the form is decided within the graduation team. |
| 5th to 11th of May | P4 report | A3 report | The report needs to be uploaded one week before the P4 presentation. |
| 12th to 18th of May | P4 presentation | A3 presentation | The secondary pass/fail moment that initiates the end of the thesis. First-time presence of the co-reader. |
| 9th to 15th of June | P5 report | A4 report | The report needs to be uploaded one week before the P5 presentation. |
| 16th to 22nd of June | P5 presentation | A4 presentation | Also referred to as the finalization phase on MyCase. This marks the end of the thesis. |

Table 2: Thesis period deadlines and milestones in accordance with MyCase and the Bouwkunde Graduation Manual

## 5.2 Time planning

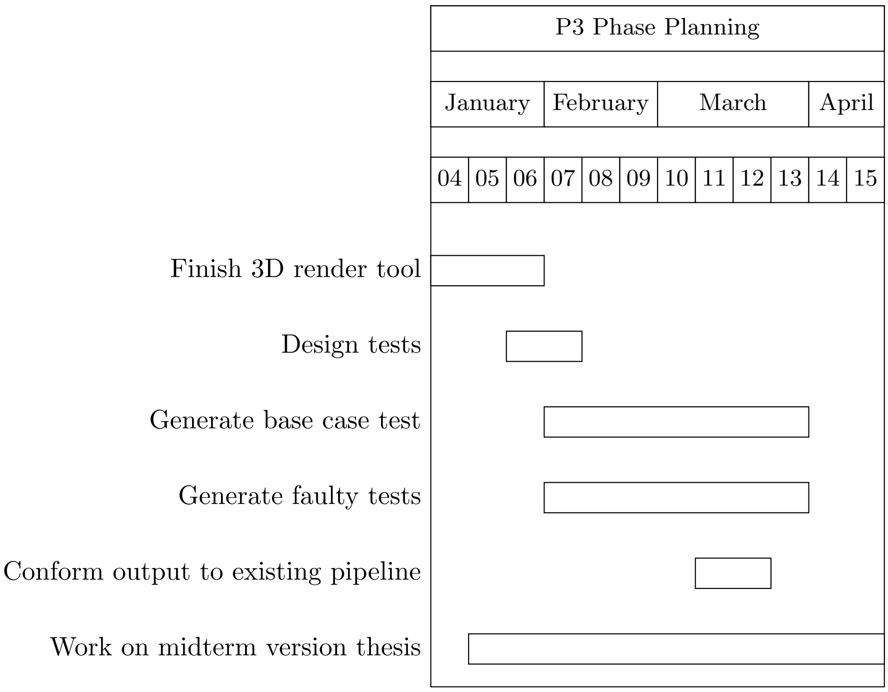
**P2 Phase**

The first phase which takes place during the second quarter of the 2024-2025 academic year. It is marked mostly by the literature study which takes place during most of the phase as well as the writing and the creation of the P2 document that describes the initial research approach through the creation of the methodology and research questions as well as the demarcation of the thesis topic in the scope. The Phase is concluded by the P2 presentation which takes place on January the 22nd at 8:45.



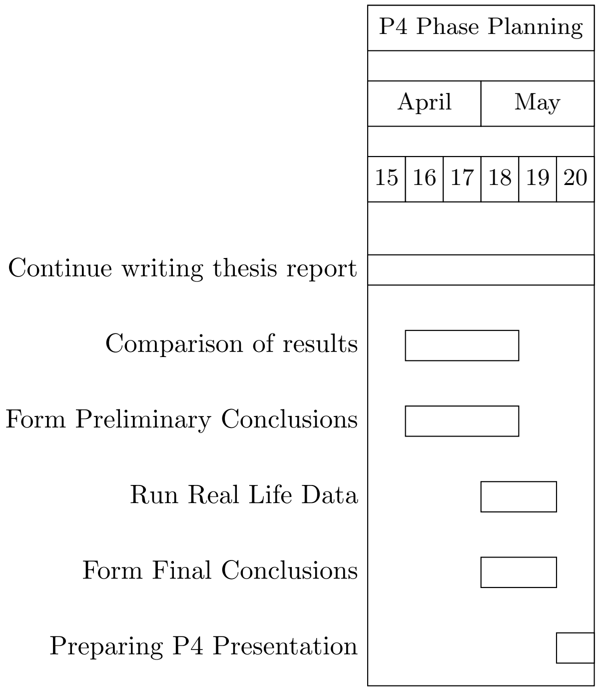

**P3 Phase**

After the approval of the P2 document through the P2 presentation the second phase which takes place during the third quarter of the 2024-2025 academic year with it ending with a midterm meeting which is purely formative meaning that no grade and no structure is attached to it. The phase is mostly about setting up the methodology discussed in Chapter 2 as well as the creation of synthetic test data.



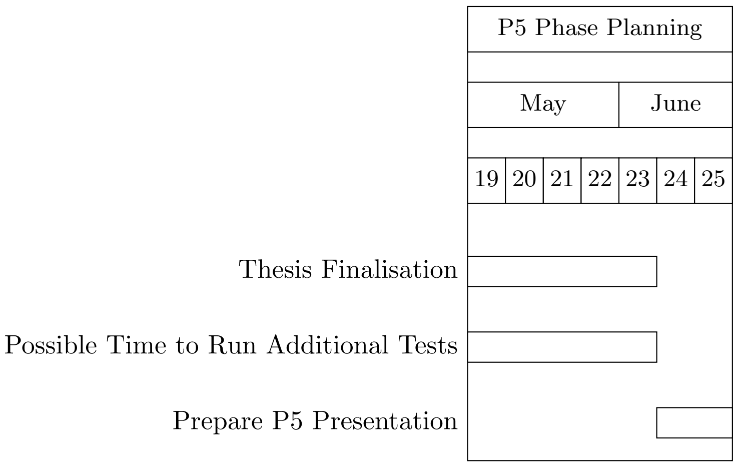
**P4 Phase**

The third phase which is to work towards the P4 pass/ fail which takes place during the third and fourth quarter of the 2024-2025 academic year. Whilst it is the shortest phase it might be one of the most important mostly due to being important for the finalization of the thesis as well as the comparison of results and forming conclusions based on the synthetic tests.




**P5 Phase**

The last and final phase of the thesis takes place during the fourth quarter of the 2024-2025 academic year and will only commence after the green light has been given for P4. It is mostly used for reworking the feedback provided by the supervisors as well as the co-reader. The period is marked at the end by the P5 presentation.



# 6 General Information

The final chapter of the Graduation Plan pertains to some general information which defines the process of the thesis but does not necessarily fit into the rest of the P2 report.

Supervision will be provided by the following, who together with the student form the graduation team:

- Delft University of Technology
  - First supervisor: Edward Verbree of GDMC
  - Second supervisor: Martijn Meijers of GDMC
- GeoDelta
  - External supervisor: Annemieke Verbraeck

Other general aspects related to the thesis are:
**Workplace** The thesis will take place entirely at the GeoDelta offices, located at Kanaalweg 4, 2628 EB Delft, which from here will be referred to as the place of work. As mentioned before, the research will roughly take place between the 1st of November 2024 and the 31st of July 2025. This corresponds to the second, third, and fourth quarters of my third year of MSc Geomatics, which is the 2024–2025 academic year. The time spent during the second quarter will roughly amount to 24 hours a week, all of which will take place at the place of work. During the second semester of the academic year, a full 40 hours a week is planned to be spent on the thesis, which will also fully take place at the place of work.

**Meetings** Meetings will take place on a weekly basis, with the first and second supervisors attending bi-weekly. The meetings themselves will occur on Tuesdays. Furthermore, there might be meetings with external experts, colleagues, or other teachers, which will take place at moments to be determined.

**Communication** Text-based communication will mainly occur through the use of TU Delft student email. All three supervisors will be included in each email sent by the student to prevent information from being lost.

**Progress** Both the meetings and emails will be used to inform the supervisors of the current progress. Certain milestones and deadlines, as laid out in the previous chapter (Schedule), will also be part of disclosing the progress.

**Report and Code** The final submission of the report will be uploaded to the TU Delft educational repository after receiving the green light upon the completion of P5. The final submission will include the P5 report, P5 presentation, and the P2 Graduation Plan. Due to the use of internal libraries in both the keypoint selection and matching phases, as well as the bundle adjustment phase, no code will be made public. Instead, the focus will be on the report to ensure a level of reproducibility in accordance with TU Delft rules.

## References


[1] Sameer Agarwal, Keir Mierle, and The Ceres Solver Team. Ceres Solver, 10 2023.

[2] Kevin Wood Robotics AI. Opencv python sift feature detection (sift algorithm explained + code), 2023. Accessed: 2025-01-13.

[3] Applied-Geodesy. GitHub - applied-geodesy/jag3d: Java·Applied·Geodesy·3D - Least-Squares Adjustment Software for Geodetic Sciences.

[4] Yiye Chen, Yunzhi Lin, Ruinian Xu, and Patricio A Vela. Keypoint-graspnet: Keypoint-based 6-dof grasp generation from the monocular rgb-d input. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7988–7995. IEEE, 2023.

[5] Technische Universiteit Delft. Graduation Manual Master Geomatics, 2024.

[6] Chris Engels, Henrik Stewénius, and David Nistér. Bundle adjustment rules. *Photogrammetric computer vision*, 2(32), 2006.

[7] Matheus A Gadelha and Bruno M Carvalho. Drink: Discrete robust invariant keypoints. In *2014 22nd International Conference on Pattern Recognition*, pages 821–826. IEEE, 2014.

[8] Norbert Haala, Mathias Rothermel, and Stefan Cavegn. Extracting 3d urban models from oblique aerial images. In *2015 Joint Urban Remote Sensing Event (JURSE)*, pages 1–4. IEEE, 2015.

[9] Joachim Höhle. Oblique aerial images and their use in cultural heritage documentation. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 40:349–354, 2013.

[10] Joachim Höhle. Photogrammetric measurements in oblique aerial images. *Photogrammetrie - Fernerkundung - Geoinformation*, pages 7–14, 01 2008.

[11] Kadaster. *Handleiding voor de Technische Werkzaamheden van het Kadaster*, 1996. Accessed: 2025-01-14.

[12] Kadaster. Basisregistratie grootschalige topografie (bgt), 2017. Accessed: 2025-01-15.

[13] Craig Kolb, Don Mitchell, and Pat Hanrahan. A realistic camera model for computer graphics. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pages 317–324, 1995.

[14] Mathias Lemmens, Christiaan Lemmen, Martin Wubbe, et al. Pictometry: potentials for land administration. In *Proceedings of the 6th FIG regional conference 12-15 November 2007, San José, Costa Rica./Fredriksberg: International Federation of Surveyors (FIG) 13p*, 2007.

[15] Philipp Lindenberger, Paul-Edouard Sarlin, and Marc Pollefeys. Lightglue: Local feature matching at light speed, 2023.

[16] Xingyu Liu, Shun Iwase, and Kris M Kitani. Kdfnet: Learning keypoint distance field for 6d object pose estimation. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4631–4638. IEEE, 2021.

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

[18] Michael Lösler. Java·Applied·Geodesy·3D (JAG3D) — Least-Squares Adjustment Software for Geodetic Sciences — software.applied-geodesy.org.

[19] Omar Ocornut. Dear imgui. https://github.com/ocornut/imgui, 2025. Accessed: 2025-01-14.

[20] Municipality of Rotterdam. 3D Rotterdam.

[21] Ives Rey Otero. *Anatomy of the SIFT Method*. PhD thesis, École normale supérieure de Cachan-ENS Cachan, 2015.

[22] Chao Pang, Jiang Wu, Jian Ding, Can Song, and Gui-Song Xia. Detecting building changes with off-nadir aerial images. *Science China Information Sciences*, 66(4):140306, 2023.

[23] QGIS Development Team. *QGIS: A Free and Open Source Geographic Information System*, 2025. Version 3.x, Accessed: 2025-01-14.

[24] Nikhila Ravi, Jeremy Reizenstein, David Novotny, Taylor Gordon, Wan-Yen Lo, Justin Johnson, and Georgia Gkioxari. Accelerating 3d deep learning with pytorch3d. *arXiv:2007.08501*, 2020.

[25] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks, 2020.

[26] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4104–4113, 2016.

[27] Nicholas Shorter and Takis Kasparis. Automatic vegetation identification and building detection from a single nadir aerial image. *Remote Sensing*, 1(4):731–757, 2009.

[28] Dave Shreiner, Graham Sellers, John Kessenich, and Bill Licea-Kane. *OpenGL Programming Guide: The Official Guide to Learning OpenGL*. Addison-Wesley Professional, 9th edition, 2016.

[29] Cyrill Stachniss. Bundle adjustment - 5 minutes with cyrill, 2020. Accessed: 2025-01-13.

[30] Cyrill Stachniss. Sift - 5 minutes with cyrill, 2020. Accessed: 2025-01-13.

[31] Peter Sturm. Pinhole camera model. In *Computer Vision: A Reference Guide*, pages 983–986. Springer, 2021.

[32] Peter Teunissen. *Adjustment Theory An Introduction*. VSSD, first edition, 2000.

[33] Peter Teunissen. *Network Quality Control*. VSSD, second edition, 2000.

[34] Peter Teunissen. *Testing Theory An Introduction*. Second edition, 2006.

[35] Michal J. Tyszkiewicz, Pascal Fua, and Eduard Trulls. DISK: learning local features with policy gradient. *CoRR*, abs/2006.13566, 2020.

[36] Hiddo Velsink. The elements of deformation analysis - Blending Geodetic Observations and Deformation Hypotheses.

[37] M.G.V Vosselman. Fotogrammetrie I, 1 1996.

[38] Jinwang Wang, Lingxuan Meng, Weijia Li, Wen Yang, Lei Yu, and Gui-Song Xia. Learning to extract building footprints from off-nadir aerial images. *IEEE transactions on pattern analysis and machine intelligence*, 45(1):1294–1301, 2022.

[39] Jian Wu, Zhiming Cui, Victor S Sheng, Pengpeng Zhao, Dongliang Su, and Shengrong Gong. A comparative study of sift and its variants. *Measurement science review*, 13(3):122–131, 2013.