# Automated, 3D, Airborne Modeling of Large Scale Urban Environments

by Min Ding

## Research Project

Submitted to the Department of Electrical Engineering and Computer Sciences, University of California at Berkeley, in partial satisfaction of the requirements for the degree of **Master of Science, Plan II**.

Approval for the Report and Comprehensive Examination:

**Committee:**

Professor Avideh Zakhor
Research Advisor

Date

* * * * * *

Professor Ruzena Bajcsy
Second Reader

Date

# Abstract

A fast 3D model reconstruction methodology is desirable in many applications such as urban planning, training, and simulations. In this paper, we develop an approach for fast, automated 3D modeling of large scale urban environments based on airborne data. Since airborne data acquisition is considerably faster than ground based collection, our proposed methodology can scale to very large regions. At the core of our approach lies an automated algorithm for texture mapping oblique aerial images onto a 3D model generated from airborne Light Detection and Ranging (LiDAR) data. Our proposed texture mapping algorithm consists of two steps. In the first step, we combine vanishing points and global positioning system aided inertial system readings to roughly estimate the extrinsic parameters of a calibrated camera. In the second step, we refine the coarse estimate of the first step by applying a series of processing steps. Specifically, We extract 2D orthogonal corners (2DOCs) corresponding to orthogonal 3D structural corners as features from both images and the untextured 3D LiDAR model. The correspondence between an image and the 3D model is then performed using Hough transform and generalized M-estimator sample consensus. The resulting 2DOC matches are used in Lowes algorithm to refine camera parameters obtained earlier. Our system achieves 91% correct pose recovery rate for 90 images over the downtown Berkeley area, and overall 61% accuracy rate for 358 images over the residential, downtown and campus portions of the city of Berkeley.

# Acknowledgement

This research was conducted under the support of Heterogeneous Sensor Webs for Automated Target Recognition and Tracking in Urban Terrain (W911NF-06-1-0076).

I would like to thank my research advisor, Professor Avideh Zakhor, for all her guidance. Without her close attention and clear vision, this research could not be completed. I would like to thank Professor Ruzena Bajcsy, Professor Shankar Sastry and Doctor Allen Yang for teaching me necessary knowledge in computer vision. I would also like to thank all the members of the VIP lab and my friends in EECS at Berkeley for their generous help during the data acquisition process. Especially, I would like to thank Kristian Lyngbaek for his generous effort on manual point correspondence based camera pose determination.

Finally I would like to thank my parents and my girlfriend Angela for their warm support. Without their love, I could not have come this far.

# Contents

# Chapter 1

# Introduction

3D modeling of large scale environments is needed in many applications such as city planning, architecture design, telecommunication network design, cartography and fly/drive-through simulation. Due to their significance and vast potential, fast and automated model reconstruction has drawn great deal of attention and effort from many researchers in the past three decades. Large scale 3D model reconstruction methodologies can be generally classified into three main categories: image sensor based, range sensor based and combination of the two.

Image sensor based methods can be further classified into single view [1–7], multiview [8–12] and video based reconstruction [13, 14]. A detailed review and performance comparison for monocular building extraction can be found in [3]. In [4], Lin and Nevatia have generated 2D roof hypotheses from line segments extracted from aerial images. Good hypotheses are selected based on 2D evidence and verified against 3D evidence such as shadows and walls. Hypotheses are further pruned to avoid conflicts which do not exist in man-made structures. Fischer et al. have applied a similar concept to more complex building reconstruction, where a generic modeling approach based on hierarchical aggregation is taken [1]. This concept is further enhanced by the use of vanishing points for line segment extraction [3, 5, 6]. In particular, Shufelt et al. have used vanishing points to extract vertical, orthogonally horizontal and slanted roof lines [3].

In multiview based model reconstruction, vanishing points can also be used to calibrate the camera and to find the relative pose. Cipolla et al. require manual line segment matching between images, and use the projection matrix calibrated from vanishing points for model rectification [8, 9]. Werner and Zisserman avoid the user intervention of line matching by a combination of photometric constraints and trifocal geometry [10], and by affine invariant neighborhood based on isotropy of the gradient second moment matrix [11]. In [12], Noronha and Nevatia extend their single view approach by verifying the hypotheses generated in each image with the evidences from multiple images.

In [13, 14], Pollefeys et al. and Nistér et al. use video based model reconstruction. They first track features across frames and obtain relations among multiple views. Both the structure of the scene and the camera motion can be computed from the tracked features. Pairwise images are first rectified with the standard stereo configuration; these results are then integrated for the entire image sequence.

Even though the above vision based systems are partially successful, they are either too complex to scale to large urban model reconstruction, or lack the desired accuracy and true automation. Typically, these techniques are suitable for open and smooth terrain surface, but perform poorly in urban areas where occlusions are common place. Since late nineties, Light Detection and Ranging (LiDAR) sensors on airplane platforms have been actively used for urban model reconstruction [15–19] in order to mitigate the problems in vision based systems. LiDAR measures the time delay and intensity difference between the transmitted laser pulses and the reflected ones. It then finds the range and reflection property of the ground objects. It is a very accurate and cost-effective device, and can be used to create a digital surface model (DSM) with vertical accuracy up to 15 centimeters [18]. As a result, several approaches have been developed for model reconstruction from LiDAR data. Majority of the approaches try to fit parameterized roof planes to point clouds by applying invariant moment analysis [17]. A major problem in airborne LiDAR based model reconstruction is building extraction among trees and other natural objects. Brunn and Weinder have applied a statistical classification technique based on Bayesian nets to discriminate buildings from vegetation [15]. Haala and Brenner have used difference in surface reflectance measured from multi-spectral imagery to classify buildings, tree, and grass-covered areas [16]. Lodha et al. have applied AdaBoost algorithm to classify road, grass, buildings and tress using five features including height, height variation, normal variation, LiDAR return intensity and image intensity [20]. Recently, Secord and Zakhor have developed a tree detection technique based on weighted support vector machine (SVM) [21].

LiDAR based 3D model reconstruction is fast and accurate; however, it lacks texture information which can be essential for architecture design and drive/fly through simulations. Therefore, approaches combining the electro-optical vision sensors and LiDAR sensors have been actively investigated. For instance, Frueh and Zakhor have developed a fast and truly automated ground-based data acquisition system with two laser scanners and a digital camera mounted on a truck [22, 23]. The horizontal laser scanner is used to localize the truck and hence the acquisition apparatus, and the vertical laser scanner is used to acquire depth information. Since the digital camera is synchronized with the laser scanners and is rigid with respect to them, the texture information can be automatically mapped to the 3D model acquired by the laser scanner. Independently, Zhao and Shibasaki have developed a similar system but with a line camera [24].

## 1.1 Motivation for airborne based fast 3D model reconstruction

Clearly, there exists an ongoing trend of combining range sensors with image sensors for fast and automated 3D model reconstruction. However, all of the above approaches lack either accuracy or the scalability needed for creating textured models of large urban areas. Even though model geometry can be quickly generated from aerial images or LiDAR data as described previously, facade texture mapping is considerably more time-consuming due to the lengthy image acquisition process and manual correspondence between a 3D model and images. For instance, even though the approach in [22,23] is capable of continuously capturing detailed facade texture from a moving truck, it takes a long time to capture an entire city.

In this thesis, we develop an airborne based data acquisition and processing approach to achieve the desired scalability. Airborne LiDAR data is first collected to reconstruct the geometry of the models as demonstrated in [22]. Oblique aerial photos covering wide areas are then taken to achieve fast texture acquisition. Note that oblique pictures can cover both the rooftops and facades of buildings. This results in the scalability needed for generating a textured 3D city model in a much shorter time. This scalability is enabled by our fully automated image acquisition and camera registration system to be described shortly.

## 1.2 Related work

It is important to contrast our work with other related work. Stomas and Liu have attempted similar texture mapping from ground based multiview images [25–28]. They first use vanishing points to find the rotation and the focal length of all the images. They propose rectangular parallelepipeds, windows and doors which are inherently present in building facades, as features to match LiDAR data with images. With two pairs of correct 3D to 2D parallelepiped matches, they are able to identify the translation parameters of a subset of images. They further refine the camera parameters by performing 3D point cloud correspondence between the LiDAR data and the sparse point cloud generated from multiview geometry, i.e. structure-from-motion. Their algorithm requires multiview imagery, and run into difficulties if there are no two pairs of correctly matched parallelepipeds. Furthermore, they take advantage of the ground based image acquisition where clear parallelism and orthogonality of building contours are visible with little occlusion. In contrast, our system only uses a single aerial view, and can handle complex urban scenes with significant occlusions.

As for the multiview approach, Zhao et al., Hsu et al., and Neumann et al. have used video to texture map the 3D point clouds from range sensors [29–31]. Hsu et al. first use the tracked features for inter-frame pose prediction. The predicted pose is then refined by aligning the projected 3D model lines to those in images [30]. Neumann et al. follow a similar idea by implementing an extended Kalman filter (EKF) to perform inter-frame camera parameter tracking using point and line features [31]. Zhao et al. criticize these methods in that they cannot handle scenes lacking dominant lines, and that they can lose track in situations with large pose prediction error due to occlusions [29]. Instead, they generate 3D point clouds from a video as in the multiview case mentioned previously, and use iterative closest point (ICP) algorithm to align the video to the 3D point clouds from a range sensor. Even though this method has been demonstrated to be reliable, it still requires manual correspondences to initialize the first few video frames. More importantly, it is computationally expensive to generate 3D point clouds from a video. Therefore, these video based methods cannot achieve the scalability and automation of our proposed system.

Our approach is similar to the single view approach taken by Lee and Nevatia et al. [32–34]. They use vanishing points and random sample consensus (RANSAC) based 3D-2D line pair match

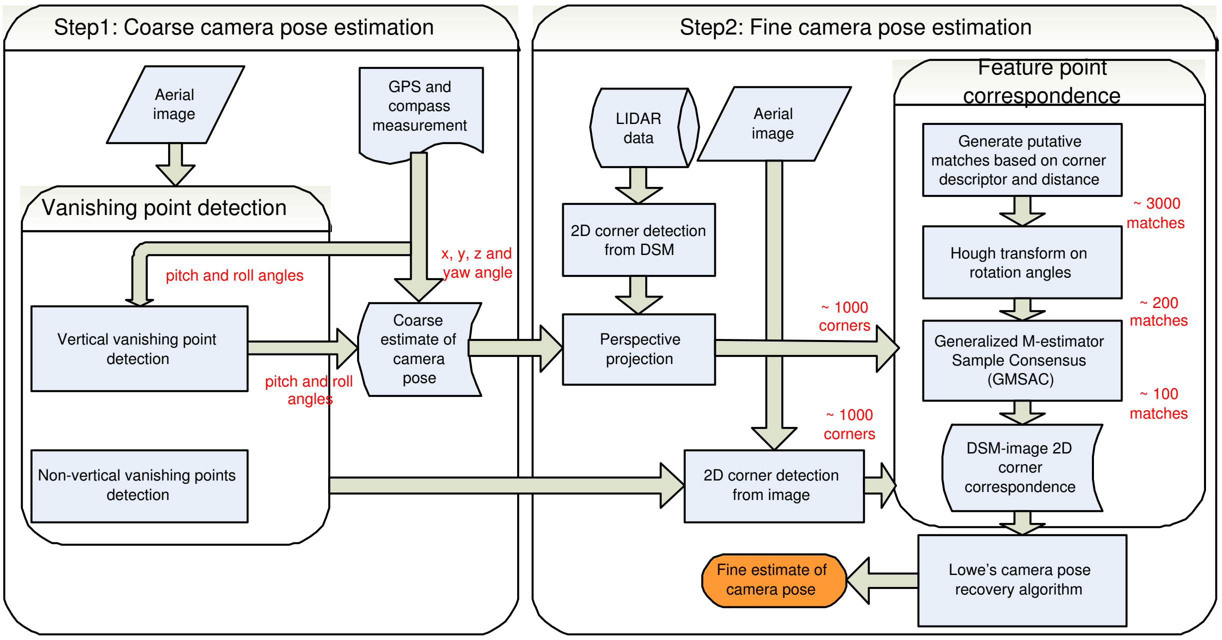

Step1: Coarse camera pose estimation
Aerial image
GPS and compass measurement
Vanishing point detection
pitch and roll angles
x, y, z and yaw angle
Vertical vanishing point detection
Coarse estimate of camera pose
pitch and roll angles
Non-vertical vanishing points detection
Step2: Fine camera pose estimation
LIDAR data
Aerial image
2D corner detection from DSM
Perspective projection
~ 1000 corners
~ 1000 corners
2D corner detection from image
Feature point correspondence
Generate putative matches based on corner descriptor and distance
~ 3000 matches
Hough transform on rotation angles
~ 200 matches
Generalized M-estimator Sample Consensus (GMSAC)
~ 100 matches
DSM-image 2D corner correspondence
Lowe's camera pose recovery algorithm
Fine estimate of camera pose

*Figure 1.1: Camera registration system overview*

to find the camera pose. However, they share the same weakness in the multiview based methods as they only deal with ground based images for a single building where clear parallelism and orthogonality of building contours are visible with little occlusion. Recently, Hu et al. have created a system capable of aerial and ground based image mapping [35, 36]. However it requires human interactions in many places. For instance, it requires interactive edge extraction from aerial images to obtain building contours. It also requires 10 pairs of manually identified point correspondence to align aerial images to LiDAR data.

## 1.3 Overview of the proposed system and major contributions

Our system first creates a non-textured 3D model from LiDAR data as described in [22]. It then tackles the camera registration problem in two steps as depicted on Fig. 1.1. First step is to obtain coarse camera parameters from a global positioning system (GPS) aided inertial system and vanishing points from image analysis. From the measurement device, coarse estimates on camera position, and its heading angle can be obtained. The other two angles of the camera's rotation are estimated from the position of the vanishing point of vertical lines in the 3D world space. With the intrinsic camera parameters fixed and known as a priori, the entire set of camera parameters can therefore be coarsely estimated. In this step, vanishing points corresponding to non-vertical lines are also detected. They will be used for 2D orthogonal corner extraction from images in the second step.

The second step of our proposed approach uses 2D orthogonal corners in the 3D space as features, in order to refine the coarse camera pose estimate obtained in the first step. 2D orthogonal corners (2DOCs) correspond to orthogonal structural corners, where two orthogonal building contour lines intersect. 2DOCs are extracted from a digital surface model (DSM) obtained via LiDAR data processing [22], as well as from aerial images based on the orthogonality information implied by the vanishing points detected earlier. After projecting 2DOCs from the DSM by the coarse camera parameters obtained in the first step, putative DSM-image 2DOC matches are generated based on distance and similarity in corners' descriptors. Finally Hough transform is performed to screen out majority of the spurious matches, and then generalized M-estimator sample consensus (GMSAC), a modified RANSAC is used to identify the correct DSM-image 2DOC matches. From the correct 2DOC matches, Lowe's camera pose recovery algorithm [37] is used to obtain the refined camera parameters. With the refined camera parameters, texture is mapped from images to the triangular mesh model generated from the LiDAR data.

Our contribution in this thesis is threefold. First is our vanishing points detection algorithm which is capable of handling aerial images in complex urban scenes. Second is our innovative feature point, 2DOC, which is the key element for pose refinement. The last one is our overall scalable system design. This system is considerably more computationally efficient as compared to other existing techniques [29, 53]. In particular, it is over 450 times faster than our previous exhaustive search approach [53]. By taking advantage of the parallelism and orthogonality inherently present in city models, we are able to apply well-justified heuristics to design a scalable and automated camera registration and texture mapping system for oblique aerial imagery.

## 1.4 Organization of this thesis

This thesis is organized as follows. The next chapter explains in details how a coarse set of camera parameters is obtained with emphasis on the new vanishing point detection algorithm. Chapter 3 presents the 2DOC and how it can be extracted from a DSM and aerial images. It then describes the point correspondence algorithm based on GMSAC for camera parameter refinement. Chapter 4 gives a brief description on texture-mapping once the camera pose is available. Following data acquisition process in Chapter 5, Chapter 6 examines the performance of the proposed system on 358 aerial images taken over three square-kilometers of urban and suburban areas in Berkeley, California. Finally this thesis concludes with some future directions of research.

# Chapter 2

# Coarse Camera Parameters Acquisition

As mentioned in the previous chapter, the extrinsic camera parameters are obtained in two steps: a coarse estimation followed by a refinement process. The refinement step is presented in the next chapter. In this chapter, the camera parameters are decoupled into those that can and cannot be estimated from aerial imagery. First the camera position and yaw angle are recorded from a GPS aided inertial system, NAV420CA from Crossbow in this thesis. This is because the position and yaw angle of a camera cannot be identified from an image unless some landmarks, position of the sun or shadows are considered. The other camera parameters are estimated from a vanishing point corresponding to vertical lines in the 3D world space. Finally, a robust non-vertical vanishing points detection algorithm in urban settings is presented, which is used in the pose refinement step presented in the next chapter.

## 2.1 The camera model and notation

A calibrated camera model is first assumed:

$$\lambda x = [\mathbf{R} \quad \mathbf{T}]X \tag{2.1}$$

where $x = [u, v, 1]^T$ is the coordinate on the image plane of $X = [x_w, y_w, z_w, 1]^T$ in the 3D world space after perspective projection; **R** and **T** are matrices related to the extrinsic parameters, and $\lambda$

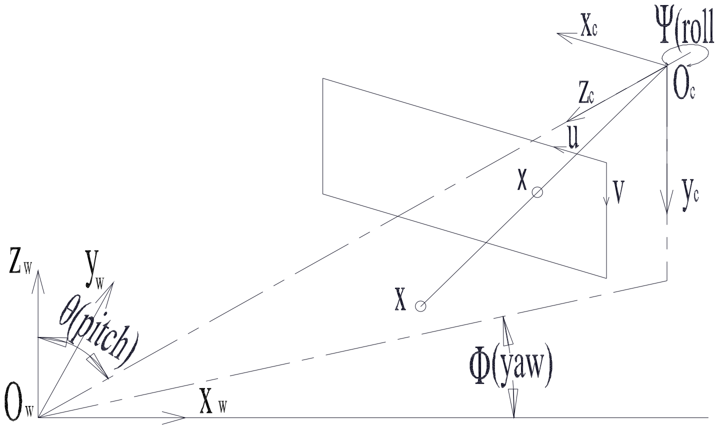


*Figure 2.1: Definition of extrinsic camera parameters. Yaw ($\phi$) is the angle between $x_w$ and projection of $z_c$ onto $x_w y_w$ plane. Pitch ($\theta$) is the angle between $z_w$ and $z_c$. Roll ($\psi$) is the rotation angle with respect to $z_c$.*

is a scaling factor. **R** is the relative rotation matrix, and **T** is the relative position matrix from the world coordinate, $O_W$, to the camera coordinate, $O_C$, as shown in Fig. 2.1. With the yaw ($\phi$), pitch ($\theta$) and roll ($\psi$) of a camera defined as shown in Fig. 2.1, **R** can be shown to be:

$$\mathbf{R} = \begin{bmatrix} -cos\psi sin\phi + sin\psi cos\phi cos\theta & cos\psi cos\phi + sin\psi sin\phi cos\theta & -sin\psi sin\theta \\ sin\psi sin\phi + cos\psi cos\phi cos\theta & -sin\psi cos\phi + cos\psi sin\phi cos\theta & -cos\psi sin\theta \\ -sin\psi cos\phi & -sin\theta sin\phi & -cos\theta \end{bmatrix} \tag{2.2}$$

$\mathbf{T} = [T^x, T^y, T^z]^T$ is the position of the origin of the world reference frame with respect to the camera reference frame. The calibrated camera assumption is not restrictive because the camera's intrinsic parameters are fixed during image acquisition and estimated afterwards. This implies that our camera focal length is fixed during the entire data acquisition.

## 2.2 Vertical vanishing point detection for pitch and roll estimation

With the assumption of a pin hole camera projection shown in Equation (2.1), it can be shown that a set of parallel lines in a 3D space is projected to a set of lines on the image plane which intersect at a common point. This point is referred as a vanishing point. Vanishing points have been widely exploited to obtain camera parameters such as focal length and rotation angles [8, 9, 25–28, 32–34, 36]. In the literature, robust and accurate vanishing point detection methods have been intensively explored. The existing approaches can be divided into four major categories. The most commonly used one is Gaussian sphere approach [38, 39] which can be generalized into Hough transform. The other three recent approaches are based on Expectation-Maximization (EM) algorithm [40], General Principle Component Analysis (GPCA) [41] and RANSAC [6]. All the above methods aim to find intersections among detected line segments, assuming simple geometry in a scene where parallel lines dominate. They have been shown to yield successful detection in indoor settings or outdoor settings where only a few buildings appear on an image. However, these algorithms typically fail in complex urban settings where multiple buildings exist and their footprint alignments are not necessarily parallel. In this situation, intersections in the 3D space are falsely classified as vanishing points. Instead, we develop an efficient and robust method for complex urban scenes. The vanishing point corresponding to vertical lines in the 3D model is first detected to estimate the pitch and roll angles of a camera. This particular vanishing point is referred as vertical vanishing point. Then an innovative non-vertical vanishing points detection algorithm is presented, which is used in the second step of pose refinement to be described in the next chapter.

### 2.2.1 Line segment extraction

To detect vanishing points, it is necessary to extract line segments from images. Canny edge detector is first used, and recursive endpoint subdivision is applied on the contours to obtain line segments [37]. Any two line segments are linked if they have similar angles and their endpoints are close to each other. A typical result after line segment extraction on an aerial image with 1024

X 768 pixels, covering three city blocks in a downtown area, is shown in Fig. 2.2.

### 2.2.2 Vertical vanishing point detection

The vertical vanishing point is extremely important since its coordinate on the image plane provides an estimate for the pitch and roll angles of a camera. It is also a consistent measurement since no matter how the footprints of buildings are aligned on the ground, their vertical contour lines are almost always parallel to each other. We find the vertical vanishing point with a method similar to Gaussian sphere approach [38]. A Gaussian sphere is a unit sphere with its center at $O_c$, the origin of the camera coordinate, as shown in Fig. 2.3. Each extracted line segment from an image forms a plane with $O_c$, intersecting the sphere to create a circle. These circles from multiple line segments are accumulated on the Gaussian sphere. It is assumed that the point on the sphere at which the maximum number of circles cross, represents the direction shared by multiple line segments, e.g. v in Fig. 2.3; furthermore, the intersection between the line connecting $O_C$ to v and the image plane is the vanishing point.

Although this consensus-based approach is robust in many situations, Shufelt has noted situations where the underlying assumptions can fail [39]. For instance, the texture pattern and natural urban setting can lead to maxima on the sphere which do not correspond to the true vanishing points. To overcome this shortcoming in our application, we limit the search for the maximum to a smaller specific region on the sphere centered at NAV420CA's pitch and roll readings and spanned by the angle measurement error of three degrees. This follows from the direct relationship between the vertical vanishing point's position on the Gaussian sphere and the pitch and roll angles of the camera, which will be presented in the next subsection. All the line segments intersecting this vertical vanishing point within a certain tolerance are considered as the projected vertical lines in the 3D space. They are highlighted with dark blue as shown in Fig. 2.4. After this process, all the identified vertical lines are eliminated and the rest of the line segments are used to obtain the non-vertical vanishing points.

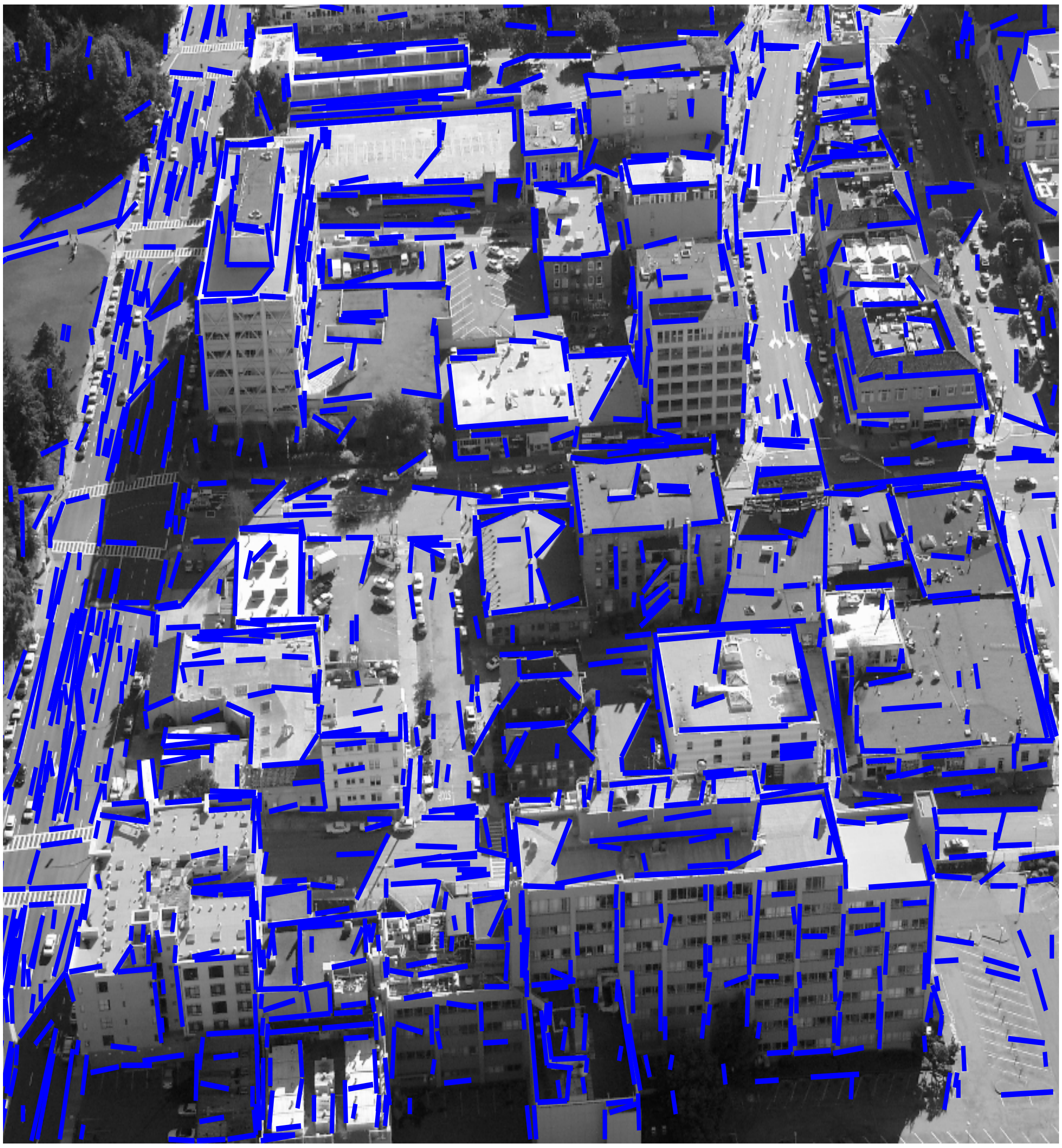


*Figure 2.2: Line segments extracted from a sample image*

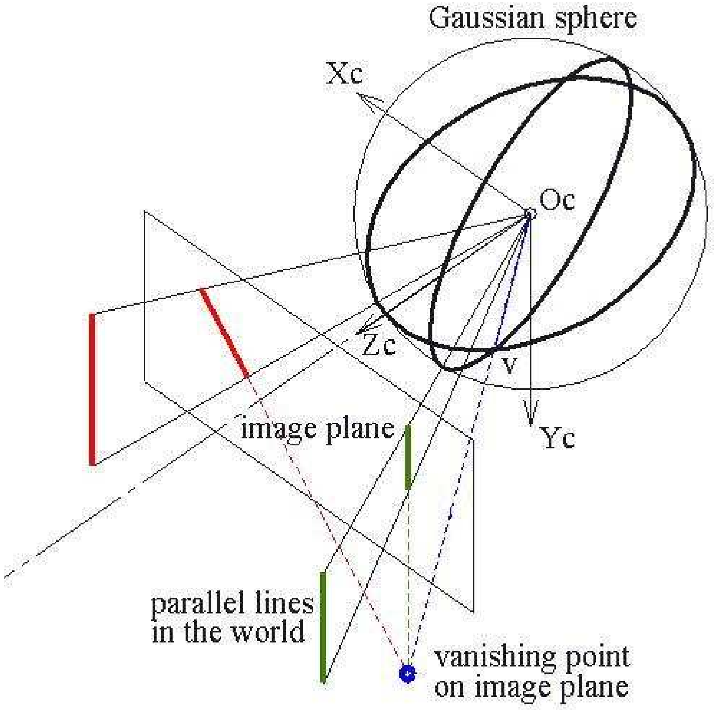


*Figure 2.3: Illustration of the formation of a vanishing point on the image plane using the Gaussian sphere approach: $l_1$ and $l_2$ are the projected parallel lines $L_1$ and $L_2$ in a 3D model respectively. A plane containing $l_1$ and $O_C$ intersecting the Gaussian sphere results in a circle $C_1$, and similarly, $l_2$ and $O_C$ result in a circle $C_2$. One of the intersections of the two circles facing the image plane points towards the vanishing point, v.*

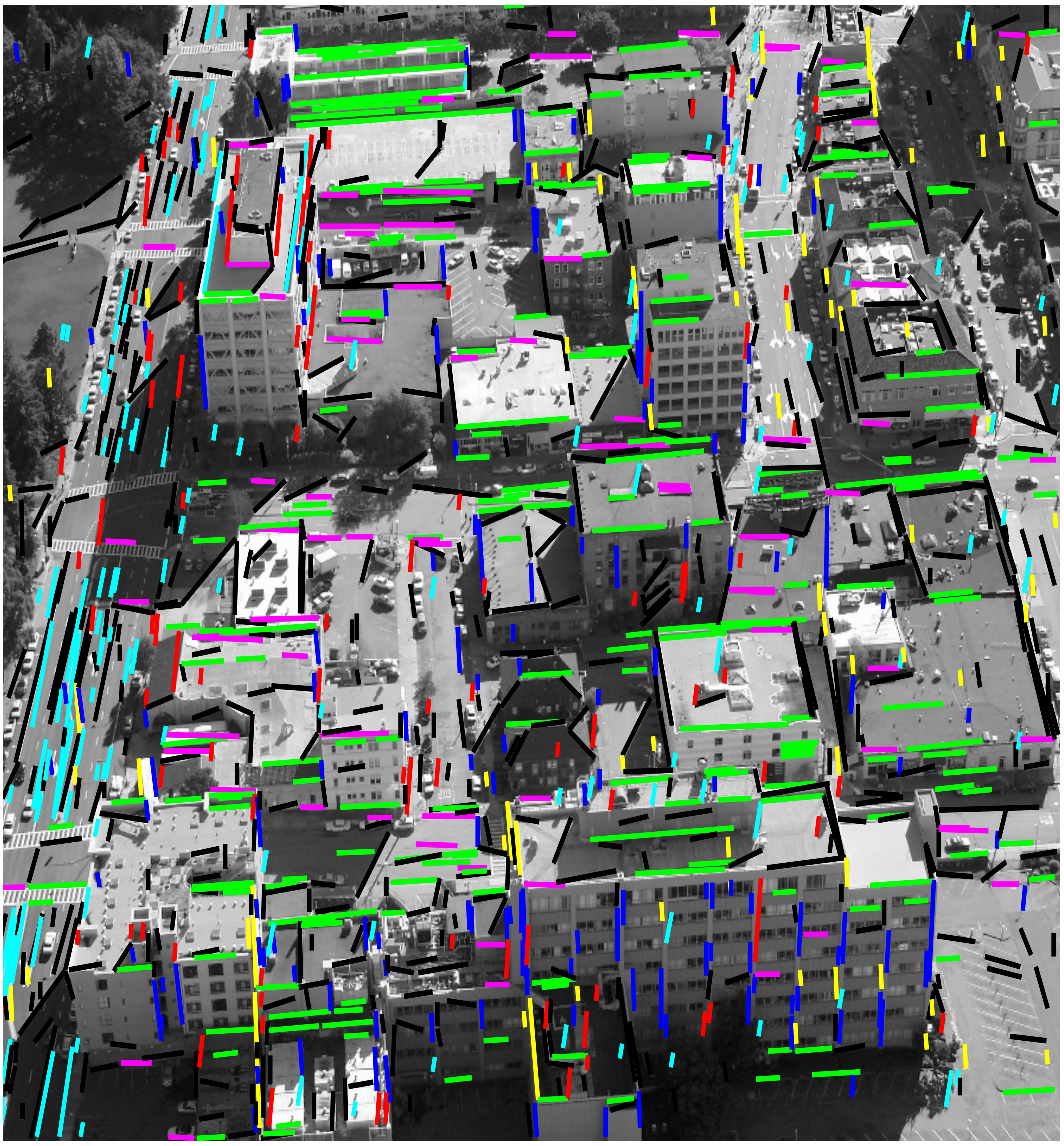


*Figure 2.4: Extracted line segments are colored according to their perspective vanishing points. The dark blue lines are the vertical lines from the vertical vanishing point detection.*

### 2.2.3 Camera parameter extraction from the vertical vanishing point

With the vertical vanishing point detected from the method above, the camera's pitch and roll angles can be estimated. Given that the vertical lines in the world reference frame are represented as $e_z = [0,0,1,0]^T$ in homogeneous coordinate, the vertical vanishing point, $v_z$ can be shown to assume the last column of **R** based on Equation (2.1), specifically:

$$\lambda v_z = [-sin\psi sin\theta, -cos\psi sin\theta, -cos\theta]^T \tag{2.3}$$

where $\lambda$ is a scaling factor. From (2.3), the pitch, roll angles and the scaling factor can be easily calculated. The pitch and roll angles derived from the vertical vanishing point are typically more accurate than NAV420CA readings, with error being less than one degree compared to three degrees from raw NAV420CA readings. A set of coarse estimates on all the camera parameters is therefore obtained. However, they are still not accurate enough for texture mapping. It is necessary to use feature point correspondence to refine the camera parameters, as discussed in the next chapter.

It is also possible to estimate the intrinsic camera parameters and rotation angles from three orthogonal vanishing points as in [32]. This method is included in Appendix A. However we have empirically observed that more accurate camera angles can be obtained from the vertical vanishing point with the intrinsic parameters known as a priori.

## 2.3 Non-vertical vanishing points detection

Although non-vertical vanishing points are not used for the coarse camera pose estimation, they are useful for determining 2DOCs on images as described in the next chapter. Our proposed non-vertical vanishing points detection algorithm is summarized in Fig. 2.5 with the following steps.

1. Bin all the line segments with a pre-determined bin width of $\frac{\pi}{20}$ according to their angles, except for the ones corresponding to the vertical vanishing point. Examine all the line segments in a bin with the highest frequency. Identify a seed vanishing point where most of the

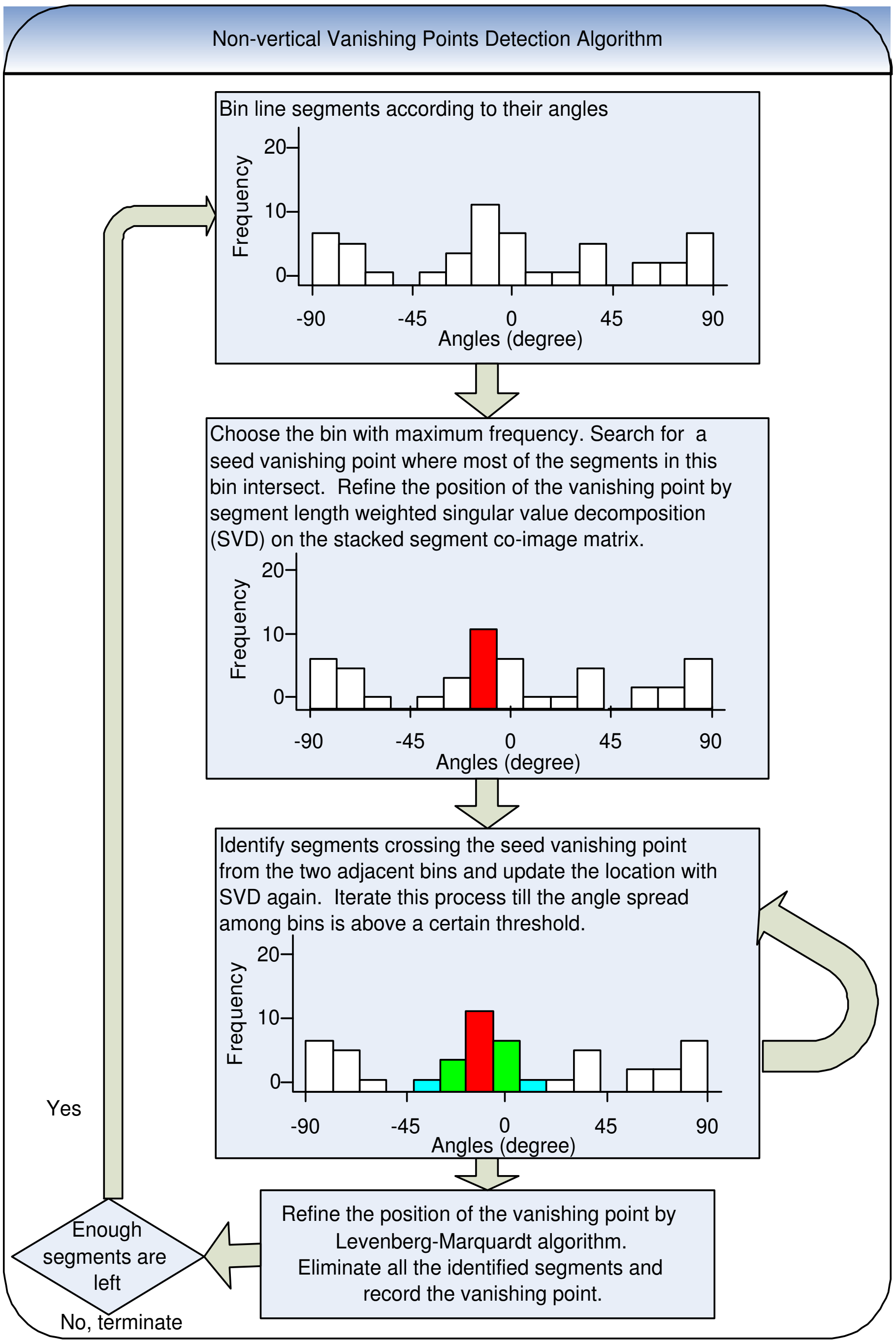

Non-vertical Vanishing Points Detection Algorithm
Bin line segments according to their angles
Frequency
20
10
0
-90
-45
0
45
90
Angles (degree)
Choose the bin with maximum frequency. Search for a seed vanishing point where most of the segments in this bin intersect. Refine the position of the vanishing point by segment length weighted singular value decomposition (SVD) on the stacked segment co-image matrix.
Frequency
20
10
0
-90
-45
0
45
90
Angles (degree)
Identify segments crossing the seed vanishing point from the two adjacent bins and update the location with SVD again. Iterate this process till the angle spread among bins is above a certain threshold.
Frequency
20
10
0
-90
-45
0
45
90
Angles (degree)
Refine the position of the vanishing point by Levenberg-Marquardt algorithm. Eliminate all the identified segments and record the vanishing point.
Enough segments are left
Yes
No, terminate

*Figure 2.5: Non-vertical vanishing point detection algorithm flow chart*

segments in that bin intersect. Although this can be performed with RANSAC, an exhaustive search is used since there are relatively few segments in each bin. The intersection of a pair of segments is first calculated as a seed vanishing point and the number of segments crossing near this intersection is computed. These segments are referred as inliers in contrast to outliers for the segments which do not cross. After an intersection with the largest number of inliers is identified, its location is refined by choosing the right singular vector with the least significant singular value of **WL** where **W** is a weighting square diagonal matrix with its diagonals as the lengths of the segments and **L** stores the co-images [1] of the segments in its rows.

2. Examine the segments in the two adjacent bins. If a line segment passes near the seed vanishing point from Step 1, it is included as an inlier. After this step, the location of the vanishing point is refined again using segment length weighted singular value decomposition as described in Step 1.

3. Repeat Step 2 by examining the segments in the next two adjacent bins until the angle spread across the two farthest bins for the vanishing point under consideration is above a certain threshold. In our application, we choose this threshold to be sixty degrees.

4. Refine the location of the vanishing point by minimizing the expression below:

$$\sum_i d_{i,1}^2 + d_{i,2}^2 \tag{2.4}$$

   where $d_{i,1}$ and $d_{i,2}$ are the distances from the two endpoints of $i$-th inlier segment to the closet line passing the vanishing point as shown in Fig. 2.6. This minimization is performed by Levenberg-Marquardt algorithm [42]. Output the resulting vanishing point and remove all the inlier segments. If there are still enough number of segments left, go back to Step 1.

The above proposed vanishing points detection algorithm has a few distinct advantages over the existing techniques. In our application, since objects are far away from a camera in aerial imaging, parallel lines in the 3D space appear nearly parallel on images. By initializing the seed vanishing point in a bin where all the segments share a similar angle, our algorithm avoids choosing a seed vanishing point which is an actual intersection in the 3D space. This preference is also

[1] Co-image is the cross product of two vectors from an origin to the two endpoints of a segment.

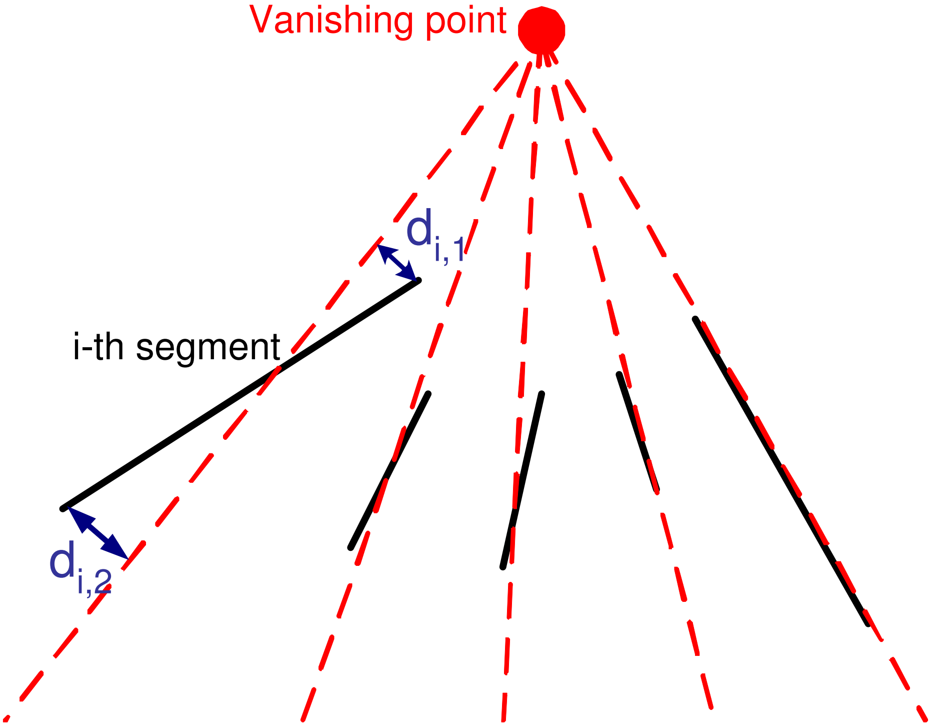


*Figure 2.6: Illustration of the distance metric used in Levenberg-Marquardt algorithm for the 4th step of non-vertical vanishing point detector*

reinforced by only considering segments whose slope angle difference is less than the angle spread threshold. Another advantage is that a segment length weighted singular value decomposition is used. It favors longer segments since they bear less uncertainty in their orientations. This idea is similar to the interpretation plane swatch model in [39]. Finally note that since inlier segments are eliminated after each iteration, the convergence of the algorithm is guaranteed without need of a priori knowledge on the number of the vanishing points. The result after this algorithm for the same sample image is shown in Fig. 2.4.

# Chapter 3

# Camera Parameters Refinement

In the previous chapter, we obtain a set of coarse extrinsic camera parameters from the instrument measurement and the vertical vanishing point. In this chapter, we develop a method to refine these parameters based on feature point correspondence for accurate texture mapping. A novel feature called 2DOC is introduced during this process. In our application, 2DOCs correspond to orthogonal structural corners where two orthogonal building contour lines intersect in the 3D space. First, 2DOCs are extracted from a DSM and aerial images, respectively. Then potential matches between DSM 2DOCs and image 2DOCs are generated. Since majority of the matches are erroneous, they are filtered out in two steps: Hough transform based on the most common rotation and GMSAC based on the Homography matrix with the least fitting error. The correct matches identified at the end are used in Lowe's camera pose recovery algorithm [37] to refine the camera parameters.

## 3.1 2DOC feature point

The feature points used in this paper are 2D orthogonal corners corresponding to orthogonal structural corners where two orthogonal building contour lines intersect in the 3D space. They are referred as 2DOCs throughout this thesis. Note that texture corners are inappropriate because our city model does not have texture. It is also extremely difficult to match structural corners from a city model to texture corners such as Harris corner [43] on an image since texture corners are ubiquitous resulting in high percentage of mismatches.

It might be intuitively appealing to use 3D corners where three orthogonal lines intersect. However it is difficult to identify sufficient number of 3D corners from images given the imperfect line segments extraction. Since the extracted building contours are typically incomplete due to occlusion, shadow and noise, it is unlikely that the endpoints of three lines are close to each other. Without sufficient number of 3D corners, it is impossible to perform Homography fitting. Therefore, we relax our constraint of three mutually orthogonal line segments to two. Naturally this leads to much more false structural corners. We will show in the rest of the chapter that these false corners can be eliminated by a unique feature descriptor, Hough transform and GMSAC. At the end, we will demonstrate that the price paid here is well compensated by greater number of correct structural corners.

### 3.1.1 2DOC extraction from digital surface model

DSM is a depth map representation of a 3D model obtained from LiDAR data. To obtain 2DOCs, building's contour needs to be extracted from a DSM. Although standard edge extraction algorithms from image processing can be applied, region growing approach based on thresholding on height difference is proven to be extremely efficient and accurate [23]. With a threshold on the height difference and the area size of a region, small isolated regions such as cars and trees are replaced with the ground level altitude, and objects on the rooftop such as signs and ventilation ducts are merged to the roof region. The outer contour of each region is then extracted [23].

Due to the resolution limitation of LiDAR data and inevitable noise, the contours have jittery edges even for straight structural lines of a building. To straighten jittery edges, we use Douglas-Peucker (DP) line simplification algorithm [44]. DP algorithm is a recursive algorithm which simplifies a chain of vertices from $p_0$ to $p_n$. It first links the endpoints $p_0$ and $p_n$ with a straight line $\overline{p_0 p_n}$. If the most deviating point away from $\overline{p_0 p_n}$, $p_i$ is above a certain distance threshold or a distance over segment length ($\overline{p_0 p_n}$) ratio threshold, $\overline{p_0 p_n}$ is broken into two pieces at $p_i$, and two lines, $\overline{p_0 p_i}$ and $\overline{p_i p_n}$ are used to represent the original chain. This process is then repeated iteratively for each line segment until the most deviated vertex is below the two deviation thresholds,

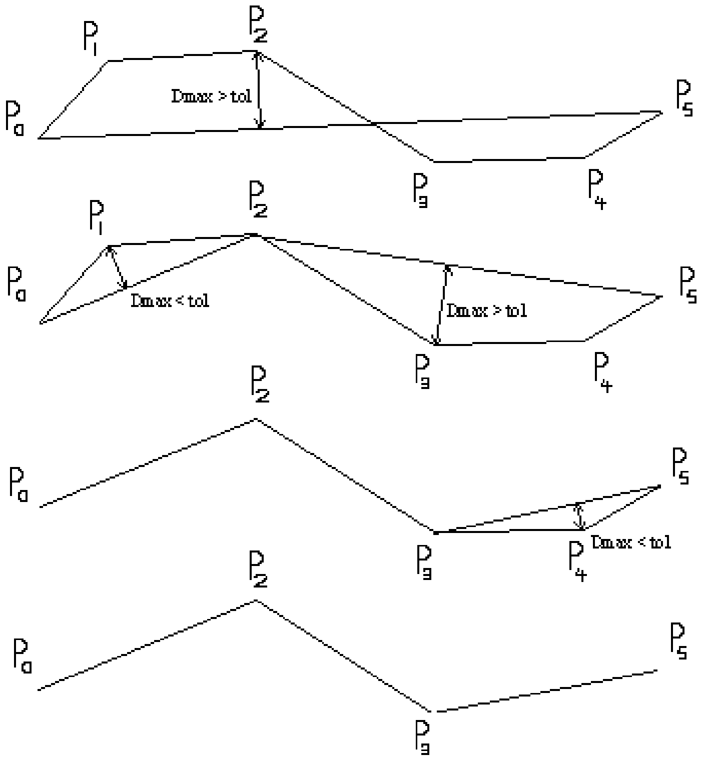


*Figure 3.1: Illustration of Douglas-Peucker line simplification algorithm.*

as shown in Fig. 3.1.

Since this algorithm is not designed for closed contours, it is necessary to break the region boundary into two open pieces. First, the origin is located at the centroid of a region. The farthest point away from the origin is identified as the first end point of the contour. Then the farthest point away from this point is selected to be the second end point. Once the contour is separated into two chains, DP algorithm is applied separately to the two chains. Finally, the two chains are joined back together. DP algorithm is not only easy to implement but also proven to be the best perceptual representation of the original chain from a psychological point of view [45]. What is more important is its intrinsic ability to preserve the positions of the 2DOCs, since real structural corners tend to be the extreme vertices in a contour.

Once the outer contour of each region is simplified, the vertices of the simplified contour are analyzed. A simple thresholding is performed on the lengths of the two adjacent line segments and on the intersection angle to identify 2DOCs under the assumption that structural corners are

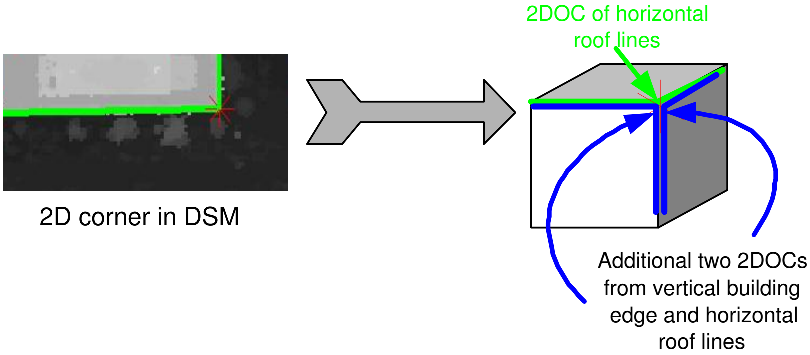
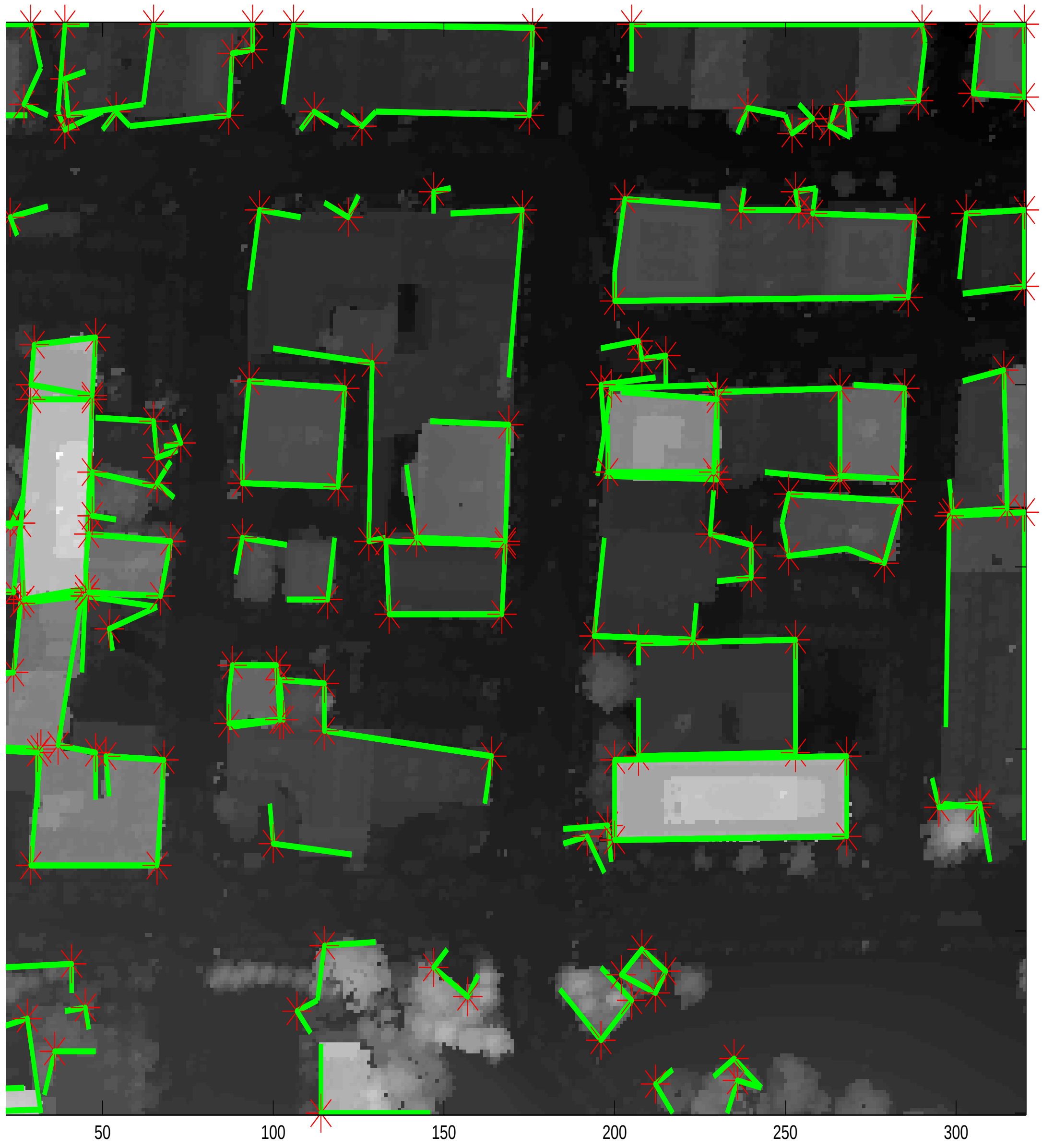

2DOC of horizontal roof lines
2D corner in DSM
Additional two 2DOCs from vertical building edge and horizontal roof lines

*Figure 3.2: Two more 2DOCs are augmented in a 3D model for each 2DOC detected in a DSM at the original location.*

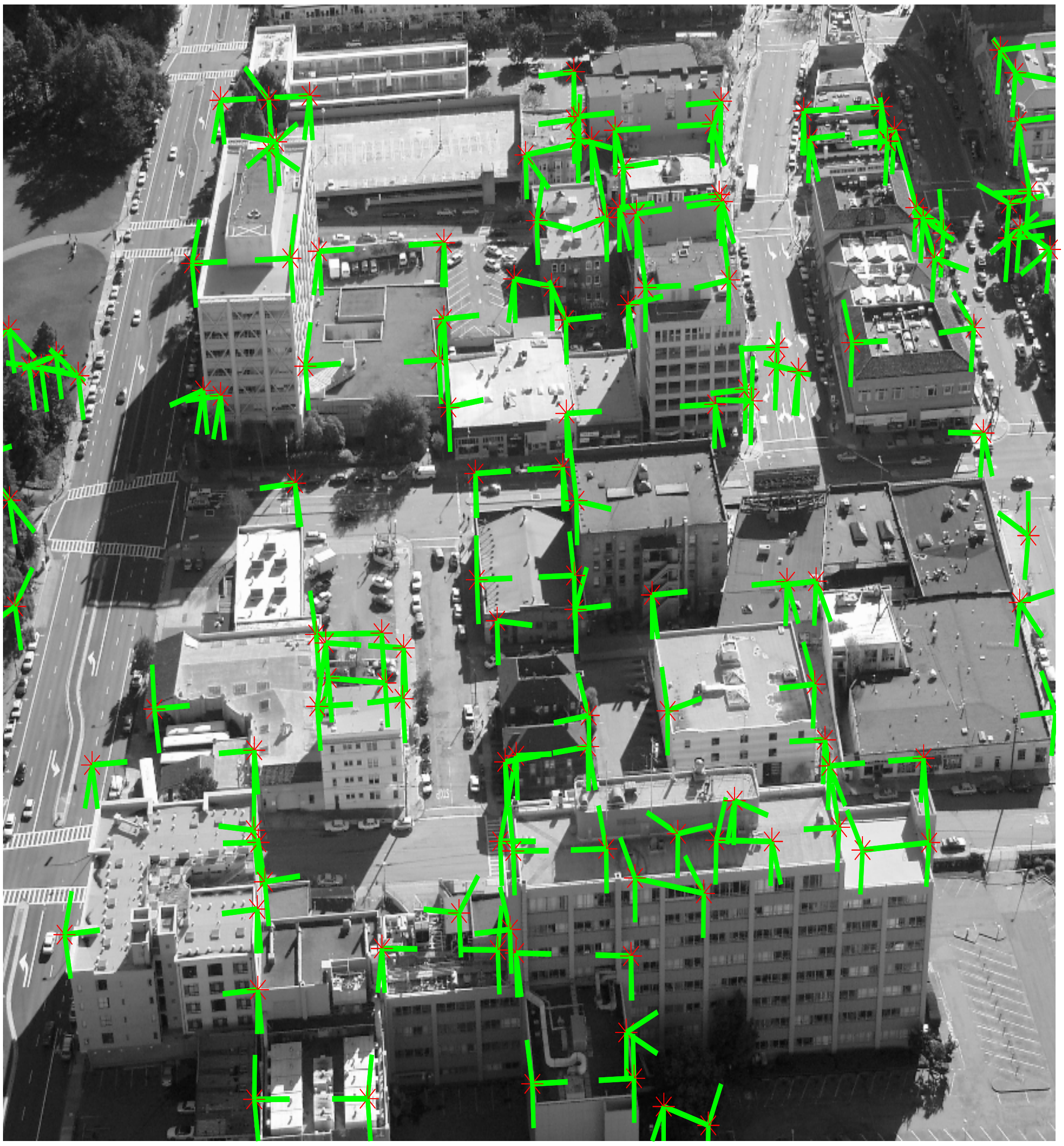
the intersections of two dominant roof lines with a $90^o$ intersection angle. Note that these 2DOCs are the intersections of two horizontal roof lines in the 3D space. Assuming no wall is slanted, two more 2DOCs are automatically generated at the same location, which are formed by the vertical building edge and the two horizontal roof lines as shown in Fig. 3.2. All these 2DOCs are then projected to the image plane using the coarse camera parameters obtained from the previous chapter. Fig. 3.3(a) shows the DSM corresponding to the previous image. The identified 2DOCs are superimposed on top of it. Their projected counterparts are shown on Fig. 3.3(b) with 1548 corners in total. Clearly, the projected 2DOCs do not match well with the building contours on the image due to the error in the coarse camera parameters.

### 3.1.2 2DOC extraction from aerial images

In general, it is difficult to infer 3D information from 2D images. However, it is possible to infer orthogonality information in the 3D space from the vanishing points obtained in the previous chapter. Since these vanishing points represent the directions of the corresponding groups of line segments in the 3D space, the orthogonality between two vectors from the camera origin to the vanishing points, $v_i$ and $v_j$ for instance, also implies the orthogonality between the two groups of line segments even though they do not appear to be orthogonal at all on images as demonstrated in Fig. 3.4.



(a)



(b)

*Figure 3.3: 2DOCs extracted from a DSM: (a)2DOCs on the DSM: red * are the 2DOCs and green lines are the two corresponding horizontal lines in the 3D space, (b)2DOCs projected on the image: red * are the 2DOCs and green lines are the two corresponding orthogonal lines.*

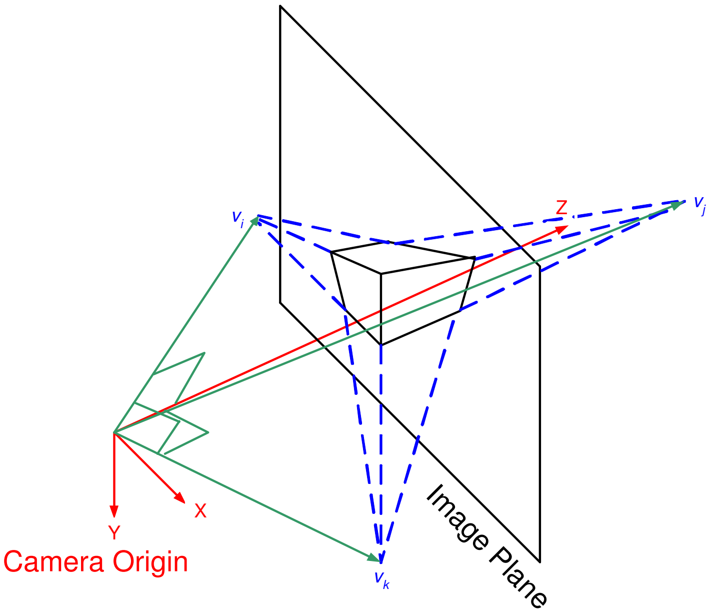


*Figure 3.4: Orthogonal vectors from the camera origin to the vanishing points indicate the orthogonality among the corresponding lines in the 3D space.*

From the vanishing points detected in the previous chapter, orthogonal pairs are first identified. The endpoints of every line segment belonging to a particular vanishing point, $v_i$ are then examined individually. If there is an endpoint of another line belonging to an orthogonal vanishing point, $v_j$, within a certain distance away, the midpoint of these two endpoints is identified as a 2DOC. Note that the intersection between the two line segments is not used because sometimes it can be far off from the actual intersection due to any inevitable slope angle error. This process is repeated for every line segment in every vanishing point group. The 2DOCs extracted from the sample image after this process are shown on Fig. 3.5 with 1099 corners in total.

## 3.2 2DOC putative matches generation

The 2DOCs from both DSM and an image, referred to as DSM 2DOCs and image 2DOCs respectively from now on, can be erroneous with no match from the other data source. These false detections are referred to as outliers. It is therefore necessary to implement a unique and simple descriptor based on geometric information to generate meaningful putative matches. Note that our

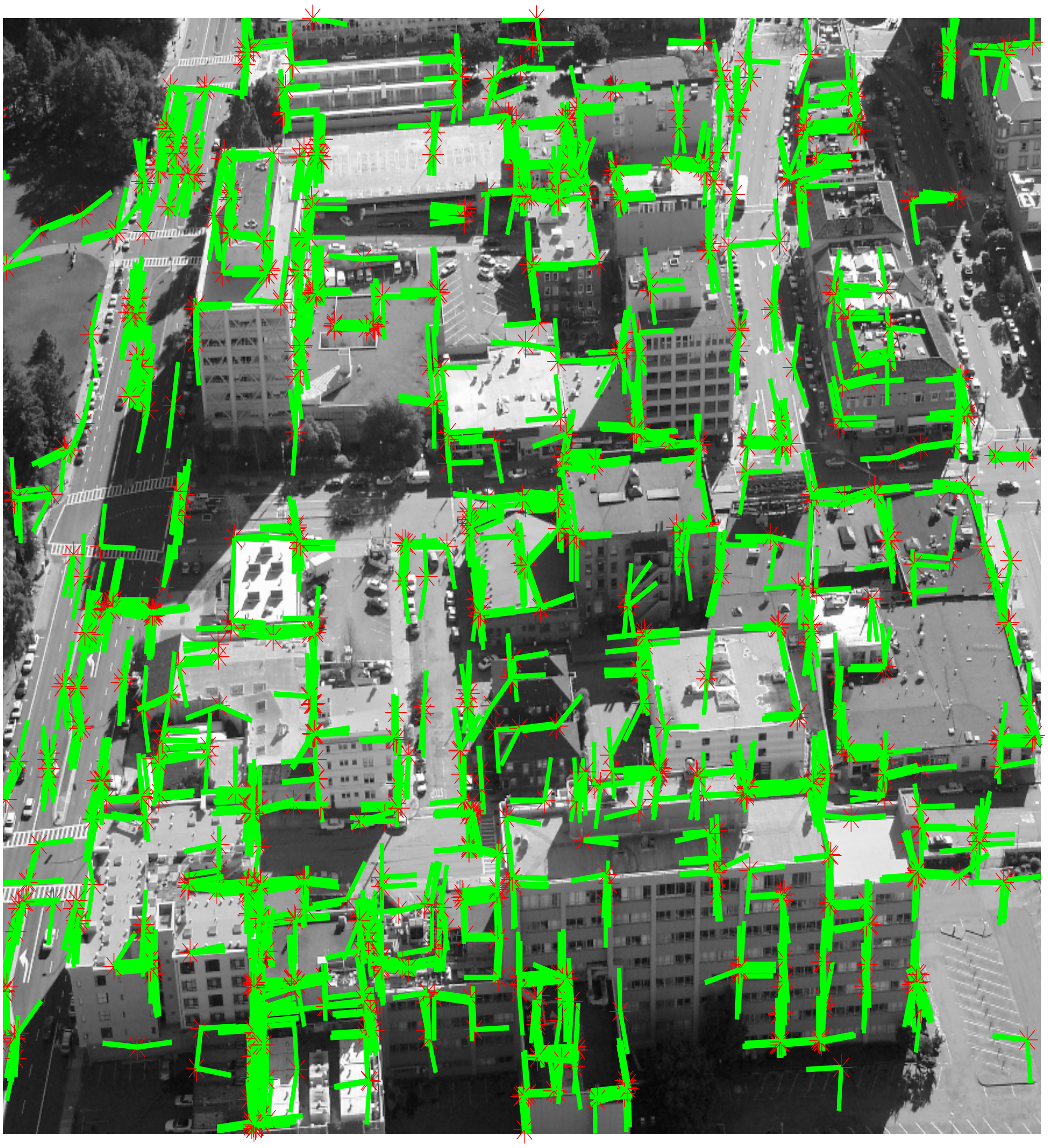


*Figure 3.5: 2DOCs extracted from an aerial image: red * are the 2DOCs and green lines are the two corresponding orthogonal lines.*

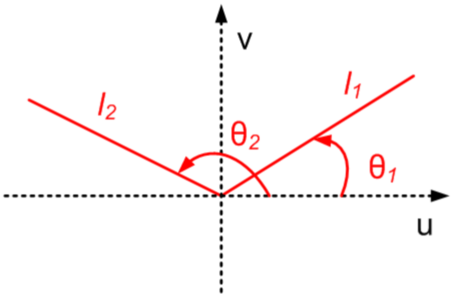


*Figure 3.6: 2DOC descriptor based on geometric information only.*

3D model does not have any texture information; thus, it is not possible to use many powerful feature descriptors developed in computer vision such as Scale-Invariant Feature Transform (SIFT) descriptor [46]. Hence we are forced to only use geometric information for 2DOC descriptor. The descriptor we propose in this thesis consists of the two intersecting lines' angles in the image plane as shown in Fig. 3.6, i.e. $\theta_1$ and $\theta_2$. To ensure that the order of the two angles is consistent, a fixed rule is followed where the relative angle between the two lines is in the range of 0 to $\pi$ counting counter-clockwise from the first line to the second one.

The feature descriptor for each 2DOC can now be examined to verify whether there are potential matches. If no good match is available, it is discarded as an outlier. The more outliers are eliminated at this step, the more efficient GMSAC algorithm becomes in future processing steps. In this putative match selection process, two selection criteria are applied.

The first criterion is search radius. With respect to each projected DSM 2DOC, $x^d$, all the image 2DOCs, $\{x^i\}$, within a certain search radius are examined. Note that the superscripts, $^d$ and $^i$ are used to indicate DSM and image 2DOCs respectively in the rest of the thesis, and $x$ is the coordinate on the image plane as defined in Chapter 2. The second criterion is Mahalanobis distance between the feature descriptors of $x^d$ and $x^i$, which is computed assuming a constant variance representing angle's measurement error:

$$d(x^d,x^i) = \sqrt{([\theta_1^d,\theta_2^d]-[\theta_1^i,\theta_2^i])\Sigma^{-1}([\theta_1^d,\theta_2^d]-[\theta_1^i,\theta_2^i])^T} \tag{3.1}$$

From the assumption of constant independent angle measurement error of three degrees (0.05 radian), $\Sigma$ is the predetermined covariance matrix, equal to $0.05I_{2\times2}$, where $I_{2\times2}$ is a $2\times2$ identity matrix. If the Mahalanobis distance is within a tolerance threshold, a putative match is created.

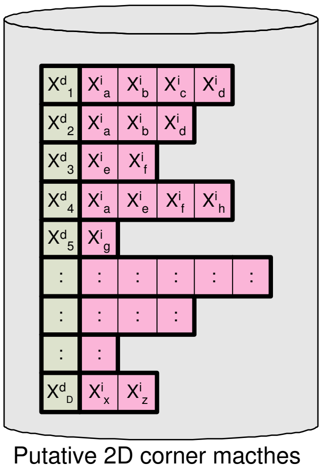

$X^d_1$ $X^i_a$ $X^i_b$ $X^i_c$ $X^i_d$
$X^d_2$ $X^i_a$ $X^i_b$ $X^i_d$
$X^d_3$ $X^i_e$ $X^i_f$
$X^d_4$ $X^i_a$ $X^i_e$ $X^i_f$ $X^i_h$
$X^d_5$ $X^i_g$
: : : : : :
: : : :
: :
$X^d_D$ $X^i_x$ $X^i_z$
Putative 2D corner macthes

*Figure 3.7: Putative match data structure where multiple image 2DOCs, $\{x^i\}$, are allowed for each projected DSM 2DOC, $x^d$*

Note that it is possible for a projected DSM 2DOC to have multiple image 2DOC matches. This reduces the possibility of missing correct matches since the shortest Mahalanobis distance does not necessarily indicate the correct match due to noise and intrinsic discrepancy between the two heterogenous data sources. This is in line with the observations in [47]. Furthermore, allowing multiple image 2DOC matches is especially important for repetitive structures such as Manhattan grid-structured city blocks where most of the buildings share similar orientations. The resulting putative match data structure is shown in Fig. 3.7, and these matches from the sample image are shown on Fig. 3.8.

## 3.3 Hough transform based on rotation

A large number of putative matches are typically generated from the previous step. For instance, our sample image results in 3750 matches, among which only 117 pairs are manually found to be correct matches or inliers. This creates an extremely large burden on GMSAC in the next stage since the inlier percentage is small. Furthermore, it is necessary to obtain four correct inliers at the same time in order to fit a Homography matrix, resulting in even larger number of required iterations to achieve the desired confidence level. Homography is assumed because the camera position error ($< 3m$) from the GPS is sufficiently small compared to the distance between the

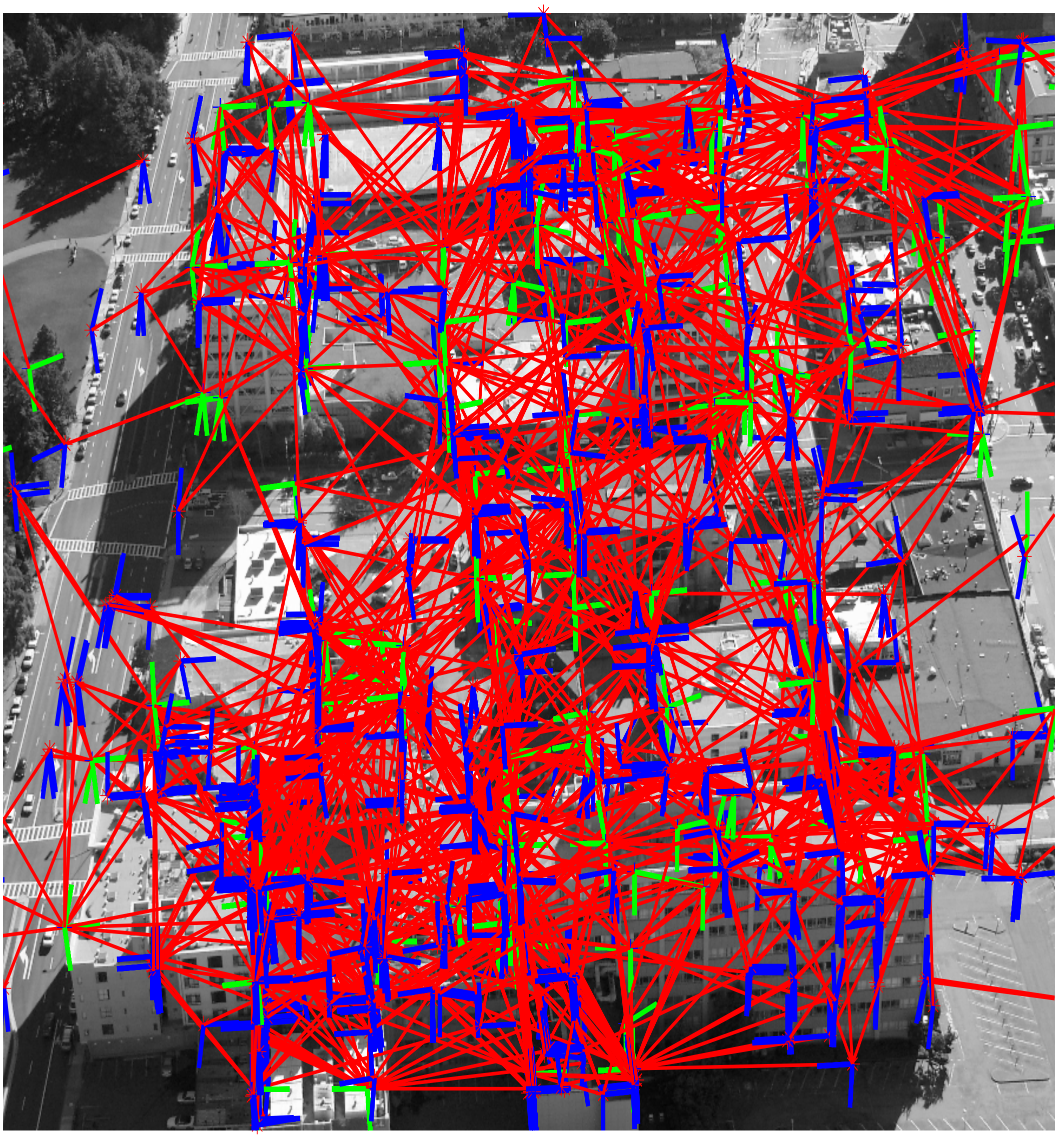


*Figure 3.8: 3750 putative matches found based on distance and feature descriptors' Mahalanobis distance thresholding: blue intersection is an image 2DOC, green intersection is a projected DSM 2DOC and red line indicates the correspondence.*

camera and the buildings on an image ($> 300m$). Thus, the difference between the projected DSM and image 2DOCs can be considered to be purely due to the camera rotation.

Let $p_{inliers}$ be the ratio of the inliers among all the putative matches, and $p_{conf}$ be the desired confidence level, then the required number of iterations in GMSAC, $N$, is [42]:

$$N = log(1 - p_{conf})/log(1 - p^4_{inliers}) \tag{3.2}$$

where the exponent of 4 on $p_{inliers}$ is from the fact that four correct matches need to be sampled at the same time to find the true Homographic matrix. With only 117 correct matches out of the 3750 in the sample image, the number of required iterations is nearly 3 million with 99% confidence level.

To mitigate this problem, Hough transform based on rotation is implemented. Before the actual explanation on the algorithm, several symbols need be defined. The meaning of these symbols is also pictorially presented in Fig. 3.9. Following the same convention where the superscripts, $^d$ and $^i$ are used to indicate DSM and image 2DOCs respectively, $x^i_m$ is the $m$-th image 2DOC in the image 2DOC series. Similary, $x^d_j$ are the $j$-th DSM 2DOC in the DSM 2DOC series. Furthermore, $X^d_j$ is $x^d_j$'s corresponding DSM 2DOC in the world reference frame, where $X$ is the coordinate in the 3D world space as defined in Chapter 2. Assuming the position error of the camera is insignificant, it can be immediately shown that the vector $v^i_m = [v^i_{x,m}, v^i_{y,m}, v^i_{z,m}]^T$ is the rotated vector $v^d_j = [v^d_{x,j}, v^d_{y,j}, v^d_{z,j}]^T$, if $x^i_m$ and $x^d_j$ are indeed the correct match. Here $v^i_m$ is from the camera center, $C$, to $x^i_m$ in the camera reference frame and $v^d_j$ is from the camera center, $C$, to $X^d_j$ in the world reference frame. For instance in Fig. 3.9(a), a common ration matrix exists between $(v^i_m, v^d_j)$ and $(v^i_n, v^d_k)$, since both sets of vector matches are correct. On the other hand in Fig. 3.9(b), since one set of vector match is correct and the other is incorrect, no common rotation exists between $(v^i_m, v^d_j)$ and $(v^i_n, v^d_l)$. Therefore one way to identify correct match is to find the common rotation matrix among pairs of matched vectors, and to eliminate vector pairs that do not satisfy the common rotation matrix. Here we implicity assume that the majority of the matched vectors are correctly paired and that incorrect matches are not dominant.

Since it is possible to find a rotation matrix from any pair of vectors, it is necessary to constrain the problem by finding a common rotation matrix among $N$ pairs of vectors which maximizes:

$$\sum_{j=1}^{N} \mathbf{R} v_j^d \cdot v_j^i \tag{3.3}$$

Here unit quaternion is used to represent rotation for its simplicity and closed form solution to the above problem. The next subsection gives a short summary from [48] on unit quaternion representation of rotation and the solution which maximizes (3.3). Readers are refereed to Appendix B or [49] for general quaternion definition and its operation.

### 3.3.1 Optimal unit quaternion solution

The most common representation of a rotation is an orthonormal matrix whose entries are nonlinear representations of three rotating angles. It is therefore difficult to maximize the expression in Equation (3.3) while enforcing orthonormal condition of the matrix without iterative methods. However, it will be shown that it is much easier to enforce the unit magnitude constraint of a unit quaternion and find a closed form solution for (3.3).

Any rotation can be characterized by a counterclockwise rotation angle $\theta$ with respect to an axis $a = [a_x, a_y, a_z]^T$ or by a unit quaternion $\dot{q}$:

$$\dot{q} = cos\frac{\theta}{2} + sin\frac{\theta}{2}(ia_x + ja_y + ka_z) \tag{3.4}$$

In other words, the imaginary part of the unit quaternion, $sin\frac{\theta}{2}(ia_x + ja_y + ka_z)$, represents the rotation axis and the real part, $cos\frac{\theta}{2}$, and the magnitude of the imaginary part represent the angle of rotation. The actual rotation by $\dot{q}$ of a vector $r = [r_x, r_y, r_z]^T$, represented in quaternion form $\dot{r} = 0 + (ir_x + jr_y + kr_z)$, is expressed as the composite product:

$$\dot{r}' = \dot{q}\dot{r}\dot{q}* \tag{3.5}$$

where

$$\dot{q}^* = cos\frac{\theta}{2} - sin\frac{\theta}{2}(ia_x + ja_y + ka_z) \tag{3.6}$$

Then (3.3) becomes:

$$\sum_{j=1}^{N} \dot{q}\dot{v}_j^d \dot{q}^* \cdot \dot{v}_j^i \tag{3.7}$$

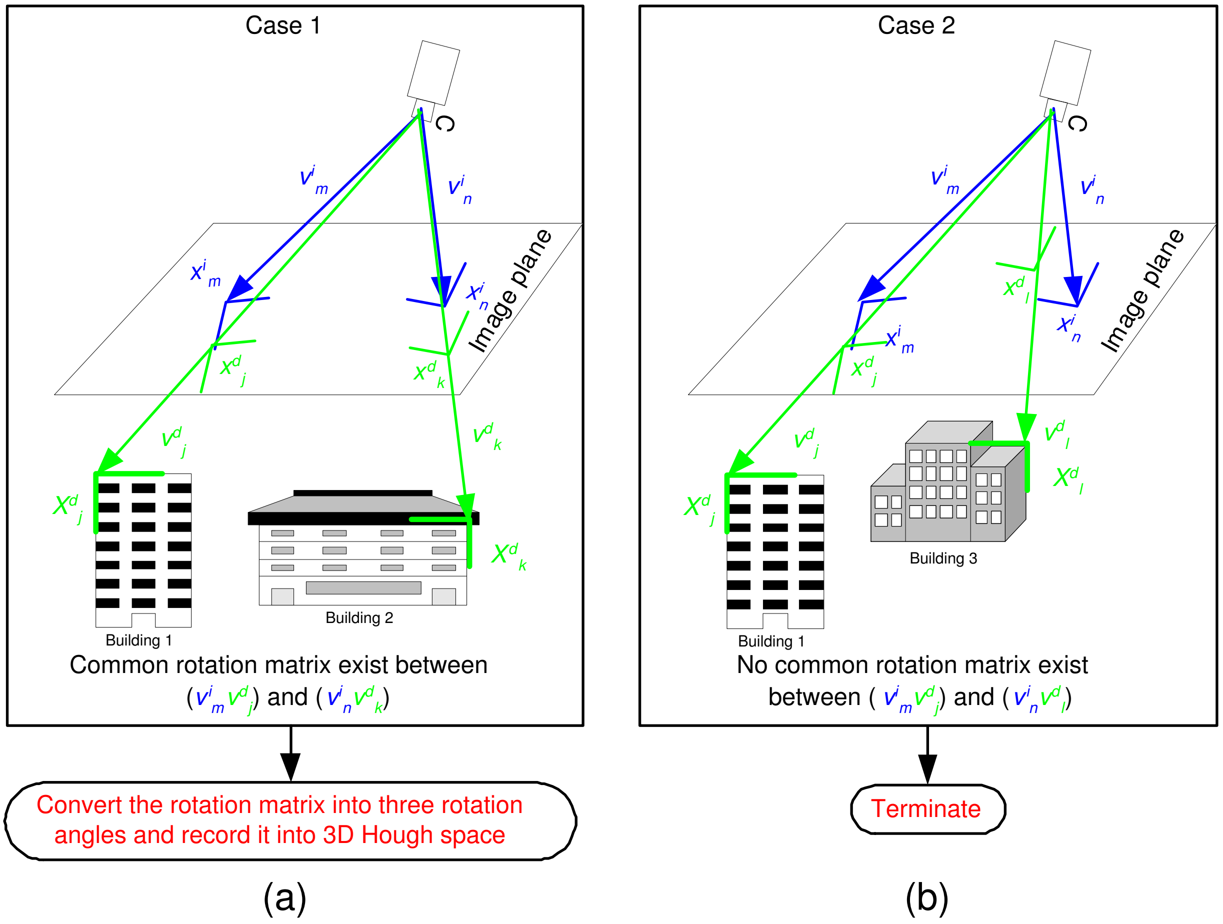

Case 1
C
$v^i_m$
$v^i_n$
Image plane
$x^i_m$
$x^i_n$
$x^d_j$
$x^d_k$
$v^d_j$
$v^d_k$
$X^d_j$
$X^d_k$
Building 1
Building 2
Common rotation matrix exist between
($v^i_m$,$v^d_j$) and ($v^i_n$,$v^d_k$)
Convert the rotation matrix into three rotation angles and record it into 3D Hough space
(a)
Case 2
C
$v^i_m$
$v^i_n$
Image plane
$x^d_l$
$x^i_n$
$x^i_m$
$x^d_j$
$v^d_j$
$v^d_l$
$X^d_l$
$X^d_j$
Building 3
Building 1
No common rotation matrix exist
between ( $v^i_m$,$v^d_j$) and ($v^i_n$,$v^d_l$)
Terminate
(b)

*Figure 3.9: Hough transform based on the common rotation for two pairs of corner correspondence: $v^i_m$ is from the camera center, C, to $x^i_m$ in the camera reference frame and $v^d_j$ is from the camera center, C, to $X^d_j$ in the world reference frame. $X^d_j$ is the corner in the 3D space corresponding to $x^d_j$ shown on the image plane; (a) a common rotation exists and it is recorded into the Hough space; (b) no common rotation exits and no further action is taken.*

After applying properties of quaternion, (3.7) becomes:

$$\begin{aligned}\sum_{j=1}^{N} \dot{q}\dot{v}_j^d\dot{q}*\cdot\dot{v}_j^i &= \sum_{j=1}^{N}(\dot{q}\dot{v}_j^d)\cdot(\dot{v}_j^i\dot{q}) \\ &= \sum_{j=1}^{N}(\bar{\mathbf{V}}_\mathbf{j}^\mathbf{d}\dot{q})\cdot(\mathbf{V}_\mathbf{j}^\mathbf{i}\dot{q}) \\ &= \sum_{j=1}^{N}\dot{q}^T\bar{\mathbf{V}}_\mathbf{j}^\mathbf{dT}\mathbf{V}_\mathbf{j}^\mathbf{i}\dot{q} \\ &= \dot{q}^T(\sum_{j=1}^{N}\bar{\mathbf{V}}_\mathbf{j}^\mathbf{dT}\mathbf{V}_\mathbf{j}^\mathbf{i})\dot{q} \qquad (3.8)\end{aligned}$$

where

$$\bar{\mathbf{V}}_\mathbf{j}^\mathbf{d} = \begin{bmatrix} 0 & -v_{x,j}^d & -v_{y,j}^d & -v_{z,j}^d \\ v_{x,j}^d & 0 & v_{z,j}^d & -v_{y,j}^d \\ v_{y,j}^d & -v_{z,j}^d & 0 & v_{x,j}^d \\ v_{z,j}^d & v_{y,j}^d & -v_{x,j}^d & 0 \end{bmatrix} \qquad (3.9)$$

and

$$\mathbf{V}_\mathbf{j}^\mathbf{i} = \begin{bmatrix} 0 & -v_{x,j}^i & -v_{y,j}^i & -v_{z,j}^i \\ v_{x,j}^i & 0 & -v_{z,j}^i & v_{y,j}^i \\ v_{y,j}^i & v_{z,j}^i & 0 & -v_{x,j}^i \\ v_{z,j}^i & -v_{y,j}^i & v_{x,j}^i & 0 \end{bmatrix} \qquad (3.10)$$

It is then obvious that the unit quaternion $\dot{q}$ which maximizes (3.3) and therefore (3.8), is the eigenvector corresponding to the most positive eigenvalue of $\sum_{j=1}^{N}\bar{\mathbf{V}}_\mathbf{j}^\mathbf{dT}\mathbf{V}_\mathbf{j}^\mathbf{i}$. And the value of the most positive eigenvalue indicates the goodness of fit of the common rotation.

### 3.3.2 Hough transform to screen out outliers

A 3D Hough space spanned by the range of angular measurement uncertainty of NAV420CA i.e. three degrees, is created for the three rotation angles. For computational efficiency, two pairs of vectors with each pair consisting of an image 2DOC matched to a DSM 2DOC, are exhaustively chosen to compute the unit quaternion. If the most positive eigenvalue is large indicating that there exists a common rotation with a good fit as shown in Fig. 3.9(a), the quaternion is decomposed into three rotation angles and accumulated into the corresponding Hough space. If the most positive eigenvalue is small indicating a poor fit as shown in Fig. 3.9(b), or the set of three angles from the unit quaternion is outside the Hough space, this unit quaternion is discarded. After the Hough transform, the most accumulated cell in the Hough space is chosen to be the correct set of rotation angles. All the matches that do not satisfy this rotation are then eliminated. This Hough transform typically results in much fewer putative matches. For instance, only 264 matches out of the 3750 matches are left on the sample image as shown on Fig. 3.10.

## 3.4 GMSAC based correct match identification

Feature matching from two data sources among outliers is encountered frequently in computer vision problems. The standard solution to this problem is random sample consensus (RANSAC) [50]. It has been shown that it is a robust technique for feature point, feature line matching to fit a Homography or a fundamental matrix [46]. However, RANSAC only allows for one match per extracted feature for each data source. In the putative match generation step, we have motivated the use of multiple matches from image 2DOCs for a projected DSM 2DOC in order to accommodate the heterogeneous sensor discrepancy and repetitive city structures.

Another deficiency of RANSAC is its hard decision rule to update the fitting matrix as observed by Torr and Zisserman [51]. The update in RANSAC is performed each time when more inliers are identified after comparing to a certain error tolerance threshold. Because of this hard decision process, even when there is a more accurate fitting matrix resulting in lower total fitting error, it is not updated if the number of inliers is smaller or equal to the less accurate one's. A soft decision such as M-estimator or maximum likelihood, which updates according to the overall fitting cost allows for continuous improvement [51].

Therefore, we propose two modifications to RANSAC for our specific application. This involves combining generalized RANSAC [47] and M-estimator Sample Consensus (MSAC) [51]. We name this newly developed algorithm generalized M-estimator Sample Consensus. The details of GMSAC are presented below and shown in flow chart in Fig. 3.11.

1. Uniformly sample four groups of DSM-image corner matches.

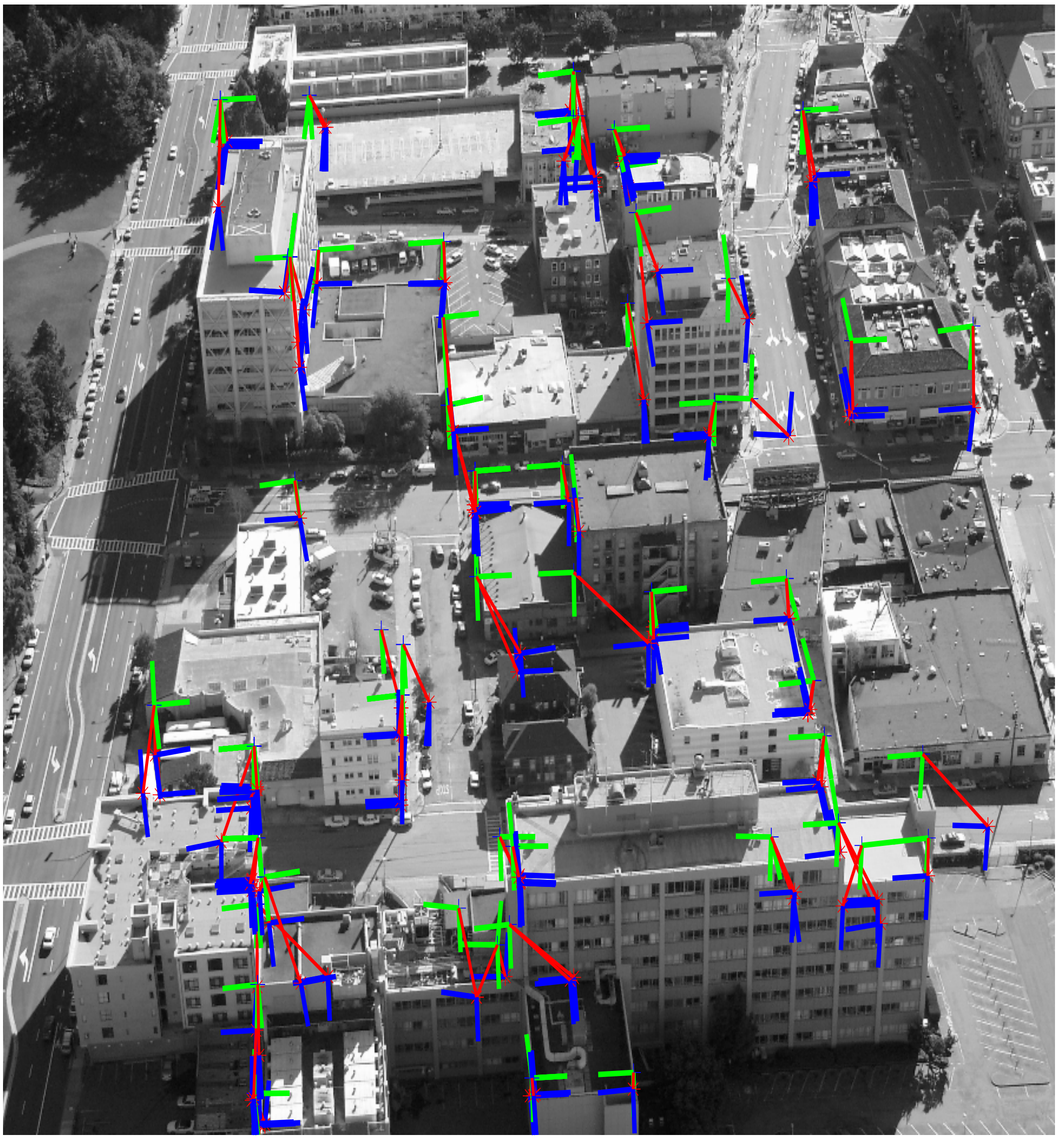


*Figure 3.10: 264 putative matches after the Hough transform: blue intersection is an image 2DOC, green intersection is a projected DSM 2DOC and red line indicates the correspondence.*

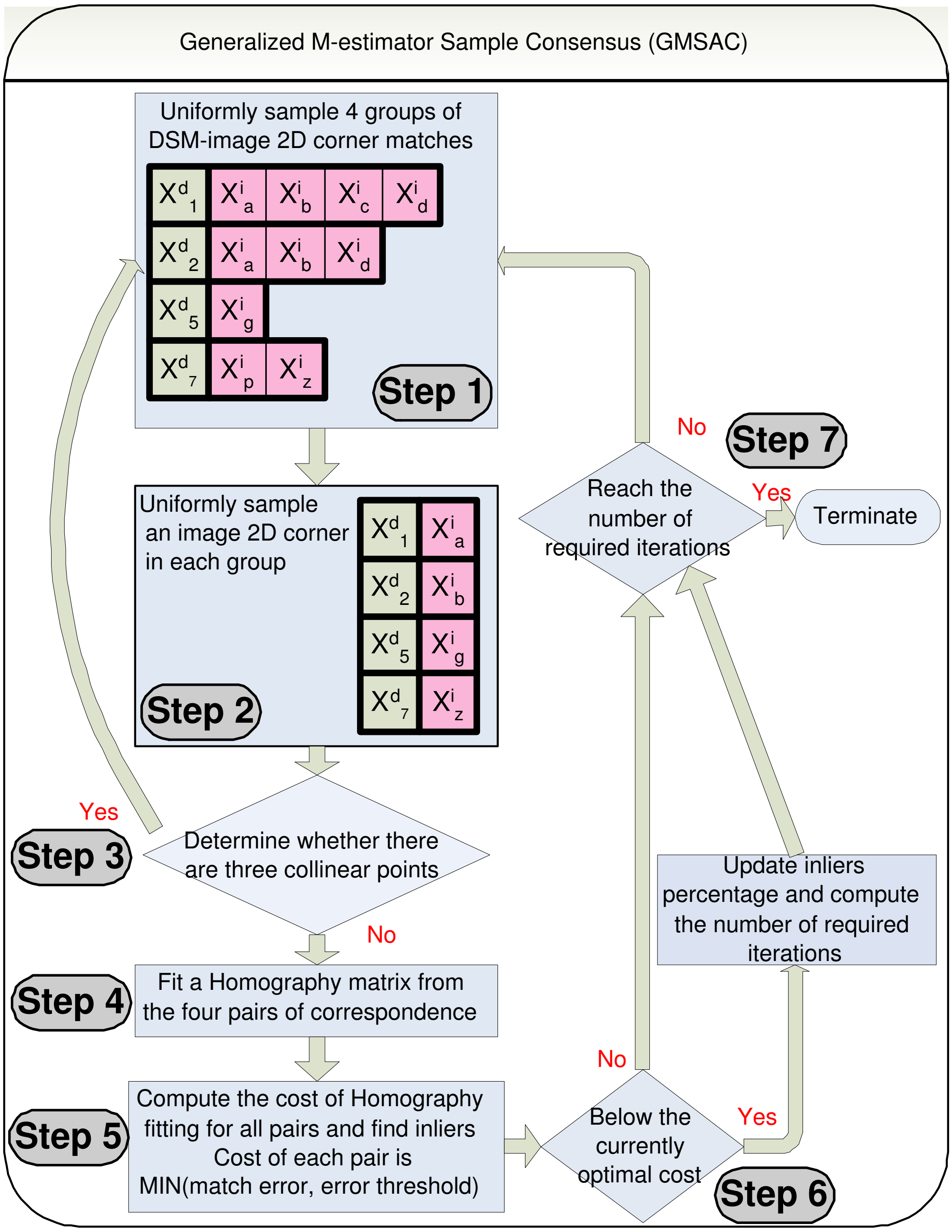

Generalized M-estimator Sample Consensus (GMSAC)
Uniformly sample 4 groups of DSM-image 2D corner matches
$X^d_1$ $X^i_a$ $X^i_b$ $X^i_c$ $X^i_d$
$X^d_2$ $X^i_a$ $X^i_b$ $X^i_d$
$X^d_5$ $X^i_g$
$X^d_7$ $X^i_p$ $X^i_z$
Step 1
Uniformly sample an image 2D corner in each group
$X^d_1$ $X^i_a$
$X^d_2$ $X^i_b$
$X^d_5$ $X^i_g$
$X^d_7$ $X^i_z$
Step 2
Yes
Step 3
Determine whether there are three collinear points
No
Step 4
Fit a Homography matrix from the four pairs of correspondence
Step 5
Compute the cost of Homography fitting for all pairs and find inliers Cost of each pair is MIN(match error, error threshold)
No
Below the currently optimal cost
Yes
Step 6
Update inliers percentage and compute the number of required iterations
No
Step 7
Reach the number of required iterations
Yes
Terminate

*Figure 3.11: Block diagram for GMSAC algorithm*

2. Inside each group, uniformly sample an image 2DOC.

3. Determine whether there are three collinear points, a degenerative case for Homography fitting. If so, go to Step 1.

4. With four pairs of DSM-image 2DOC matches, a 3X3 Homography matrix, **H**, is fitted with the least squared error [52]. Given $x^d = \lambda \mathbf{H} x^i$, where $\lambda$ is a scaling factor, a set of linear equations from the four pairs of matches can be formed. Based on the notation in Fig. 3.11 for instance, the following linear equations are formulated:

$$\begin{bmatrix} 0 & 0 & 0 & -u_1^d & -v_1^d & -1 & v_a^i u_1^d & v_a^i v_1^d & v_a^i \\ u_1^d & v_1^d & 1 & 0 & 0 & 0 & -u_a^i u_1^d & -u_a^i v_1^d & -u_a^i \\ -v_a^i u_1^d & -v_a^i v_1^d & -v_a^i & u_a^i u_1^d & u_a^i v_1^d & u_a^i & 0 & 0 & 0 \\ 0 & 0 & 0 & -u_2^d & -v_2^d & -1 & v_b^i u_2^d & v_b^i v_2^d & v_b^i \\ u_2^d & v_2^d & 1 & 0 & 0 & 0 & -u_b^i u_2^d & -u_b^i v_2^d & -u_b^i \\ -v_b^i u_2^d & -v_b^i v_2^d & -v_b^i & u_b^i u_2^d & u_b^i v_2^d & u_b^i & 0 & 0 & 0 \\ 0 & 0 & 0 & -u_5^d & -v_5^d & -1 & v_g^i u_5^d & v_g^i v_5^d & v_g^i \\ u_5^d & v_5^d & 1 & 0 & 0 & 0 & -u_g^i u_5^d & -u_g^i v_5^d & -u_g^i \\ -v_g^i u_5^d & -v_g^i v_5^d & -v_g^i & u_g^i u_5^d & u_g^i v_5^d & u_g^i & 0 & 0 & 0 \\ 0 & 0 & 0 & -u_7^d & -v_7^d & -1 & v_z^i u_7^d & v_z^i v_7^d & v_z^i \\ u_7^d & v_7^d & 1 & 0 & 0 & 0 & -u_z^i u_7^d & -u_z^i v_7^d & -u_z^i \\ -v_z^i u_7^d & -v_z^i v_7^d & -v_z^i & u_z^i u_7^d & u_z^i v_7^d & u_z^i & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} H_{11} \\ H_{12} \\ H_{13} \\ H_{21} \\ H_{22} \\ H_{23} \\ H_{31} \\ H_{32} \\ H_{33} \end{bmatrix} = 0 \quad (3.11)$$

where $[H_{11}H_{12}H_{13}H_{21}H_{22}H_{23}H_{31}H_{32}H_{33}]^T$ is the stacked column matrix of the rows of **H**. Recall that $x_i^d = [u_i^d, v_i^d]^T$ and $x_j^i = [u_j^i, v_j^i]^T$. The above linear system of equations can be solved by a variety of techniques including singular value decomposition. Specifically, the right singular vector with the least significant singular value is chosen to be the Homography matrix.

5. Every pair of DSM-image 2DOC match in every group is then examined with the computed Homography matrix from Step 4, where the sum of the squared deviation distances, $d_1$ and $d_2$ is computed. $d_1$ and $d_2$ are defined as:

$$d_1 = x^i - \lambda_1 \mathbf{H} x^d \quad (3.12)$$

and

$$d_2 = x^d - \lambda_2 \mathbf{H}^{-\mathbf{1}} x^i \quad (3.13)$$

with $\lambda_1$ and $\lambda_2$ as the necessary scaling factors. The cost of each match is the minimum of $\|d_1\|_2^2 + \|d_2\|_2^2$ and the error tolerance threshold. After performing analysis on all the pairs, the total number of the inliers with their $\|d_1\|_2^2 + \|d_2\|_2^2$ below a given error tolerance threshold and the sum of the costs from this particular Homography matrix are obtained.

6. If this overall cost is below the current minimum cost, the inlier percentage is updated and the number of required iterations to achieve the desired confidence level is recomputed from (3.2). Otherwise, another iteration is performed starting from Step 1.

7. Terminate the program if the required iteration number is exceeded.

There are 134 matches identified from GMSAC for the sample image as shown in Fig. 3.12. As seen, the DSM 2DOCs are projected very closely to image 2DOCs. Note that this result is obtained only after fewer than 100 iterations in contrast to 3 million due to the significantly higher inlier percentage after the Hough transform.

Finally Lowe's camera pose recovery algorithm [37] on all the identified corner correspondence pairs is used to obtain a more accurate set of extrinsic camera parameters with the fixed intrinsic parameters.

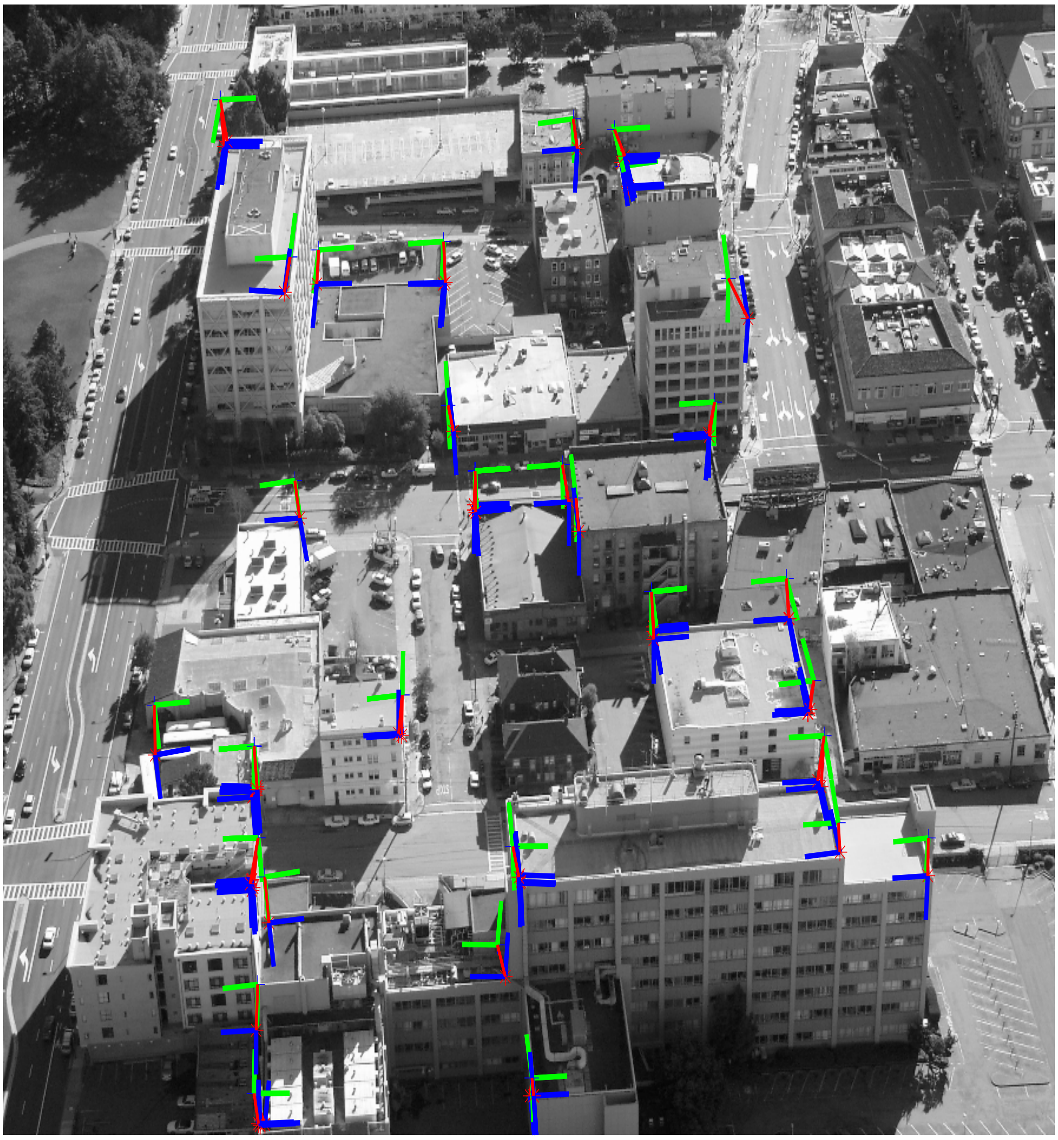


*Figure 3.12: 134 DSM-image matches after GMSAC: blue intersection is an image 2DOC, green intersection is a projected DSM 2DOC and red line indicates the correspondence.*

# Chapter 4

# Texture Mapping from Images to a 3D Model

In this chapter, we describe texture mapping from images to a 3D city model after a camera pose is computed. A more detailed presentation on the texture mapping methodology is included in [53].

## 4.1 Texture selection

The DSM model is first converted to a grided point cloud from which a triangular mesh is created. Since the resulting mesh typically contains excessive vertices redundancy especially on the flat surfaces, Qslim [54] is used to simplify the mesh to 3% of the original size.

Once a simplified triangular mesh model is obtained, texture information is then mapped to each 3D triangle in the mesh according to the camera pose. To avoid erroneous texture mapping due to occlusion and to choose the best available texture from multiple images for each triangle, the following texture selection process is applied.

### 4.1.1 Occlusion

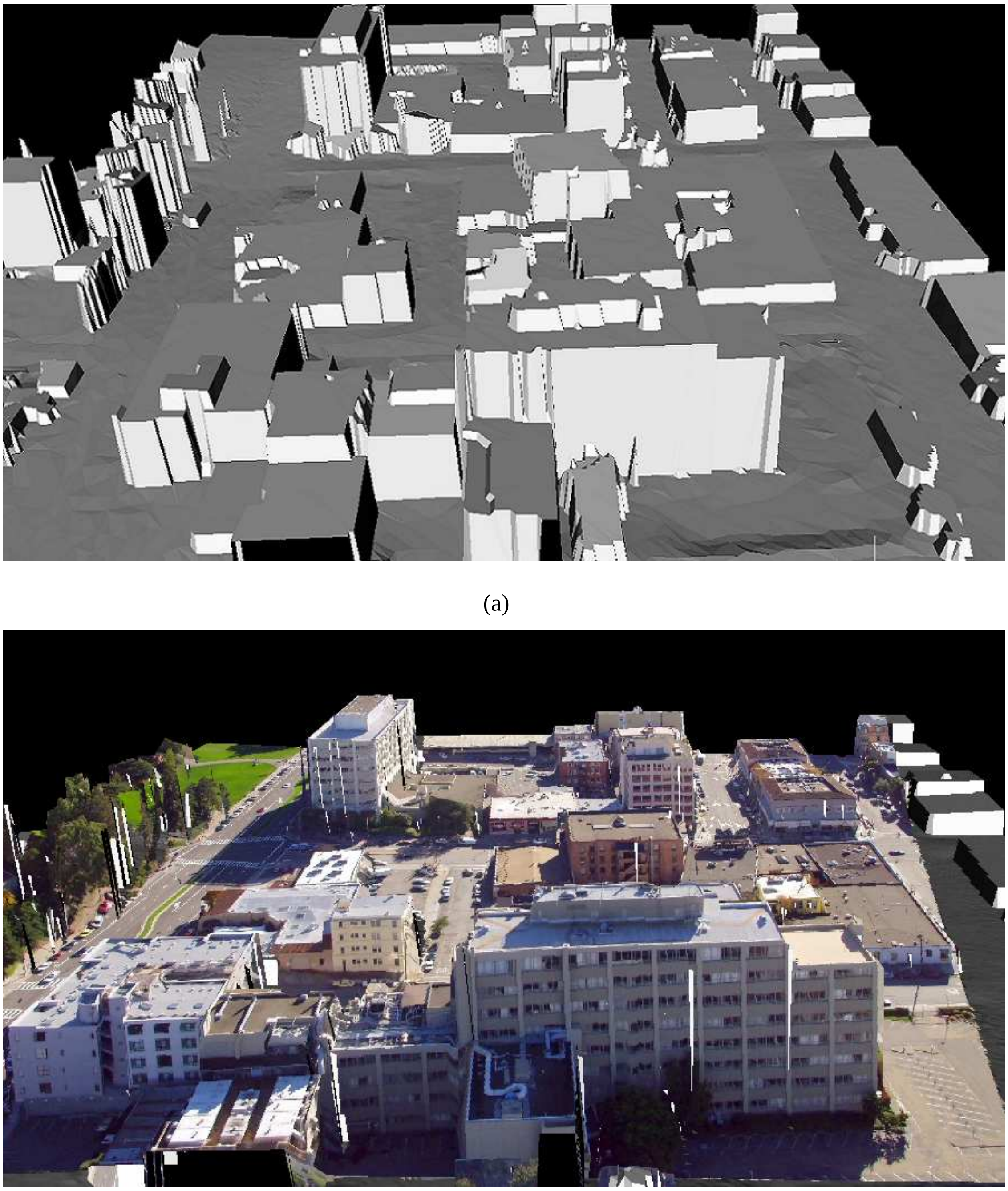
To determine whether a 3D triangle, $T_i$, is visible in an image $I_\alpha$, the z-buffer algorithm [55] is applied to detect occlusions for each pixel in the image. First, all the 3D triangles in a mesh are projected to the z-buffer, a 2D memory array, where the shortest distance between the mesh model and the camera for each pixel is continuously updated. Next, each 3D triangle is projected to the z-buffer again. For each pixel inside the projected triangular area, the distance between the 3D triangle and the camera is compared with the shortest distance recorded in the z-buffer. If the two distances are the same, it is verified that this pixel is indeed the projection of the current 3D triangle. Otherwise, it is considered as an occlusion. If there is no other image covering the whole triangle without occlusion, the following operation is performed. If the visible area is smaller than two pixels, the entire 3D triangle is not texture-mapped. Otherwise, the 3D triangle is split into smaller triangles where only the ones without occlusion are texture-mapped.

### 4.1.2 Viewing angle

For each 3D triangle, $T_i$, which can be projected onto multiple images, $\{I_\alpha\}$, without occlusion, texture selection based on viewing angle is performed. In this situation, it is desirable to choose the texture from an image with a normal view of the triangle. This is because the texture can be skewed too much resulting in unpleasant visual distortion after mapping images with extreme oblique viewing angles. Therefore, a scalar vector product is computed between the surface normal vector of $T_i$ and the view direction of each image in $\{I_\alpha\}$. The image with the greatest scalar product is then used to map $T_i$.

## 4.2 Texture-mapping on the example model and image

Based on the above texture selection criteria, the example image in the previous chapters is texture-mapped to the DSM model based on the refined camera parameters. The city models before and after the texture mapping are shown on Fig.4.1. The untextured parts are either due to lack of texture from the image or splitting of the occluded triangles.



(a)

(b)

*Figure 4.1: 3D city models before and after texture mapping from the example image: (a) City model before texture mapping, (b) City model after texture mapping.*

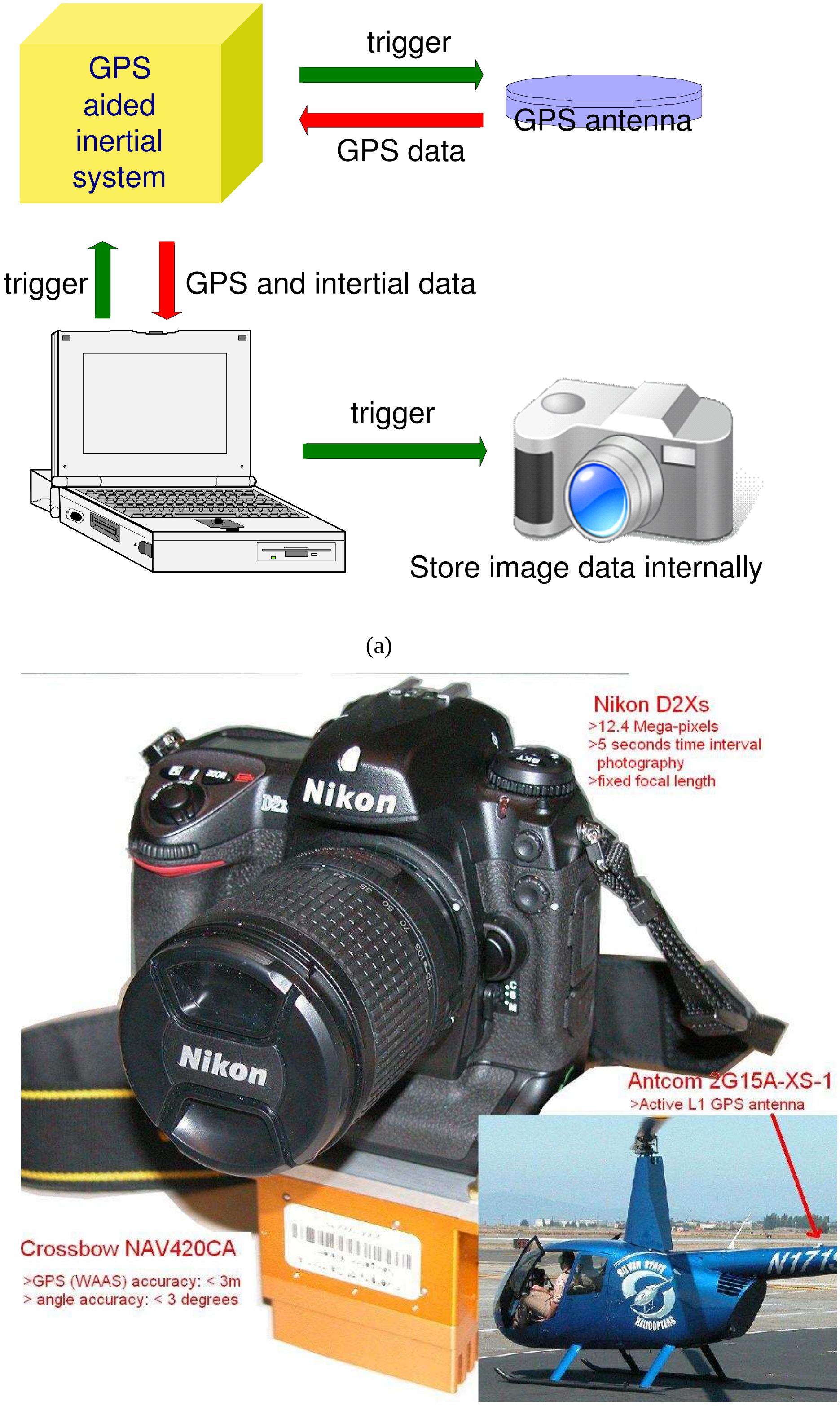
# Chapter 5

# Data Acquisition

The detailed data acquisition procedures are explained in this chapter. First, the necessary hardware is introduced. Then an optimal flight path and camera setup identified from the previous data acquisition experiences are specified in details.

## 5.1 Hardware specification

The GPS aided inertial system used in the data acquisition is NAV420CA from Crossbow with 2G15A active L1 GPS antenna from Antcom. A NAV420CA is mounted at the base of a Nikon D2Xs camera and the GPS antenna is mounted on top of the helicopter. See Fig. 5.1 for the data acquisition apparatus setup. Synchronized triggers are sent from a laptop to the camera and NAV420CA for data recording. The following accuracies in the measurements are expected from NAV420CA's specification sheet: location ($<$3m with DGPS(WAAS)) and rotation angle ($\pm 3^o$).



(a)

(b)

*Figure 5.1: Data acquisition equipment setup: (a) Block diagram of the data acquisition equipment setup, (b) Image of the actual acquisition equipment setup*

## 5.2 Flight path during the image acquisition

We acquired 358 aerial images using the system described above during a 42 minute helicopter flight over an area of 1.3km by 2.6km in the city of Berkeley, California. The helicopter flew at a constant altitude of approximately 280 meters above the ground, and followed a flight path as shown in Fig. 5.2 and 5.3, maintaining a constant speed of approximately 90 kilometers per hour. Fig. 5.2 shows the flight path to cover the entire city model from the LiDAR data and Fig. 5.3 shows the additional flight path over the downtown area.

## 5.3 Camera setup specification

The NAV420CA-camera unit is placed on the seat inside the helicopter cabin. It receives triggers from a computer to take pictures every five seconds. During the image acquisition, the camera is held by the co-pilot to maintain a pitch angle, $\theta$, of approximately 50 degrees and an image coverage area of two to three city blocks. From the previous two unsuccessful data acquisition trials, we have empirically found that our proposed algorithm is quite sensitive to the camera's pitch angle and image coverage area. If the pitch angle is too high as in the first trial, in other words, the camera is placed nearly horizontally, the image coverage area will be too large. This results in too many 2DOCs projected from the DSM to the image for DSM-image 2DOC correspondence, and presents extreme difficulty for the Hough transform and GMSAC to identify the correct matches. When the pitch angle is too small as in the second trial, top-down roof-top images are taken. Not only the vertical building edges on the images appear too short for vertical vanishing point detection, the mapped texture on the building facade becomes too distorted from such an extreme oblique angle. The image coverage area is also very important. As mention earlier, if the coverage area is too large, too many DSM 2DOCs are projected to the image, resulting in incorrect match identification. If the coverage area is too small, however, too few 2DOCs will be detected from the DSM. As discussed in the next chapter, the number of the DSM 2DOCs needs to be sufficiently large for robustness and accuracy. Finally, in order to accommodate the high speed of the flying helicopter, the shutter speed of the camera is set to be $\frac{1}{800}$ second, and the ISO sensitivity of the CCD sensor is set to be 800.

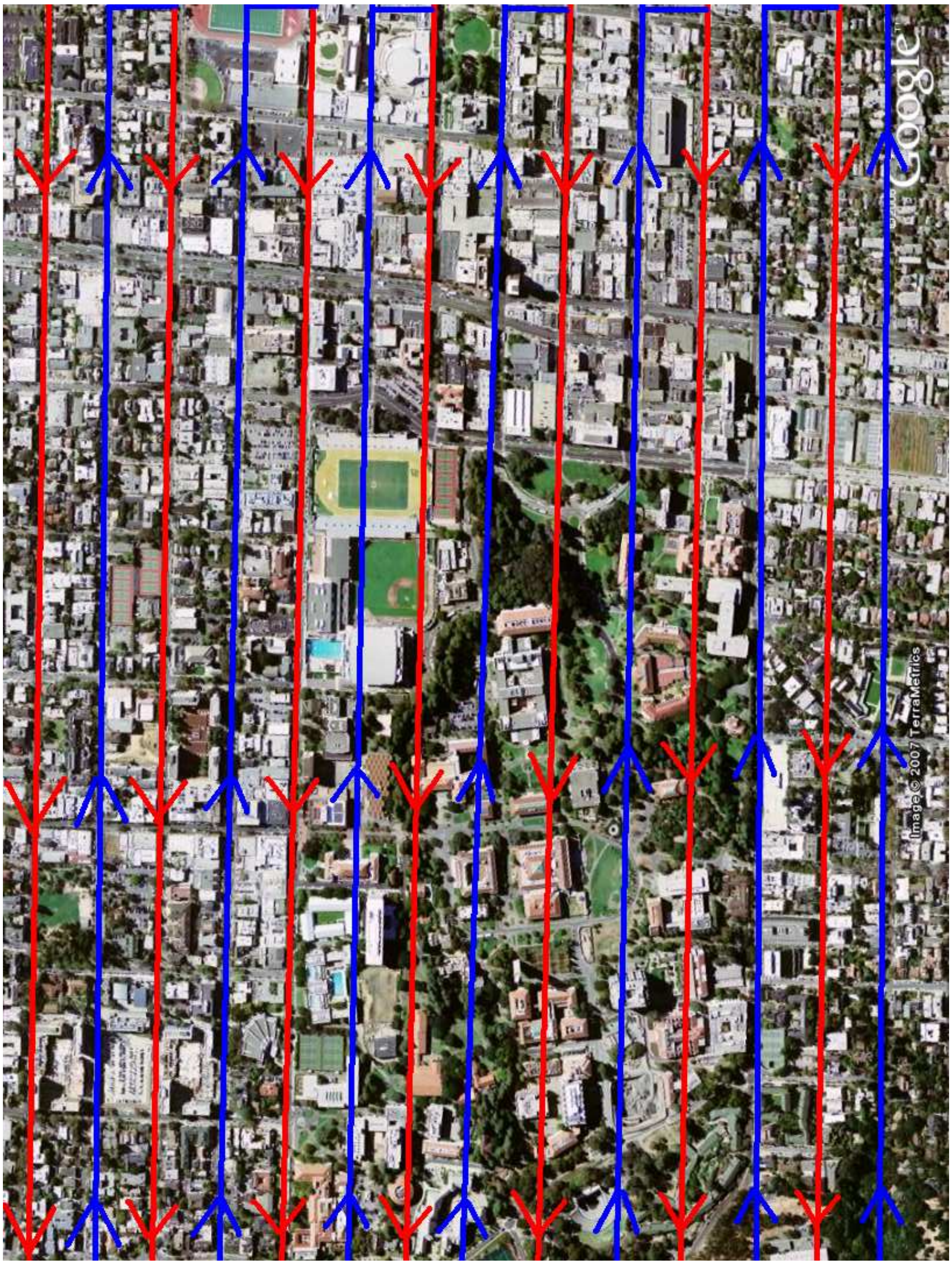


*Figure 5.2: Flight path over the entire city model: red lines are the flight paths from bottom to top, blue lines are the flight paths from bottom to top.*

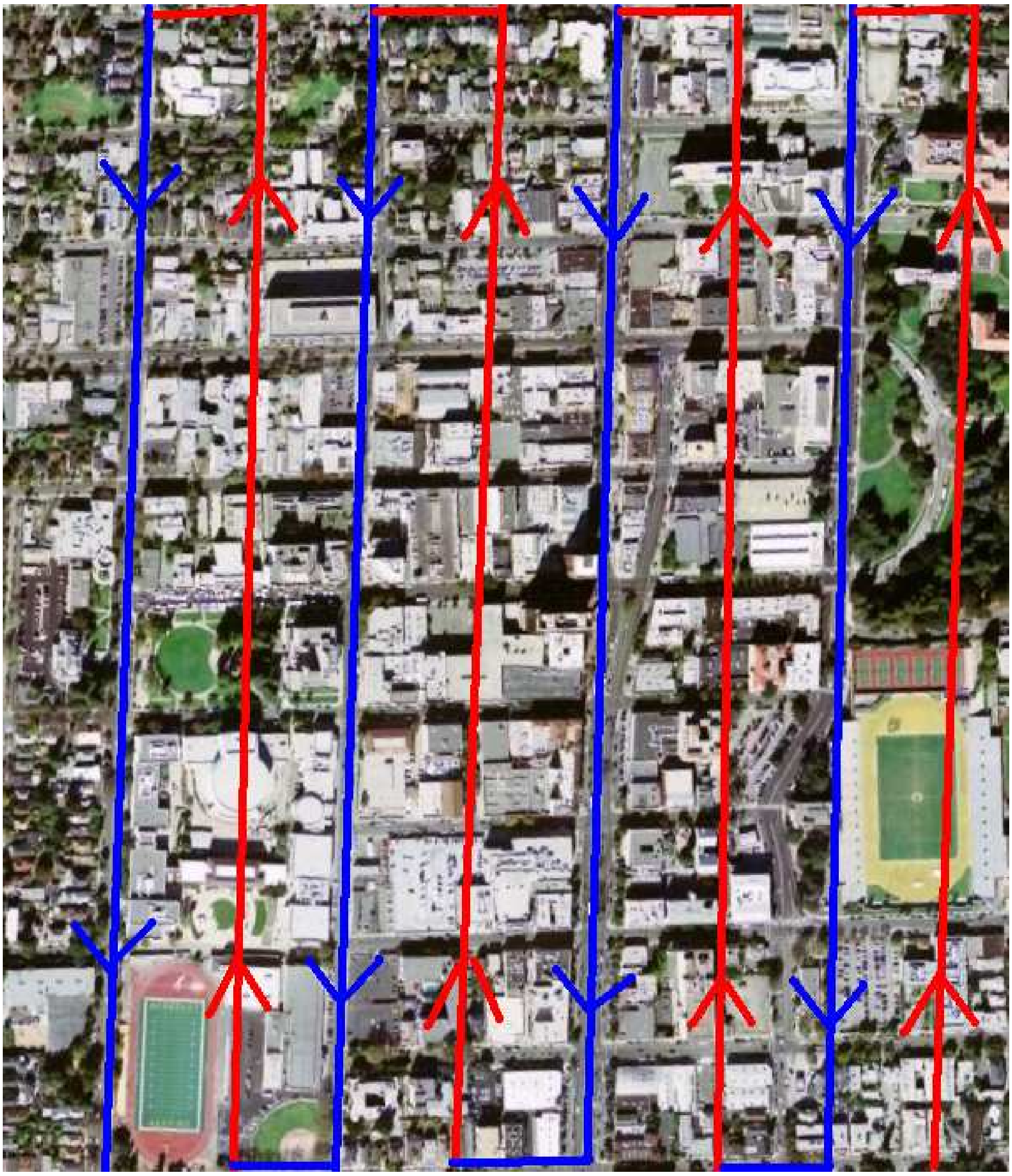


*Figure 5.3: Additional flight path over the downtown area: red lines are the flight paths from bottom to top, blue lines are the flight pathes from top to bottom.*

# Chapter 6

# Result Analysis

Raw data from NAV420CA is processed by first correcting the yaw angle bias since NAV420CA measures the magnetic north instead of the true north in the city model. This bias is compensated empirically by the mean of the differences between the NAV420CA yaw readings and the poses derived from manual point correspondence based on a subset of 50 images. This mean was empirically determined to be $5.42^o$. Secondly the GPS readings are corrected by taking into account the intrinsic delay inside the camera between the trigger time and the actual time a picture is taken. In order to correct the GPS readings, the camera delay is estimated empirically. Specifically, the differences in the camera's translation parameters between NAV420CA readings and the poses from manual point correspondence are obtained for 50 images acquired during the straight flight paths shown in Fig.5.2 and 5.3. Under constant velocity assumption for the duration of camera delay, the distance differences can be assumed to be the product between the delay and the 3D velocities recorded by NA420CA. Therefore, with the differences in translation parameters and the recorded velocities from 50 images, the mean of 1.0 second of the delay for the camera is obtained. Under the same assumption of constant velocity, all the translation parameters $[T^x, T^y, T^z]^T$ are then corrected by adding the product between this empirical delay and the velocity measurements from NAV420CA. The typically adjusted distance is about $20m$ with average speed of 90 kilometers per hour in our acquisition process.

To systematically evaluate the proposed system, the coverage area is divided into three regions with different characteristics by the red lines shown in Fig. 6.1. The first region is the downtown

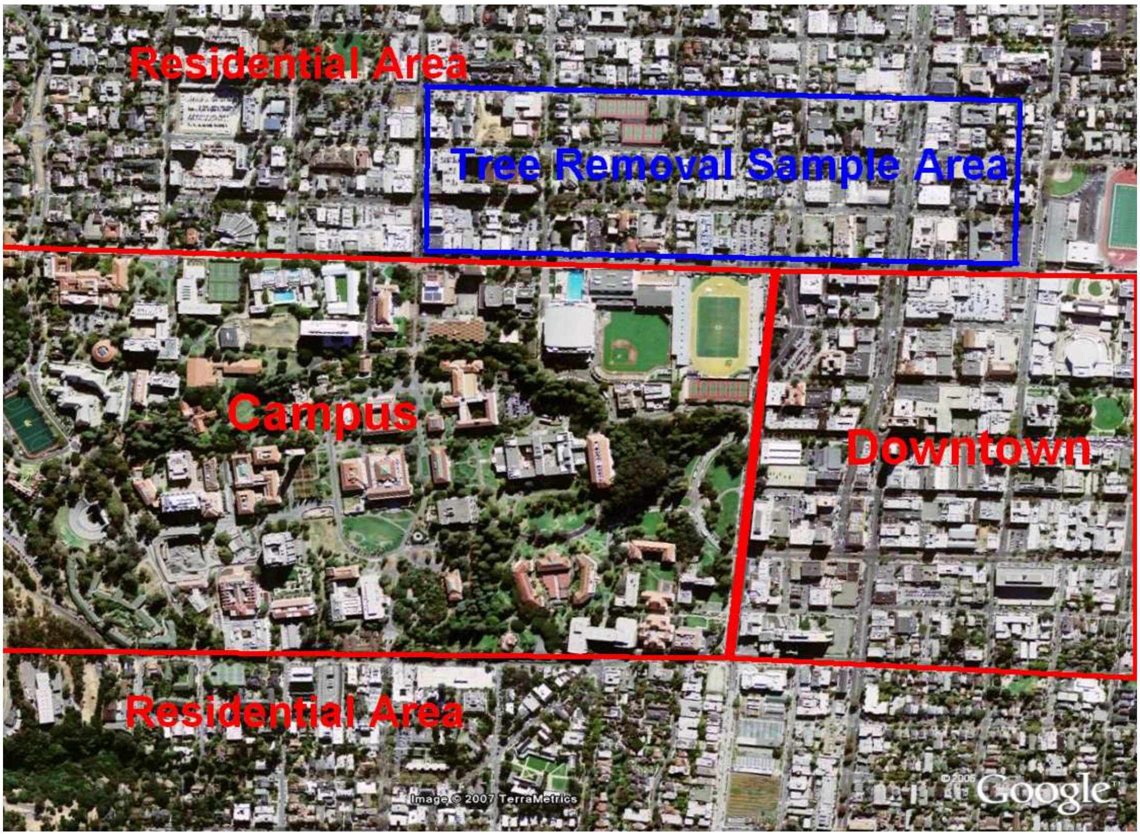


*Figure 6.1: Aerial images coverage area over Berkeley and region classification*

district, where large buildings are densely packed among few trees. The second region is Berkeley campus where large buildings are sparsely distributed among dense trees and vegetation. The rest of the area is grouped as residential area where much smaller houses are densely packed among dense trees. We use three methods to evaluate our algorithm: (a) visual inspection by projecting DSM lines to the building contours on the image, (b) comparison of camera parameters with the ones from manual point correspondence, (c) visual inspection of the textured 3D model based on the recovered camera poses.

## 6.1 Camera pose rating by visual inspection on the projected DSM lines

The correctness of the recovered camera pose is validated visually by examining the quality of the fit between the projected DSM lines and the building contours on the image. When two sets of lines align sufficiently close to each other, the recovered pose is deemed to be correct for texture mapping. In case the pose is not sufficiently accurate, the source of error is investigated by examining the extracted 2DOCs both from the image and the DSM. The sources of error are classified into two:

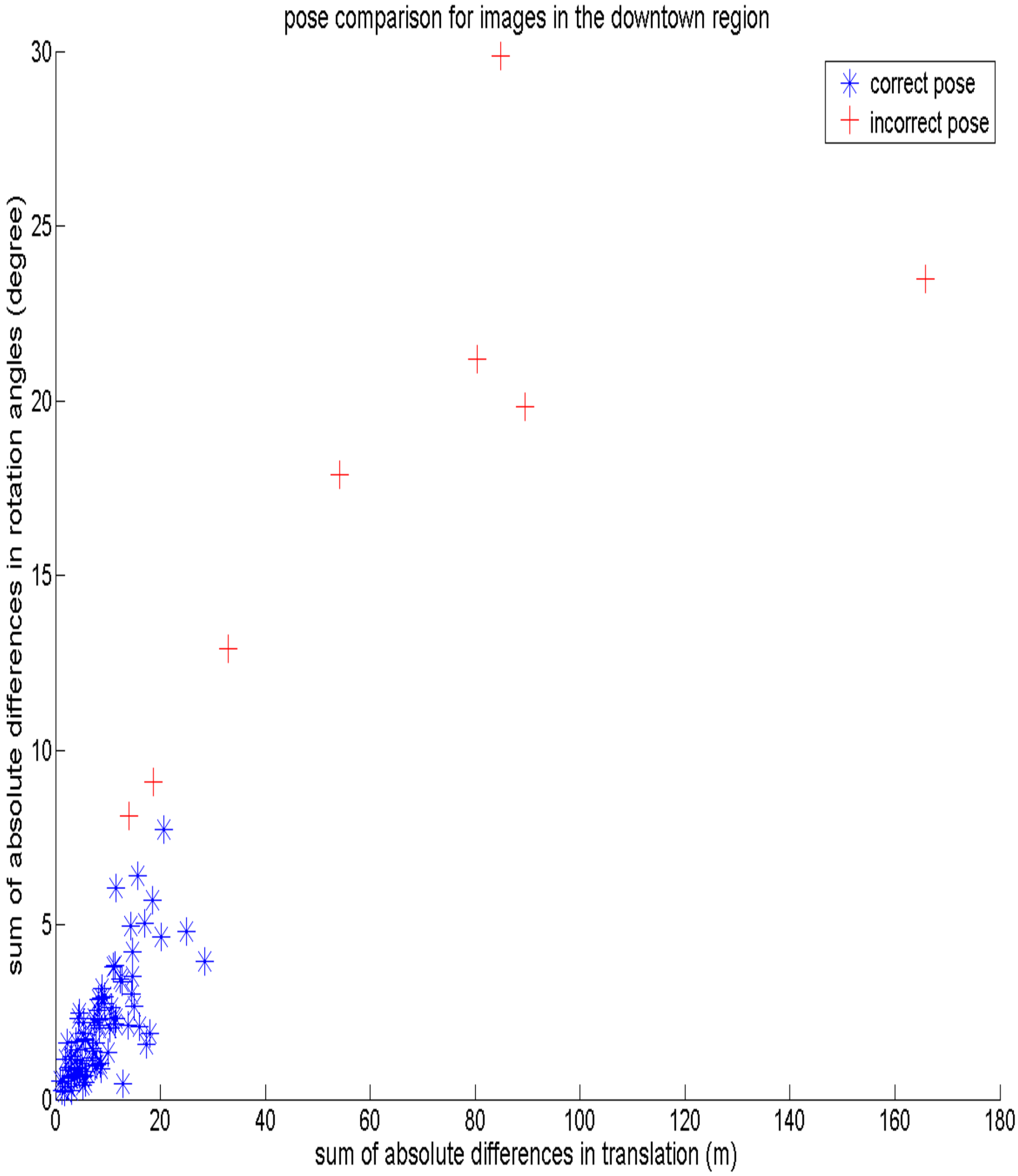
1. Too few corners: If not enough correct 2DOC matches are available, the system has no chance of finding the right camera pose. This happens when not enough correct 2DOCs are extracted from the image or the DSM or both.
2. Multiple solutions to an ill-posed problem: Even with sufficient number of correct 2DOC matches, the set of angles with the maximum frequency after the Hough transform might not be the true camera rotation. Similarly, the Homography model identified from GMSAC might not correspond to the true camera pose. It is possible that some combination of 2DOC matches accidentally yields a random camera pose with a small Homography fitting error. This is due to the ill conditioned nature of the problem. Both the Hough transform and GMSAC are based on the assumption that the pose with the maximum consensus is the true camera pose. When this assumption breaks down, the accuracy of the recovered pose is not guaranteed. This often happens when $p_{inliers}$ is significantly low.

## 6.2 Verification with camera poses from manual correspondence

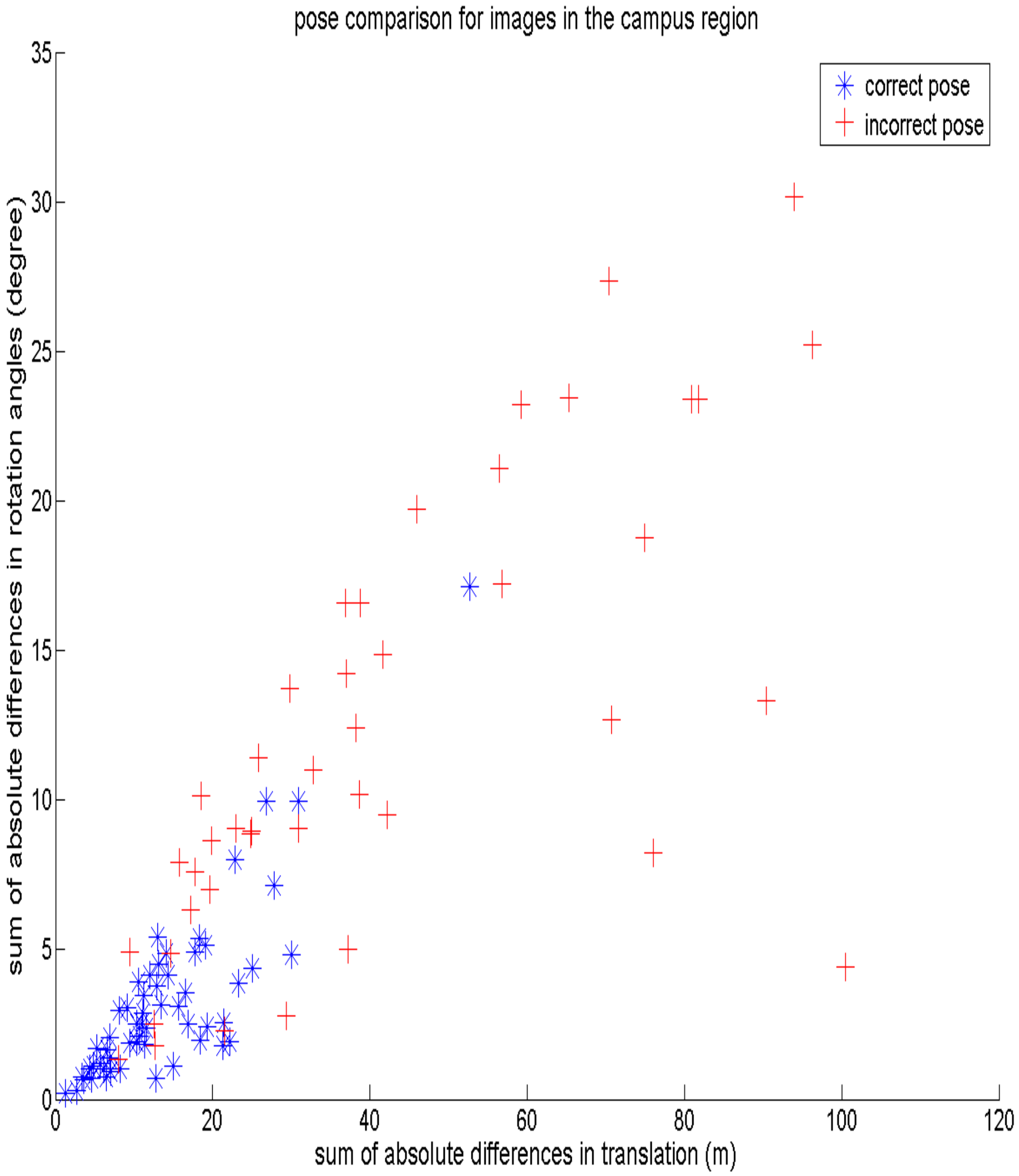
To validate the qualitative visual examination described in Section 6.1, a quantitative examination on accuracy is taken by comparing the recovered poses, ($\phi_{auto}$, $\theta_{auto}$, $\psi_{auto}$, $T^x_{auto}$, $T^y_{auto}$, $T^z_{auto}$), with those computed from manual point correspondence, ($\phi_{manual}$, $\theta_{manual}$, $\psi_{manual}$, $T^x_{manual}$, $T^y_{manual}$, $T^z_{manual}$). Since an extrinsic camera pose consists of six parameters and the accuracy of image projection depends on the overall effect of the six parameters, the component-wise difference in each parameter is not often indicative of the projection accuracy. To accommodate this issue and to systematically examine the overall projection deviation, we propose to use sum of absolute differences in translation parameters, $|T^x_{auto} - T^x_{manual}| + |T^y_{auto} - T^y_{manual}| + |T^z_{auto} - T^z_{manual}|$, and sum of absolute differences in rotation angles, $|\phi_{auto} - \phi_{manual}| + |\theta_{auto} - \theta_{manual}| + |\psi_{auto} - \psi_{manual}|$, as the metrics for comparison.

Even though this set of metrics does consider the overall parameters deviation, it is worthwhile to emphasize that it is still not the most desirable one to measure the image projection error, which is the ultimate quantity the proposed system needs to minimize. For instance, even if only one parameter has a small deviation, and the remaining ones are the same as the ones from manual correspondence, the projection error can be extremely large. On the contrary, deviations in multiple parameters can result in small overall projection error if the effect of the deviation in the parameters cancel each other out. This has to do with the ill-posed nature of the camera pose recovery problem. Nevertheless, the analysis results based on this set of metrics does offer certain degree of validation of our subjective visual rating procedure, as shown in Fig. 6.2.

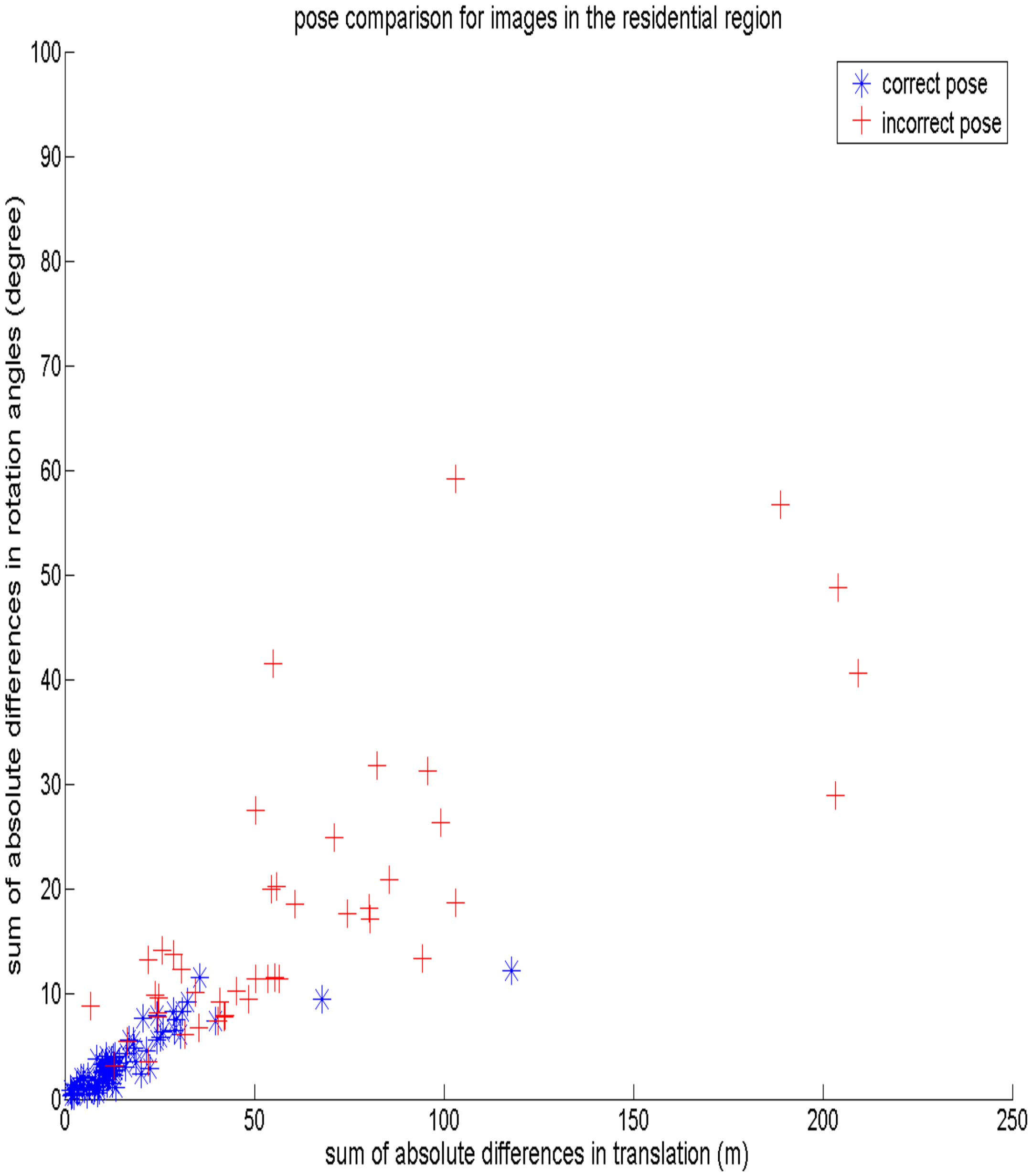
The four scatter plots in Fig. 6.2 correspond to downtown, campus, residential regions and the collection of all three regions. The vertical axis in each of the plots in Fig. 6.2 corresponds to $|\phi_{auto} - \phi_{manual}| + |\theta_{auto} - \theta_{manual}| + |\psi_{auto} - \psi_{manual}|$, and the horizontal axes correspond to $|T^x_{auto} - T^x_{manual}| + |T^y_{auto} - T^y_{manual}| + |T^z_{auto} - T^z_{manual}|$. Blue dots correspond to the difference between manual correspondence and our proposed approach for camera poses of images that are deemed to be correctly recovered via visual inspection, and red dots correspond to those that are deemed to be incorrectly recovered via visual inspection. As expected, visually inspected correct poses tend to have smaller deviations than those deemed to be incorrect via visual inspection. When these deviations are large, the automatically estimated poses, typically but not always, result in large projection errors and are visually rated as incorrect. A fairly clear boundary can be


pose comparison for images in the downtown region
correct pose
incorrect pose
sum of absolute differences in rotation angles (degree)
sum of absolute differences in translation (m)
0
5
10
15
20
25
30
0
20
40
60
80
100
120
140
160
180

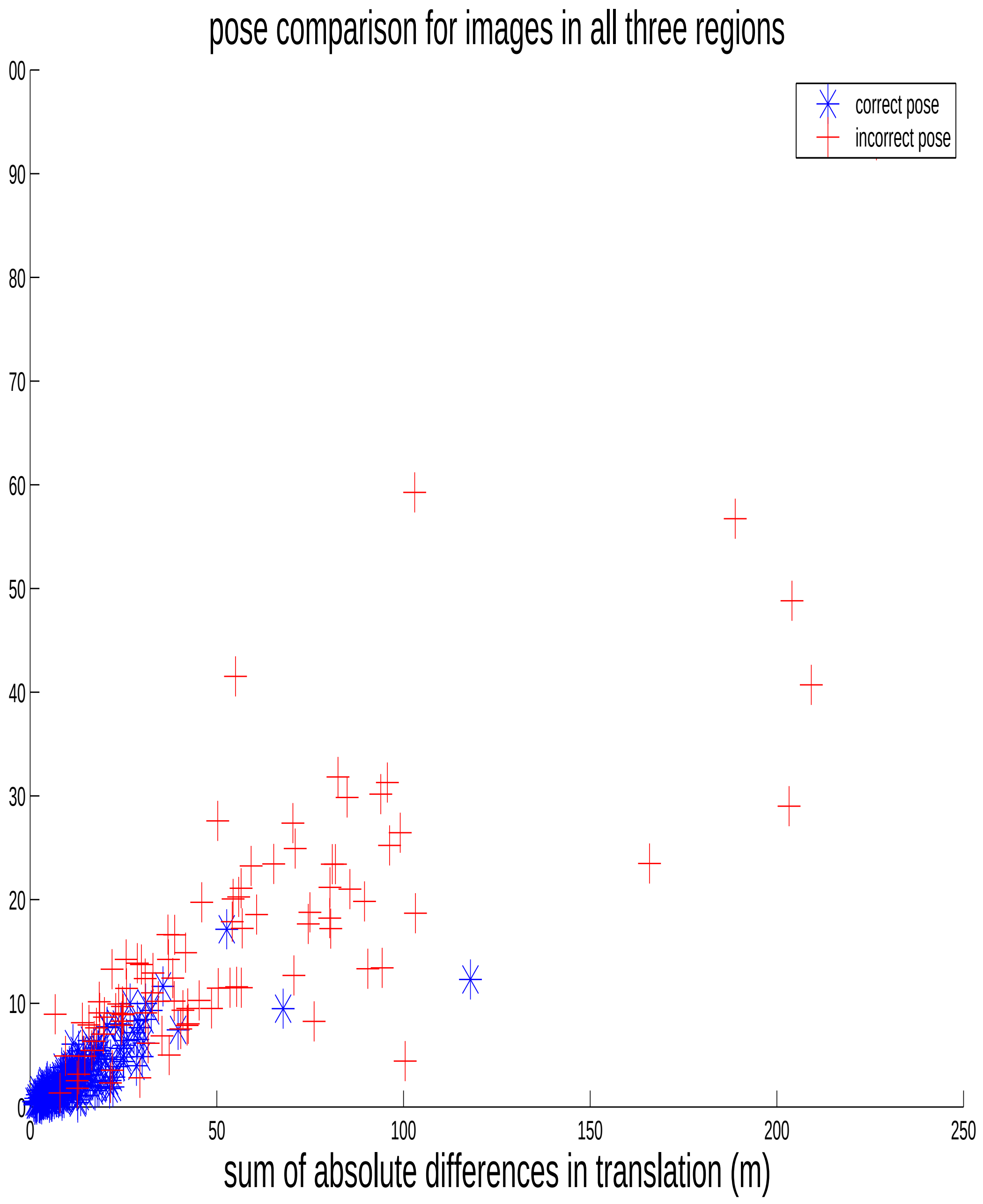
(a)


pose comparison for images in the campus region
correct pose
incorrect pose
sum of absolute differences in rotation angles (degree)
sum of absolute differences in translation (m)
0
5
10
15
20
25
30
35
0
20
40
60
80
100
120

(b)


pose comparison for images in the residential region
100
90
80
70
60
50
40
30
20
10
0
sum of absolute differences in rotation angles (degree)
0
50
100
150
200
250
sum of absolute differences in translation (m)
correct pose
incorrect pose

(c)



(d)

*Figure 6.2: Comparison between automatically estimated poses and the ones from manual correspondence based on sum of the absolute differences in translations parameters and sum of the absolute differences in rotation angles. Pose comparison for the images in (a) downtown region, (b) campus region, (c) residential region, (d) all three regions.*

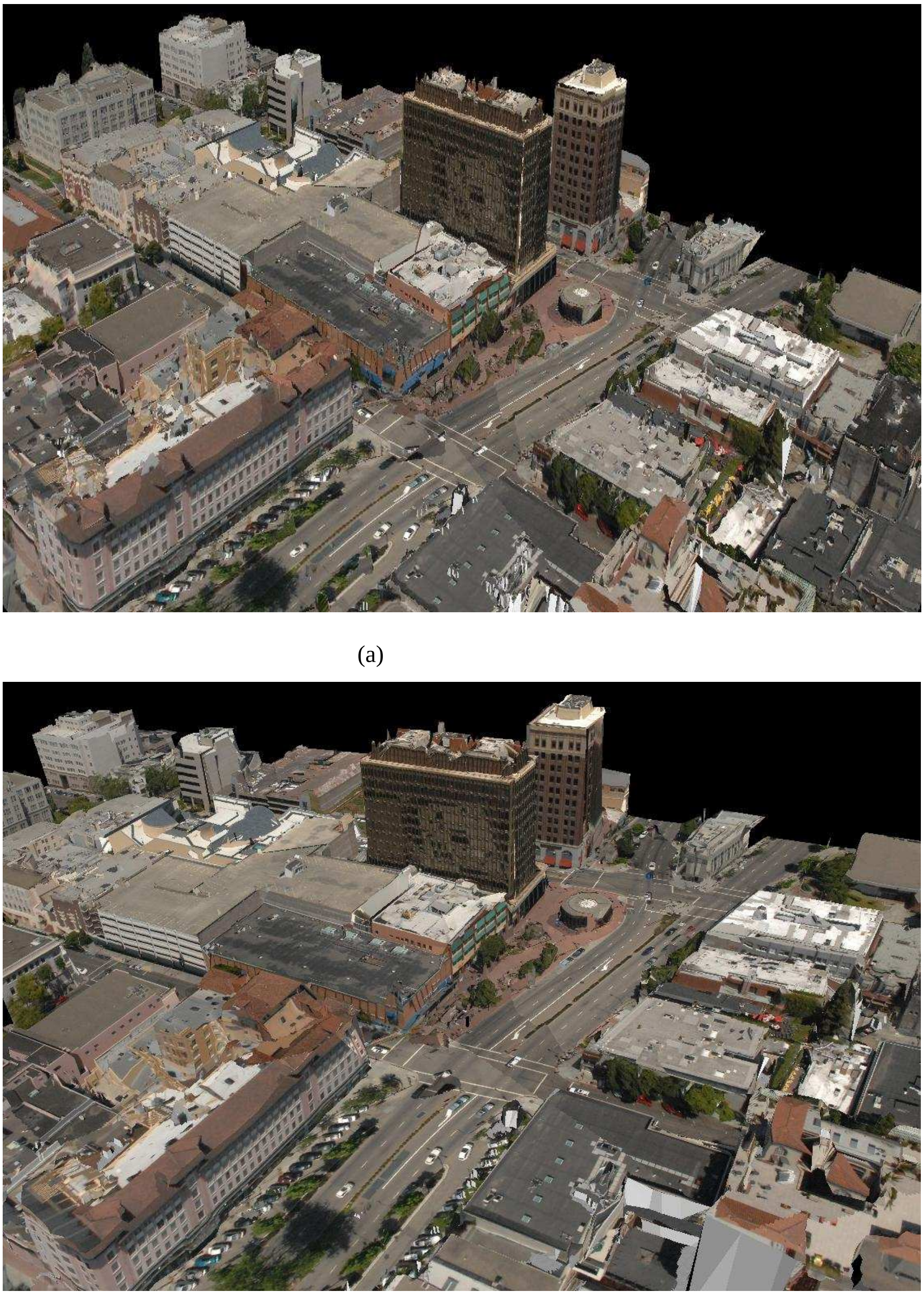
observed between the poses rated as correct and incorrect via visual inspection in all three plots in Fig. 6.2, although the distinction is more evident in the downtown. The reason behind this shall be explained in Section 6.4. This validates our visual rating procedure.

## 6.3 Verification by visual inspection of textured 3D model

Based on the camera poses visually rated as correct, several city models including downtown, campus and residential regions are texture mapped. By examining the alignment between the texture and the models, we can gain an understanding of the accuracy of the automatically generated camera poses and directly validate the visual pose rating process. Several screen shots on the textured city models are displayed in Figures 6.3 through 6.11 along with their counterparts texture mapped with the poses from manual points correspondence. Readers are also encouraged to find these models on `http://www-video.eecs.berkeley.edu/~avz/aironly.htm` for visualization and validation. As shown from the texture alignment from Fig.6.3 through 6.11, the poses rated as correct are indeed sufficiently accurate to enable visually pleasant texture mapping. By comparing the textured models from automatically generated poses and the ones from manual correspondence, nearly no difference in alignment can be identified. In some situations, we actually observe slightly better alignment from the textured model with automatically generated poses.

## 6.4 Discussion

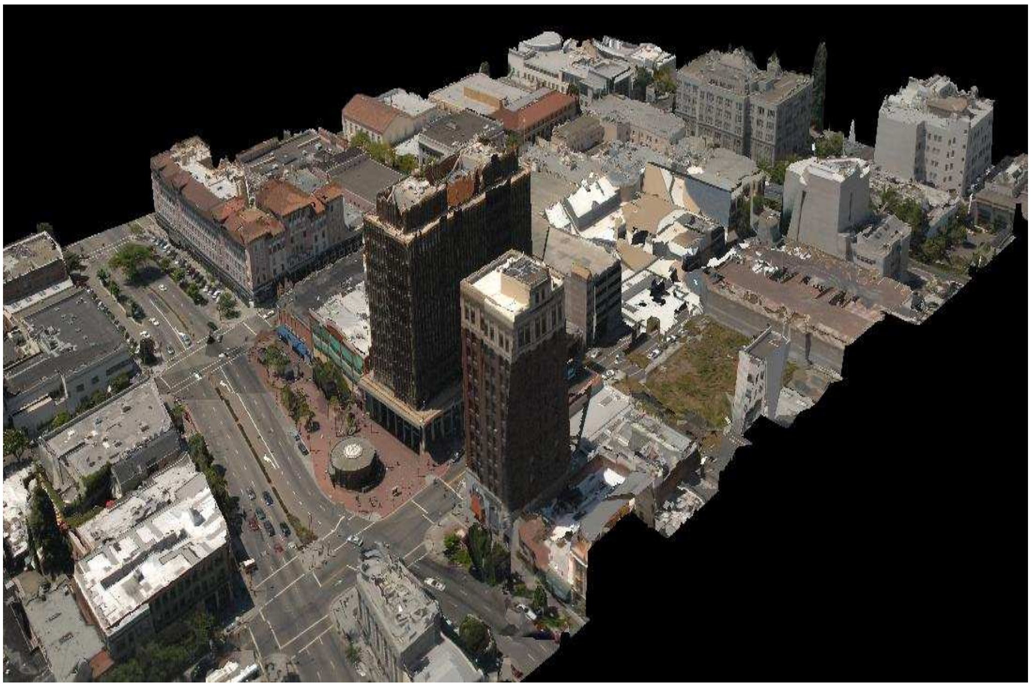
Before carrying out detailed analysis for different regions to reveal the characteristics of the proposed system, a few general remarks can be made. First of all, the average processing time of 191 seconds per image with Intel Xeon 2.8GHz processor is much shorter compared to 25 hours of exhaustive search implemented previously [53]. Furthermore, we have empirically found that over 100 seconds is actually spent on line segment extraction from an image. This shows that the proposed corner correspondence algorithm is extremely efficient.



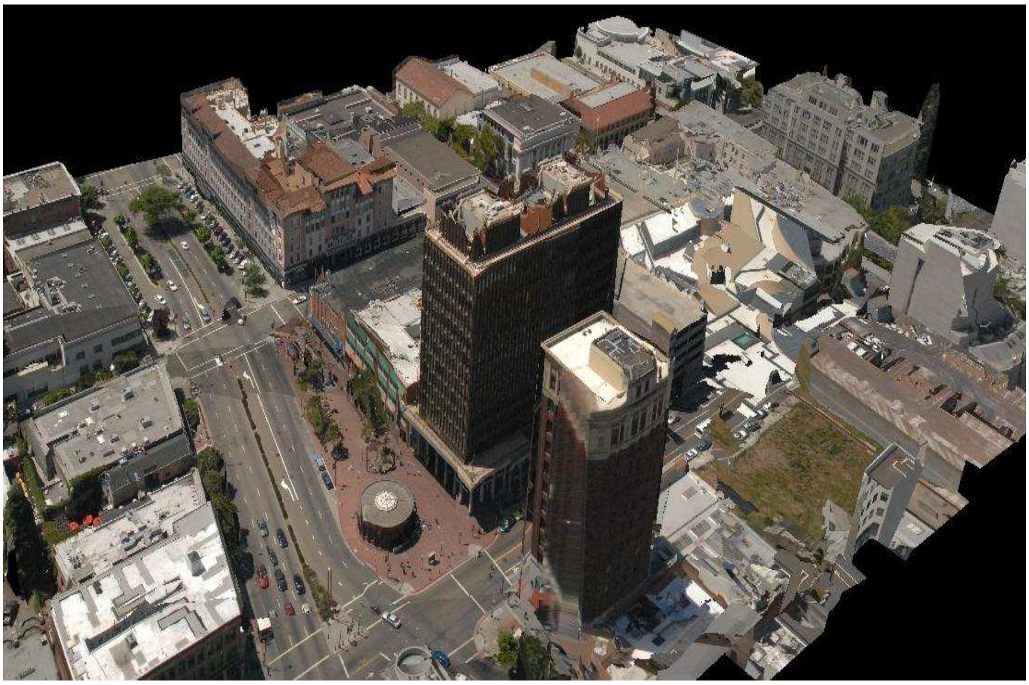
(a)

(b)

*Figure 6.3: Textured models comparison between automatically estimated poses and the ones from manual correspondence: downtown screen shot 1. Screen shot 1 of a texture mapped downtown model with (a) automatically estimated pose, (b) pose derived from manual correspondence.*

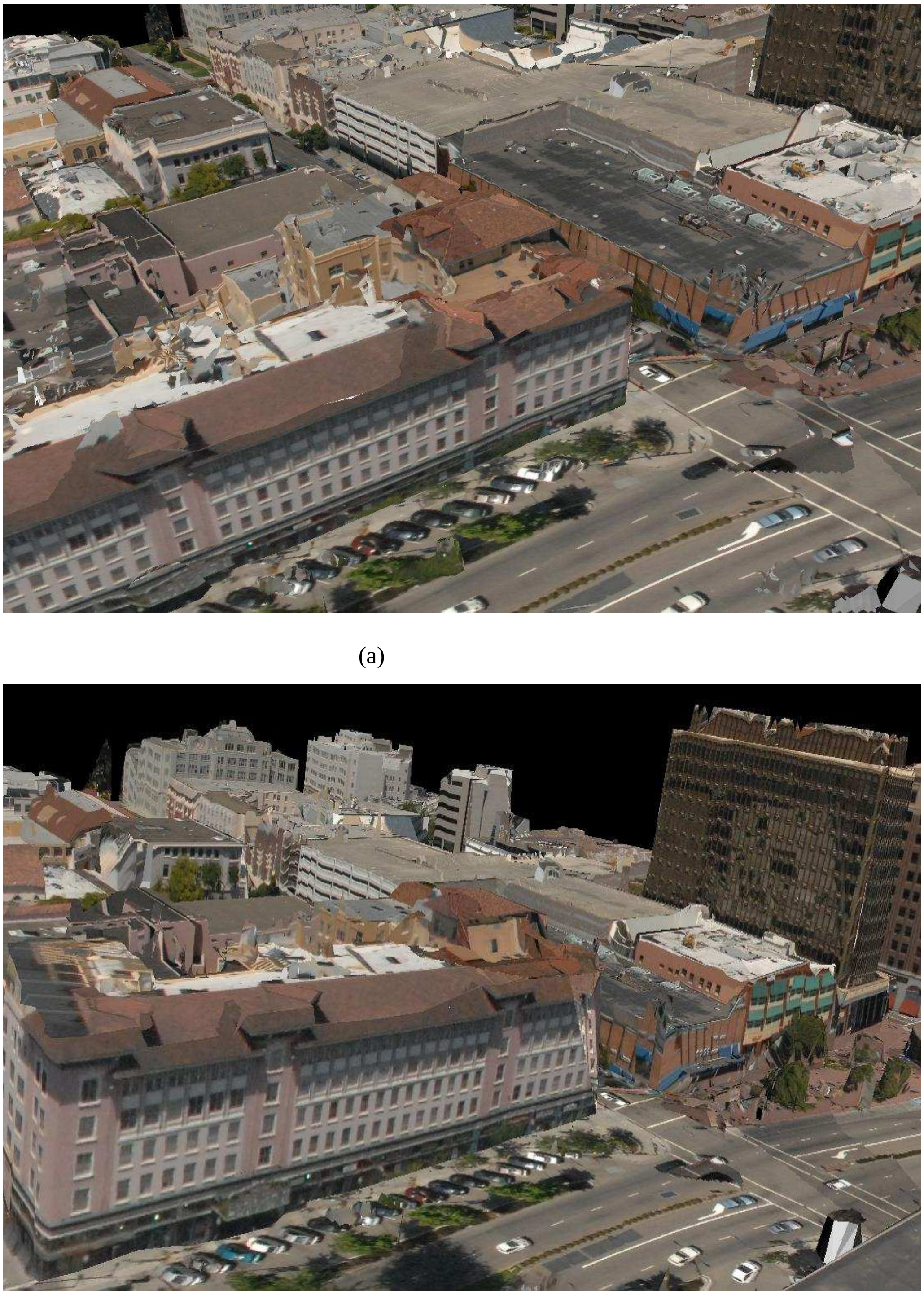


(a)



(b)

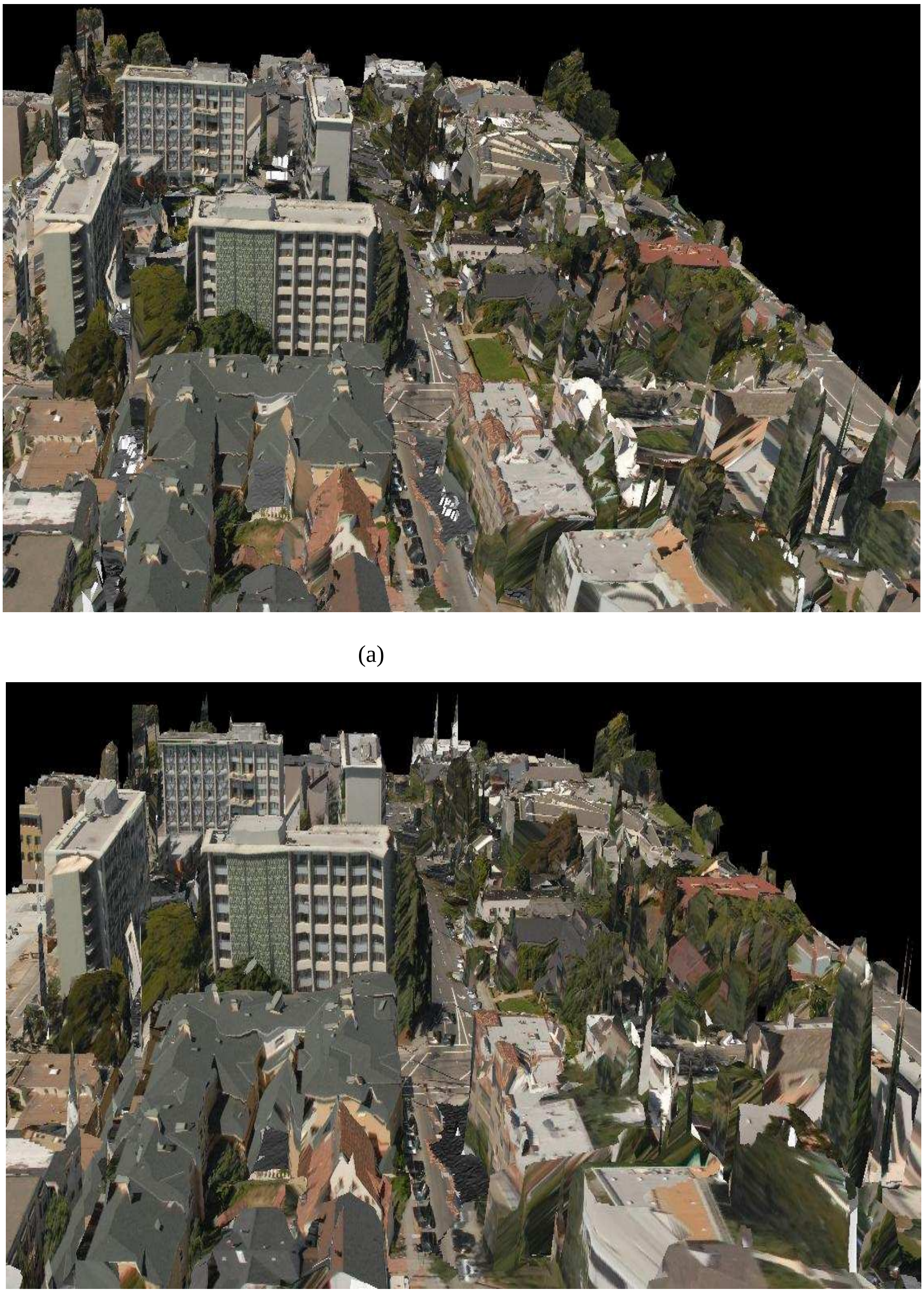
*Figure 6.4: Textured models comparison between automatically estimated poses and the ones from manual correspondence: downtown screen shot 2. Screen shot 2 of a texture mapped downtown model with (a) automatically estimated pose, (b) pose derived from manual correspondence.*



(a)

(b)

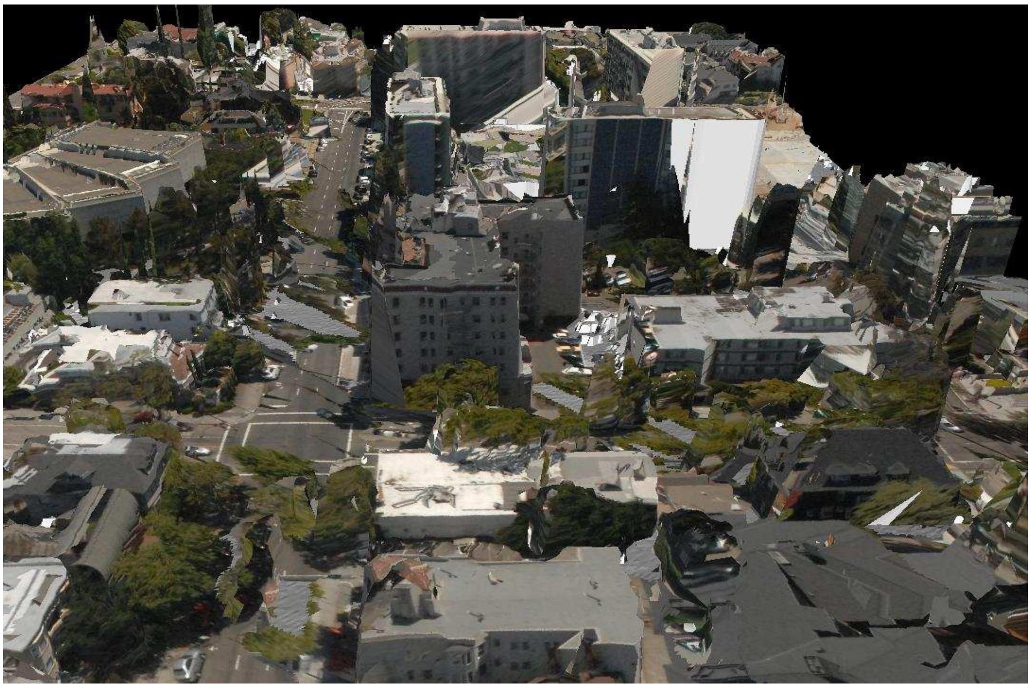
*Figure 6.5: Textured models comparison between automatically estimated poses and the ones from manual correspondence: downtown screen shot 3. Screen shot 3 of a texture mapped downtown model with (a) automatically estimated pose, (b) pose derived from manual correspondence.*



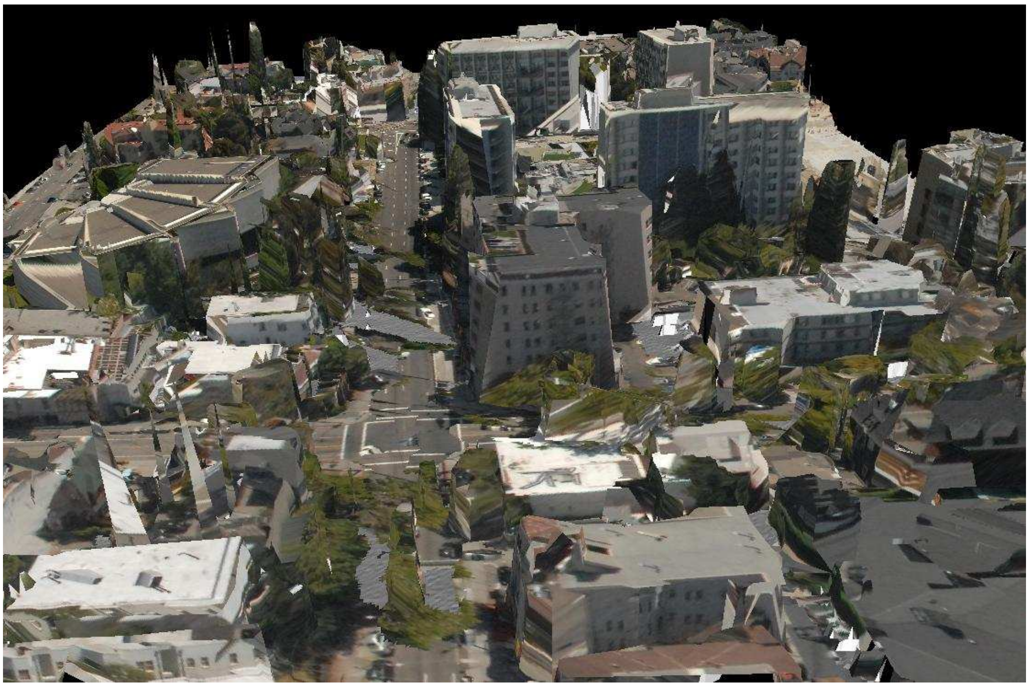
(a)

(b)

*Figure 6.6: Textured models comparison between automatically estimated poses and the ones from manual correspondence: residential area screen shot 1. Screen shot 1 of a texture mapped residential model with (a) automatically estimated pose, (b) pose derived from manual correspondence.*

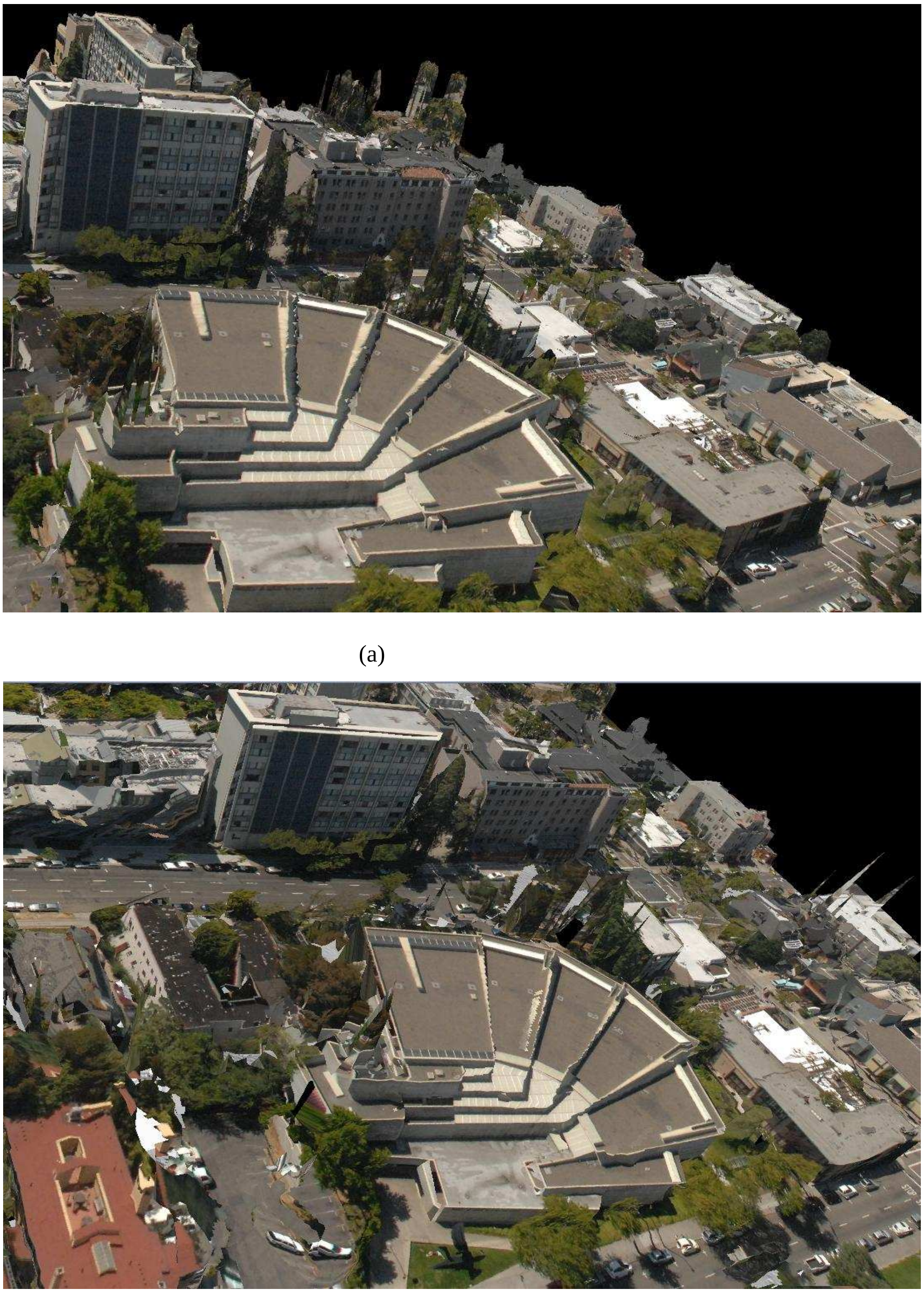


(a)



(b)

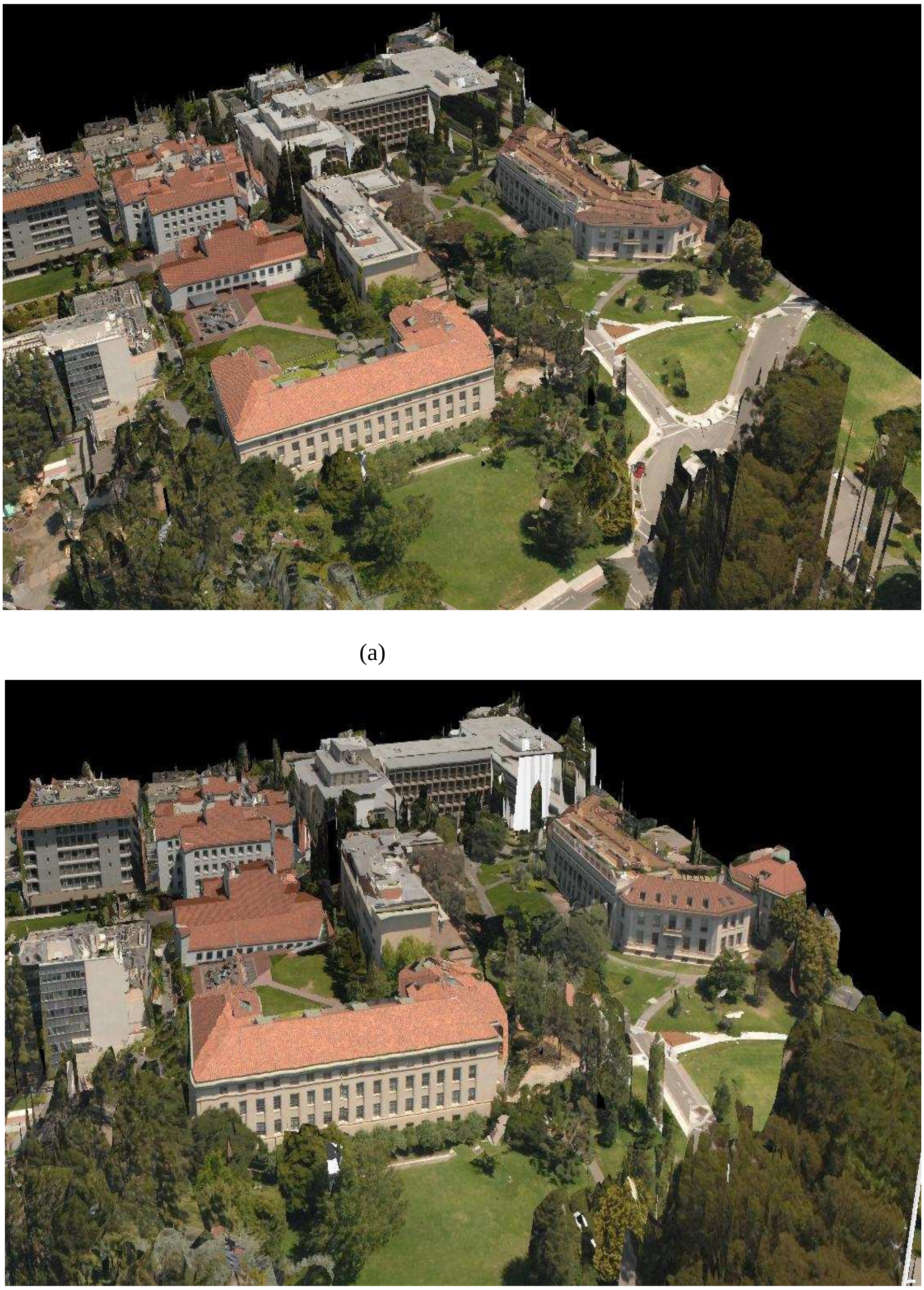
*Figure 6.7: Textured models comparison between automatically estimated poses and the ones from manual correspondence: residential area screen shot 2. Screen shot 2 of a texture mapped residential model with (a) automatically estimated pose, (b) pose derived from manual correspondence.*



(a)

(b)

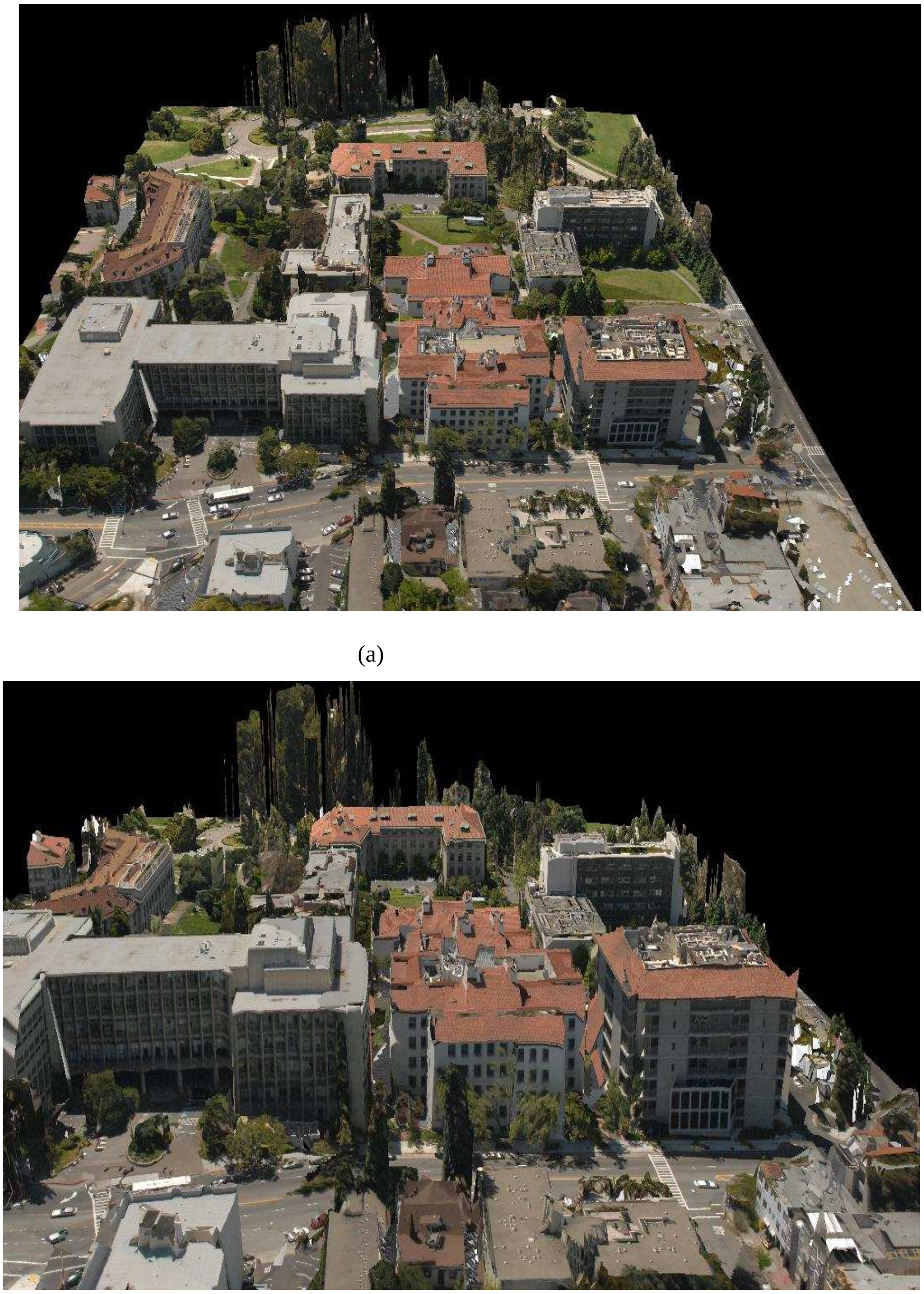
*Figure 6.8: Textured models comparison between automatically estimated poses and the ones from manual correspondence: residential area screen shot 3. Screen shot 3 of a texture mapped residential model with (a) automatically estimated pose, (b) pose derived from manual correspondence.*



(a)

(b)

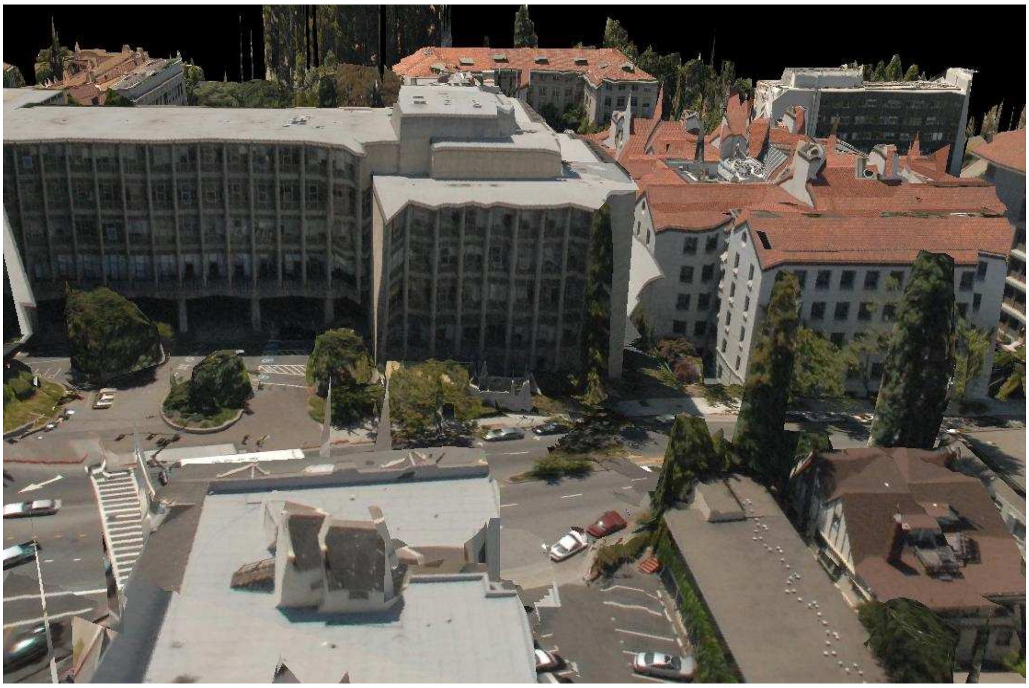
*Figure 6.9: Textured models comparison between automatically estimated poses and the ones from manual correspondence: campus screen shot 1. Screen shot 1 of a texture mapped campus model with (a) automatically estimated pose, (b) pose derived from manual correspondence.*



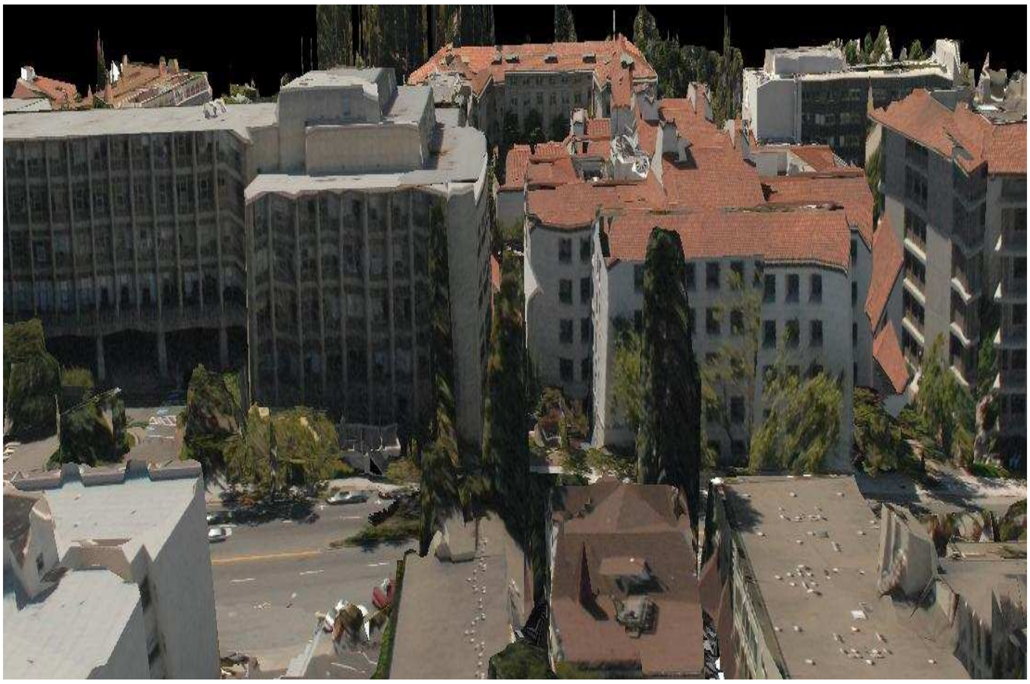
(a)

(b)

*Figure 6.10: Textured models comparison between automatically estimated poses and the ones from manual correspondence: campus screen shot 2. Screen shot 2 of a texture mapped campus model with (a) automatically estimated pose, (b) pose derived from manual correspondence.*



(a)



(b)

*Figure 6.11: Textured models comparison between automatically estimated poses and the ones from manual correspondence: campus screen shot 3. Screen shot 3 of a texture mapped campus model with (a) automatically estimated pose, (b) pose derived from manual correspondence.*

| Region | Correct | Incorrect reason | |
|---|---|---|---|
| | | Few corners | Random pose |
| Downtown | 82 | 0 | 8 |
| Campus | 57 | 37 | 18 |
| Residential | 78 | 51 | 27 |
| Total | 217 | 88 | 53 |

*Table 6.1: Correct pose recovery rate and source of error from our proposed system for different regions using visual inspection described in Section 6.1.*

The result of applying subjective visual inspection as described in Section 6.1 to the pose recovery algorithm in Chapters 2 and 3 on the collected data is summarized in Table 6.1. Refer to Appendix C for detailed result analysis for each image. The overall performance with 61% correct recovery rate is promising, especially with 91% correct recovery rate from the downtown district. By examining the regional performance difference, a major trade-off on the system performance is revealed. From Table 6.1, it is evident that the main reason for camera pose recovery failure is that not enough correct 2DOC matches are available. Since it is necessary to have a minimum four corner correspondences to fit a Homography matrix, our proposed system simply cannot find the correct pose in this situation. Although it is possible to extract more 2DOCs by relaxing certain processing parameters such as minimum line length threshold, this could potentially lead to additional erroneous 2DOCs, resulting in a decrease in the percentage of correct 2DOC matches. As stated earlier, various stages of our proposed system assume that the pose with the maximum consensus is the ground truth camera pose. When this percentage of correct 2DOC matches is significantly small, this assumption tends to break down. This situation is observed in the downtown district. The downtown is the region where the system performs the best with over ninety percent correct pose recovery rate. The main reason behind this success is that the majority of the buildings have both large and simple rigid shapes. In this area, 2DOCs from both the DSM and the images are easily extracted since the line extraction from the image and contour simplification from the DSM are both straight forward and orthogonal structural corners are abundant. As such there are plenty of correct 2DOC matches. In fact, the system failure in the 8 images of the downtown is rather due to too many erroneous 2DOCs. The percentage of inliers, $p_{inliers}$, is low enough to mislead the system to some random camera poses.

The natural question is why correct 2DOCs are difficult to extract after relaxing processing parameters beyond certain limit. The analysis on campus and residential area reveals two fundamental difficulties in extracting 2DOCs from a DSM.

1. Building density
   One of the fundamental limiting factors is the building density in a region. This effect is clearly observed by comparing the results from the downtown and campus area. Both areas are characterized by large buildings; however, the building density is dramatically lower in campus than in downtown. For instance, some of the aerial images of the campus only contain a portion of a building since the buildings (a) tend to be larger than the ones in the downtown and (b) are spread out. Even if one or two buildings are present in an image, the system might not be able to obtain enough 2DOCs due to imperfect contour simplification and complex building structures on the campus. Furthermore, this small number of 2DOCs matches often does not provide enough constraint in Lowe's pose recovery algorithm. This is the reason for generally higher camera parameter error in the campus compared to that in the downtown, as observed in Fig. 6.2. By the same reasoning, the downtown district with large number of 2DOCs has the lowest parameter error. This demonstrates that the proposed system performs poorly in open fields where buildings are sparsely distributed since it requires a large number of 2DOCs for robustness and accuracy.

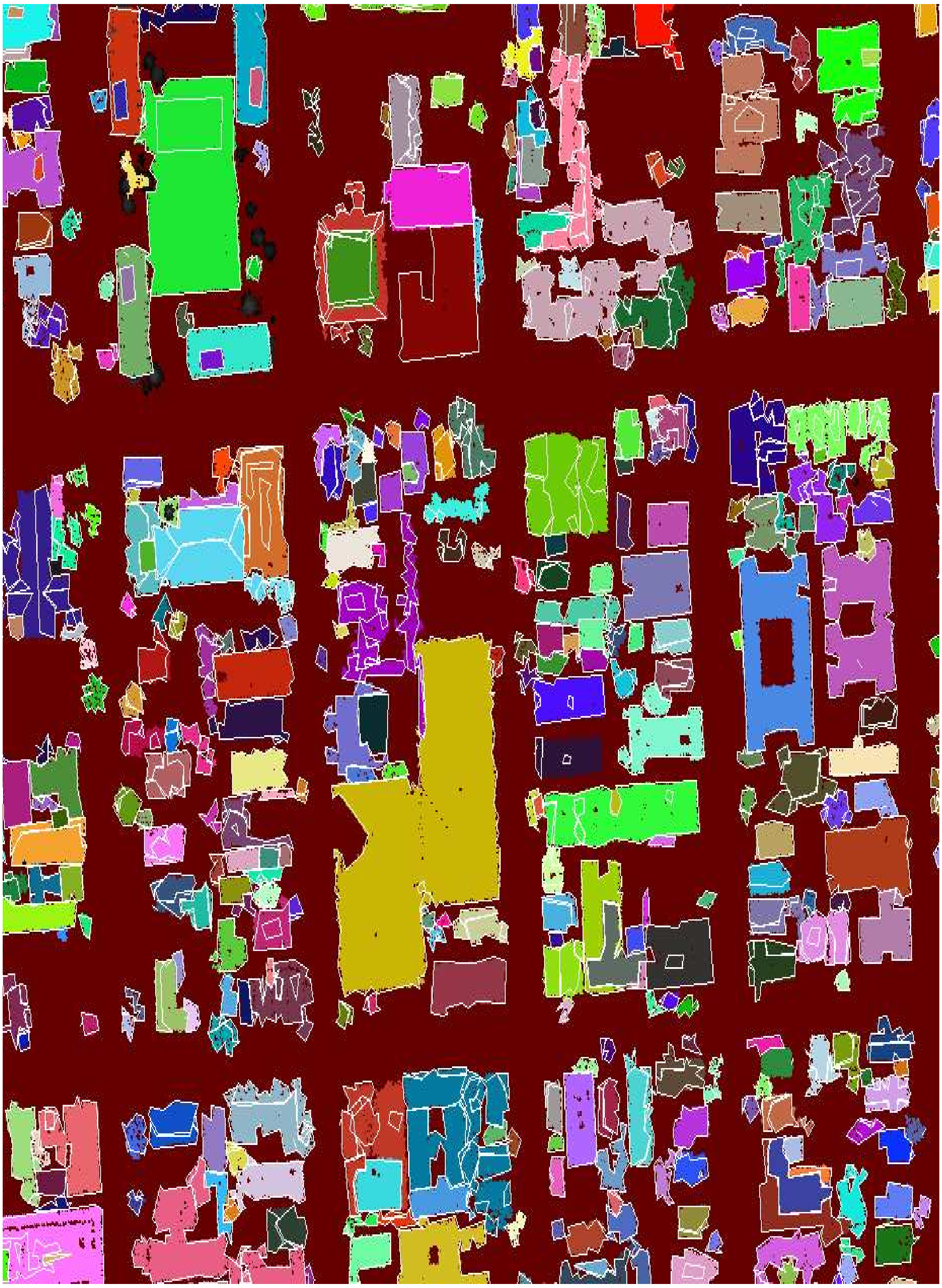
2. Tree occlusion
   The residential area has a similar building density as in the downtown. The performance however is much lower due to lack of correct 2DOCs from the DSM. This is because the trees near the houses are included as part of the buildings after region segmentation and the resulting region contours are more complex than the original ones from the buildings themselves. It is thus very difficult to extract correct 2DOCs from these irregular shaped contours. At the same time, a large number of erroneous 2DOCs are added because of this. In other words, the proposed system suffers from both limited number of correct 2DOCs and large number of erroneous 2DOCs when trees are present near small buildings. This effect is less severe in the campus where the buildings are large enough that relatively small distortions on the contour due to tree occlusion can be removed by DP algorithm.

## 6.5 Residential area analysis after tree removal

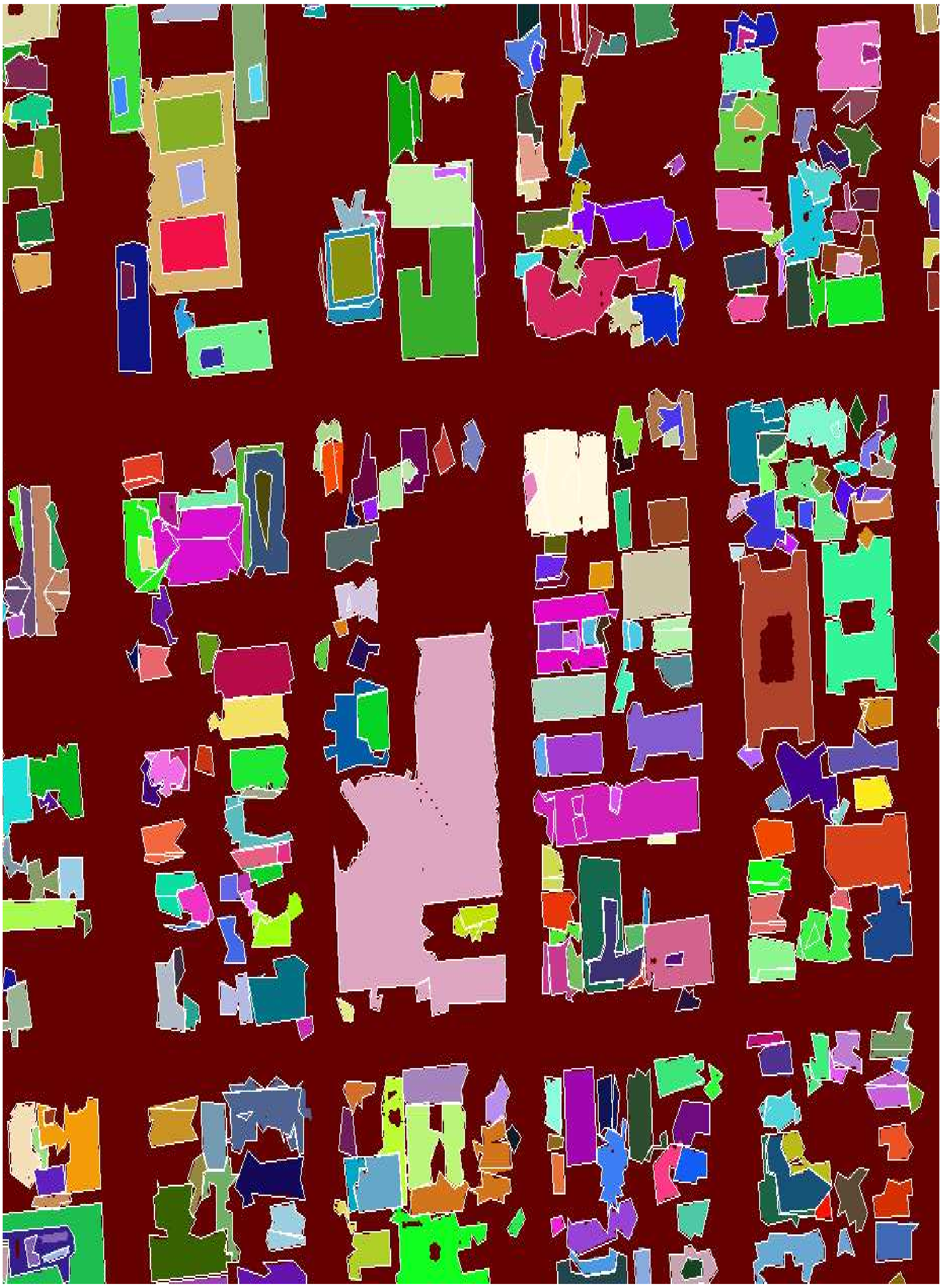
Tree occlusion might be improved by performing tree removal on a DSM priori to 2DOC detection. To assess the performance gain, the tree removal by weighted support vector machine implemented in [21] is performed on a sample area bounded by the blue lines as shown in Fig.6.1. Here we make the same assumption as in [21], where a top-down image and a DSM are assumed to be perfectly registered. Even though it is the eventual goal of our system to achieve this registration, we are merely interested in finding out whether or not a tree detection and/or removal algorithm can improve our system's performance.

Tree removal process in [21] consists of segmentation and classification. DSM is first segmented by region growing algorithm based on weighted features from aerial image and LiDAR for region similarity determination. These features are height, texture map, height variation, and normal vector estimates. The weights for the features are determined using a learning method on random walks. Weighted support vector machine is then used to perform a binary detection problem for tree classification.

DSMs before and after the tree removal are displayed in Fig. 6.12 along with the simplified contour lines. The performance comparison on 54 aerial images covering this area is presented on Table 6.2. Surprisingly, the performance improvement is less than 10%. After careful examination on the segmented regions, it is observed that the building contours after the tree removal do not necessarily correspond to the actual house outlines as desired. This is because of trees overhanging houses and that removing trees near the house in a DSM alone creates concavities on region boundaries. In some building contours, we find that correct 2DOCs detected before the tree removal are lost due to these extra concavities after the tree removal. Recall that the 2DOC determination from a DSM is based on the intersecting angle and the lengths of the intersecting contour line segments. This is the reason that more pose recovery failures due to few corners are observed after the tree removal. Another reason for the poor performance gain is the limited ability to extract 2DOCs from images. There is limited number of visible 2DOCs on images due to inevitable tree occlusion.



(a)



(b)

*Figure 6.12: Region segmentation and contour simplification comparison between before and after the tree removal from a DSM: (a) before tree removal, (b) after tree removal.*

| | Correct | Incorrect reason | |
|---|---|---|---|
| | | Few corners | Random pose |
| before tree removal | 28 | 5 | 21 |
| after tree removal | 32 | 7 | 15 |

*Table 6.2: Performance comparison among 54 images between before and after the tree removal from a DSM*

# Chapter 7

# Conclusion and Future Directions

Based on 358 images over Berkeley, we have demonstrated a monocular aerial images registration system for a 3D model texture mapping. Across three regions including the downtown, campus and residential area, the proposed system has achieved overall 61% correct camera pose recovery rate. In the downtown district, it is 91% accurate in particular.

At the same time, this system is considerably more computationally efficient as compared to other existing techniques [29, 53]. In particular, it is over 450 times faster than our previous exhaustive search approach [53]. With fast airborne data acquisition and this much more efficient automated camera pose recovery algorithm, our proposed airborne based 3D model reconstruction methodology can scale to very large regions. The Lidar data acquisition time scales linearly with the area of a region. Similarly, the aerial image acquisition time scales linearly with the area, since each image contains approximately the same coverage area when the altitude of the helicopter, the pitch angle and focal length of the camera are fixed during the image acquisition. Finally, because our proposed camera pose recovery system analyzes one image at a time independently, its complexity also grows linearly with the number of images and therefore with the area. Given both the DSM and images with recovered camera poses, it is possible to reconstruct a 3D model in linear complexity with respect to the area [22, 23]. Therefore, we have developed an automated airborne based 3D model reconstruction system with its complexity scaling linearly with the area.

Several future directions can be pursued to improve the current system based on the result analysis, where two major performance limiting factors have been identified. First one is the poor performance in low building density regions. Since this issue is difficult to resolve with the current approach, an alternative approach must be developed to specifically deal with this situation. In particular, line matching between a DSM and images migth be simple and effective since most of the long line segments are from buildings and occlusion among buildings is rare in this area. The second issue is tree occlusion. Simple tree removal alone from a DSM has resulted in limited performance gain. It is necessary not to just remove trees but also to fill in the resulting concavities to preserve the original rectangular building contours. At the same time, trees should also be removed from images to produce correct visualization. Thus, a parameterized roof region fitting might be beneficial to both tree removal and 2DOCs extraction. Finally, it is also important to develop certain automatically generated metrics to indicate whether the recovered pose is indeed correct. This not only reduces the number of incorrect poses, but also eliminates the necessity of human operator to visually validate the recovered pose of each image. We have already investigated the suitability of the following metrics, and found that none of them can be used to replace human visual inspection: (a) the magnitude of the maximum in the Hough space for rotation identification; (b) the Homography fitting error; (c) the number of correct matches from GMSAC; (d) Lowe's algorithm's fitting error. As such, it would be desirable to find a metric or an indicator to replace human visual inspection for validation purpose. For instance, the fitting error between line segments from a DSM and an image proposed in [53] might be a good candidate for this task.

# Bibliography

[1] A. Fischer, T. H. Kolbe, F. Lang, A. B. Cremers, W. Förstner, L. Plümer, and V. Steinhage, "Extracting buildings from aerial images using hierarchical aggregation in 2D and 3D," *Computer Vision and Image Understanding: CVIU*, vol. 72, no. 2, pp. 185–203, 1998.

[2] J. C. McGlone and J. A. Shufelt, "Projective and object space geometry for monocular building extraction," in *in Proceedings: IEEE Conference on Computer Vision and Pattern Recognition*, pp. 54–61, June 1994.

[3] J. A. Shufelt, "Performance evaluation and analysis of monocular building extraction from aerial imagery," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 21, no. 4, pp. 311–326, 1999.

[4] C. Lin and R. Nevatia, "Building detection and description from a single intensity image," *Computer Vision and Image Understanding: CVIU*, vol. 72, no. 2, pp. 101–121, 1998.

[5] E. Guillou, D. Meneveaux, E. Maisel, and K. Bouatouch, "Using vanishing points for camera calibration and coarse 3d reconstruction from a single image," *The Visual Computer*, vol. 16, pp. 394–410, Nov. 2000.

[6] D. G. Aguilera, J. G. Lahoz, and J. F. Codes, "A new method for vanishing point detection in 3d reconstruction from a single view," in *Proceedings of the ISPRS WG V/4 Workshop '3D-ARCH 2005'*, 2005.

[7] T. Moons, D. Frere, J. Vandekerckhove, and L. J. V. Gool, "Automatic modeling and 3d reconstruction of urban house roofs from high resolution aerial imagery," in *ECCV '98: Proceedings of the 5th European Conference on Computer Vision-Volume I*, (London, UK), pp. 410–425, Springer-Verlag, 1998.

[8] R. Cipolla, T. Drummond, and D. Robertson, "Camera calibration from vanishing points in images of architectural scenes," 1999.

[9] R. Cipolla and E. Boyer, "3d model acquisition from uncalibrated images," in *In Proc. IAPR Workshop on Machine Vision Applications*, (Chiba, Japan), pp. 559–568, Nov. 1999.

[10] T. Werner and A. Zisserman, "New techniques for automated architectural reconstruction from photographs," in *ECCV '02: Proceedings of the 7th European Conference on Computer Vision-Part II*, (London, UK), pp. 541–555, Springer-Verlag, 2002.

[11] A. Zisserman, F. Schaffalitzky, T. Werner, and A. Fitzgibbon, "Automated reconstruction from multiple photographs," in *In. Proc.International Conference on Image Processing*, vol. 3, pp. 517–520, June 2002.

[12] S. Noronha and R. Nevatia, "Detection and modeling of buildings from multiple aerial images," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 23, pp. 501–518, May 2001.

[13] M. Pollefeys, L. V. Gool, M. Vergauwen, F. Verbiest, K. Cornelis, J. Tops, and R. Koch, "Visual modeling with a hand-held camera," *Int. J. Comput. Vision*, vol. 59, no. 3, pp. 207–232, 2004.

[14] A. Akbarzadeh, J. M. Frahm, P. Mordohai, B. Clipp, C. Engels, D. Gallup, P. Merrell, M. Phelps, S. Sinha, B. Talton, L. Wang, Q. Yang, H. Stewenius, R. Yang, G. Welch, H. Towles, D. Nister, and M. Pollefeys, "Towards urban 3d reconstruction from video," 2006.

[15] A. Brunn and U. Weidner, "Hierarchical bayesian nets for building extraction using dense digital surface models," *ISPRS journal of photogrammetry and remote sensing*, vol. 53, no. 5, pp. 296–307, 1998.

[16] N. Haala and C. Brenner, "Extraction of building and trees in urban environments," *ISPRS journal of photogrammetry and remote sensing*, vol. 54, no. 2-3, pp. 130–137, 1999.

[17] H. G. Mass and G. Vosselman, "Two algorithms for extracting building models from raw laser altimetry data," *ISPRS journal of photogrammetry and remote sensing*, vol. 54, no. 2-3, pp. 153–163, 1999.

[18] J. M. Hill, L. A. Graham, R. Henry, D. M. Cotter, and D. Young, “Wide-area topographic mapping and applications using airborne light detection and ranging (lidar) technology,” *Photogrammetric engineering and remote sensing*, vol. 66, no. 8, pp. 908–960, 2000.

[19] P. Gamba and B. Houshmand, “Digital surface models and building extraction: A comparison of ifsar and lidar data,” *IEEE Trans. Geosci. Remote Sensing*, vol. 38, no. 4, pp. 1959–1968, 2000.

[20] S. K. Lodha and D. H. D. M Fitzpatrick, “Aerial lidar data classification using adabost,” in *3-D Digital Imaging and Modeling*, (Montreal, QC, Canada), pp. 435–442, Aug. 2007.

[21] J. Secord and A. Zakhor, “Tree detection in aerial lidar and image data,” in *IEEE International Conference on Image Processing*, pp. 2317–2320, Oct. 2006.

[22] C. Fruh and A. Zakhor, “Constructing 3d city models by merging aerial and ground views,” *IEEE Comput. Graph. Appl.*, vol. 23, pp. 52–61, Nov-Dec 2003.

[23] C. Fruh and A. Zakhor, “An automated method for large-scale, ground-based city model acquisition,” *International Journal of Computer Vision*, vol. 60, pp. 5–24, Oct. 2004.

[24] H. Zhao and R. Shibasaki, “Reconstructing urban 3d model using vehicle-borne laser range scanners,” in *Third International Conference on 3D Digital Imaging and Modeling*, pp. 349–356, May 2001.

[25] L. Liu and I. Stamos, “Automatic 3d to 2d registration for the photorealistic rendering of urban scenes,” in *CVPR ’05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05) - Volume 2*, (Washington, DC, USA), pp. 137–143, IEEE Computer Society, 2005.

[26] L. Liu, I. Stamos, G. Yu, G. Wolberg, and S. Zokai, “Multiview geometry for texture mapping 2d images onto 3d range data,” in *CVPR ’06: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, (Washington, DC, USA), pp. 2293–2300, IEEE Computer Society, 2006.

[27] I. Stamos and P. K. Allen, “Automatic registration of 2-d with 3-d imagery in urban environments,” in *Proceedings of the Eighth IEEE International Conference on Computer Vision*, vol. 2, pp. 731–736, IEEE Computer Society, July 2001.

[28] I. Stamos and P. K. Allen, “Geometry and texture recovery of scenes of large scale,” *Comput. Vis. Image Underst.*, vol. 88, no. 2, pp. 94–118, 2002.

[29] W. Zhao, D. Nister, and S. Hsu, “Alignment of continuous video onto 3d point clouds,” *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 27, no. 8, pp. 1305–1318, 2005.

[30] S. Hsu, S. Samarasekera, R. Kumar, and H. S. Sawhney, “Pose estimation, model refinement and enhanced visualization using video,” in *CVPR ’00: Proceedings of the 2000 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 488–495, July 2000.

[31] U. Neumann, S. You, J. Hu, B. Jiang, and J. Lee, “Augmented virtual environments (ave): Dynamic fusion of imagery and 3d models,” in *VR ’03: Proceedings of the IEEE Virtual Reality 2003*, (Washington, DC, USA), p. 61, IEEE Computer Society, 2003.

[32] S. C. Lee, S. K. Jung, and R. Nevatia, “Integrating ground and aerial views from urban site modeling,” in *In Proceedings of the 16th International Conference on Pattern Recognition*, vol. 4, pp. 107–112, 2002.

[33] S. C. Lee, S. K. Jung, and R. Nevatia, “Automatic pose estimation of complex 3d building models,” in *WACV ’02: Proceedings of the Sixth IEEE Workshop on Applications of Computer Vision*, (Washington, DC, USA), p. 148, IEEE Computer Society, 2002.

[34] S. C. Lee, S. K. Jung, and R. Nevatia, “Automatic integration of facade textures into 3d building models with a projective geometry based line clustering.”

[35] J. Hu, S. You, and U. Neumann, “Integrating lidar, aerial image and ground images for complete urban building modeling,” in *Third International Symposium on 3D Data Processing, Visualization, and Transmission*, pp. 184–191, June 2006.

[36] J. Hu, S. You, and U. Neumann, “Automatic pose recovery for high-quality textures generation,” in *18th International Conference on Pattern Recognition*, pp. 561–565, Aug. 2006.

[37] D. G. Lowe, “Three-dimensional object recognition from single two-dimensional images,” *Artificial Intelligence*, vol. 31, no. 3, pp. 355–395, 1987.

[38] S. Barnard, “Interpreting perspective images,” *Artificial Intelligence*, vol. 21, pp. 435–462, 1983.

[39] J. A. Shufelt, “Performance evaluation and analysis of vanishing point detection,” *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 21, pp. 282–288, Mar. 1999.

[40] J. Kosecka and W. Zhang, “Video compass,” in *Proceedings of European Conference on Computer Vision*, pp. 657 – 673, 2002.

[41] R. Vidal, M. Yi, and J. Piazzi, “A new gpca algorithm for clustering subspaces by fitting, differentiating and dividing polynomials,” in *Proceedings of CVPR 2004*, pp. 510–517, June 2004.

[42] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second ed., 2004.

[43] C. Harris and M. J. Stephens, “A combined corner and edge detector,” in *In Alvey Vision Conference*, pp. 147–152, 1988.

[44] D. H. Douglas and T. K. Peucker, “Algorithms for the reduction of the number of points required to represent a digitized line or its caricature,” *Cartographica: The International Journal for Geographic Information and Geovisualization*, vol. 10, pp. 112–122, oct 1973.

[45] J. S. Marino, “Identification of characteristic points along naturally occurring lines: An empirical study,” *Canadian Cartographer*, vol. 16, pp. 70–80, 1979.

[46] D. Lowe, “Distinctive image features from scale-invariant keypoints,” in *International Journal of Computer Vision*, vol. 20, pp. 91–110, 2003.

[47] W. Zhang and J. Kosecka, “Generalized ransac framework for relaxed correspondence problems,” in *International Symposium on 3D Data Processing, Visualization and Transmission*, 2006.

[48] B. K. P. Horn, “Closed-form solution of absolute orientation using unit quaternion,” *Journal of the Optical Society of America A*, vol. 4, no. 4, pp. 629–643, 1987.

[49] W. R. Hamilton, *Elements of Quaternions*. Chelsea Publishing Company, 3 ed., March 1969.

[50] M. A. Fischler and R. C. Bolles, “Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography,” *Communications of the ACM*, vol. 24, pp. 381–395, June 1981.

[51] P. H. S. Torr and A. Zisserman, “Mlesac: a new robust estimator with application to estimating image geometry,” *Comput. Vis. Image Underst.*, vol. 78, no. 1, pp. 138–156, 2000.

[52] Y. Ma, S. Soatto, J. Kasecka, and S. S. Sastry, *An Invitation to 3-D Vision From Images to Geometric Models*. New York: Springer, 2004.

[53] C. Frueh, R. Sammon, and A. Zakhor, “Automated texture mapping of 3d city models with oblique aerial imagery,” in *In Proceedings of the 2nd International Symposium on 3D data processing, visualization and transmission*, pp. 396–403, Sept. 2004.

[54] M. Garland and P. S. Heckbert, “Simplifying surfaces with color and texture using quadric error metrics,” in *IEEE Visualization ’98* (D. Ebert, H. Hagen, and H. Rushmeier, eds.), pp. 263–270, 1998.

[55] A. H. Watt, *3D Computer Graphics*. Pearson Education Limited, 3 ed., 2000.

# Appendix A

# Camera Calibration Based on Three Orthogonal Vanishing Points

Intrinsic camera parameters are assumed to be known as a priori in Chapter 2. Theoretically, it is possible to estimate the intrinsic camera parameters from three orthogonal vanishing points as in [32]. The details of this estimation are described in this chapter. Since we have empirically observed that more accurate camera angles can be obtained from the vertical vanishing point with fixed intrinsic parameters, the following method is not actually implemented.

## A.1 Three orthogonal vanishing points selection

After both vertical and non-vertical vanishing points detection are performed, more than three vanishing points are typically obtained due to the complexity of urban settings and noises such as shadows and trees. To select three orthogonal vectors from the camera origin to the vanishing points, which we simply refer as three orthogonal vanishing points from now on, the principal point is first assumed to be at the image center. It can be easily shown that the principal point is the orthocenter of the triangle formed by three orthogonal vanishing points on the image plane [32]. Thus, the vertical vanishing point is fixed and exhaustive search is performed for a pair of vanishing points which form a triangle with its orthocenter closest to the image center.

## A.2 Camera focal length, pitch and roll angles estimation

### A.2.1 Focal length estimation

From the three selected orthogonal vanishing points, the camera focal length, pitch and roll angles are calculated with a similar technique as in [40]. An uncalibrated camera model is first assumed:

$$\lambda x = \mathbf{K}[\mathbf{R} \quad \mathbf{T}]X \tag{A.1}$$

where $x$ is the pixel coordinate on an image of $X$ in the 3D space after perspective projection, and $\mathbf{K}$ is the intrinsic parameters of the camera, $\mathbf{R}$ and $\mathbf{T}$ are the extrinsic parameters and $\lambda$ is a scalar. Since three orthogonal vanishing points correspond to three orthogonal directions in a 3D space (Fig. 3.4), these directions are denoted as $e_i = [1,0,0,0]^T$, $e_j = [0,1,0,0]^T$ and $e_k = [0,0,1,0]^T$ in homogeneous coordinate. From (A.1), the vanishing points, $v_i$, $v_j$ and $v_k$ for the three directions are represented as:

$$\begin{aligned} v_i &= \mathbf{KR}[1,0,0]^T \\ v_j &= \mathbf{KR}[0,1,0]^T \\ v_k &= \mathbf{KR}[0,0,1]^T \end{aligned} \tag{A.2}$$

Since $e_i$ and $e_j$ are orthogonal to each other, a constraint can be derived on the focal length from $e_i^T e_j = 0$:

$$v_i^T \mathbf{K}^{-T}\mathbf{R}\mathbf{R}^T\mathbf{K}^{-1} v_j = v_i^T \mathbf{S} v_j = 0 \tag{A.3}$$

where $\mathbf{S} = \mathbf{K}^{-T}\mathbf{K}^{-1}$ is a symmetric matrix with only unknown focal length as its variable, assuming the principal point at the image center, zero skew and unit aspect ratio. Similarly, two more constraints listed below can be derived to solve for the focal length. The least squared error solution can be taken.

$$\begin{aligned} v_j^T \mathbf{S} v_k &= 0 \\ v_k^T \mathbf{S} v_i &= 0 \end{aligned} \tag{A.4}$$

### A.2.2 Pitch and roll angles estimation

After intrinsic parameter matrix **K** is uncovered, the following equation can be used to solve for the rotational matrix **R**:

$$\mathbf{K}^{-1}[v_i, v_j, v_k] = \mathbf{RI} \tag{A.5}$$

where **I** is the identity matrix formed by $[e_i, e_j, e_k]$. Since the resulting **R** typically does not belong to the special orthogonal group, a singular value decomposition on **R**: $\mathbf{R} = \mathbf{U}\Lambda\mathbf{V}^T$ is performed, and $\mathbf{R}' = \mathbf{U}\mathbf{V}^T$. It can be shown that $\mathbf{R}'$ is the closest unitary matrix to **R** in Frobenius norm. Once an unitary rotation matrix is obtained, $\mathbf{R}'$ is decomposed into three rotation angles. Since this is the rotation matrix from the camera coordinate to an arbitrary model coordinate, the yaw angle needs to be modified according to NAV420CA reading. Note that it is not necessary to change the pitch and roll angles because it is assumed that the vertical axis in the model coordinate is the same as that in the world reference frame.

# Appendix B

# Introduction to Quaternion

A brief introduction to quaternion including its definition, elementary operations and properties, is presented in this chapter. The material in this chapter is from [48].

A quaternion $\dot{q}$ can be represented as a vector of four components with one real part and three imaginary parts:

$$\dot{q} = q_0 + iq_x + jq_y + kq_z \tag{B.1}$$

where $q_0$, $q_x$, $q_y$, $q_z$ are real numbers. And the basis of the imaginary parts has the following properties:

$$\begin{aligned}
&i^2 = -1, j^2 = -1, k^2 = -1 \\
&ij = k, jk = i, ki = j \\
&ji = -k, kj = -i, ik = -j
\end{aligned} \tag{B.2}$$

The multiplication between two quaternions, $\dot{q}$ and $\dot{r}$ can be expressed as:

$$\begin{aligned}
\dot{r}\dot{q} &= (q_0 + iq_x + jq_y + kq_z)(r_0 + ir_x + jr_y + kr_z) \\
&= (r_0q_0 - r_xq_x - r_yq_y - r_zq_z) \\
&\quad + i(r_0q_x + r_xq_0 + r_yq_z - r_zq_y) \\
&\quad + j(r_0q_y - r_xq_x + r_yq_0 + r_zq_x) \\
&\quad + k(r_0q_z + r_xq_y - r_yq_x + r_zq_0)
\end{aligned} \tag{B.3}$$

This multiplication can also be written as a product between an orthogonal 4 × 4 matrix and a four-elements vector. It is possible to rearrange either the left or right quaternion into a matrix as shown below:

$$\dot{r}\dot{q} = \begin{bmatrix} r_0 & -r_x & -r_y & -r_z \\ r_x & -r_0 & -r_z & r_y \\ r_y & r_z & r_0 & -r_x \\ r_z & -r_y & r_x & r_0 \end{bmatrix} \dot{q} = \mathbf{R}\dot{q} \tag{B.4}$$

or

$$\dot{q}\dot{r} = \begin{bmatrix} r_0 & -r_x & -r_y & -r_z \\ r_x & r_0 & r_z & -r_y \\ r_y & -r_z & r_0 & r_x \\ r_z & r_y & -r_x & r_0 \end{bmatrix} \dot{q} = \bar{\mathbf{R}}\dot{q} \tag{B.5}$$

The difference between **R** and $\bar{\mathbf{R}}$ demonstrates that quaternion multiplication is not commutative.

A few other properties of quaternion operations are listed below:

$$\dot{q}^* = q_0 - iq_x - jq_y - kq_z \tag{B.6}$$

$$\dot{p} \cdot \dot{q} = (\dot{p}\dot{q}^* + \dot{q}\dot{p}^*)/2 \tag{B.7}$$

$$|\dot{p}|^2 = \dot{q} \cdot \dot{q} = \dot{q}\dot{q}^* = \dot{q}^*\dot{q} \tag{B.8}$$

$$\dot{p}^{-1} = (1/\dot{q} \cdot \dot{q})\dot{q}^* \tag{B.9}$$

Since the matrices associated with quaternions are orthogonal, dot products are preserved:

$$\begin{aligned} (\dot{q}\dot{p}) \cdot (\dot{q}\dot{r}) &= (\mathbf{Q}\dot{p}) \cdot (\mathbf{Q}\dot{r}) \\ &= (\mathbf{Q}\dot{p})^T(\mathbf{Q}\dot{r}) \\ &= \dot{p}^T\mathbf{Q}^T\mathbf{Q}\dot{r} \\ &= \dot{p}^T(\dot{q} \cdot \dot{q})\mathbf{I}\dot{r} \\ &= (\dot{q} \cdot \dot{q})(\dot{p} \cdot \dot{r}) \end{aligned} \tag{B.10}$$

In case $\dot{p}$ is a unit quaternion, (B.10) is equal to $\dot{p} \cdot \dot{r}$. A convenient property from this is:

$$(\dot{p}\dot{q}) \cdot \dot{r} = \dot{p} \cdot (\dot{r}\dot{q}^*) \tag{B.11}$$

# Appendix C

# Detailed Result Analysis Per Image

In the following table, area of each aerial image is classified into ”r”, ”d”, ”c” and ”x”, representing residential region, downtown, campus and erroneous image acquisition respectively. Reasons for incorrect camera pose recovery are classified into ”no”, ”random”, ”few corners” and ”x”, corresponding to no error, i.e. correct pose recovery, random pose from multiple solutions to an ill-posed problem, not enough correct 2DOC matches, and erroneous image acquisition respectively. Please refer to Section 6.1 for detailed definitions.

|  |  | NAV420CA Reading |  |  |  |  |  | Corner Correspondence |  |  |  |  |  | Manual Correspondence |  |  |  |  |  |  |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
|  | Area* | X | Y | Z | roll | pitch | yaw | X | Y | Z | roll | pitch | yaw | X | Y | Z | roll | pitch | yaw | source of error** |
|  |  | (m) | (m) | (m) | (°) | (°) | (°) | (m) | (m) | (m) | (°) | (°) | (°) | (m) | (m) | (m) | (°) | (°) | (°) |  |
| image 001 | r | 2454.0 | 4185.4 | 292.5 | -3.3 | 43.3 | 173.7 | 2466.4 | 4165.7 | 314.3 | -3.7 | 38.0 | 175.6 | 2451.6 | 4189.8 | 296.7 | -2.4 | 44.8 | 172.3 | random |
| image 002 | r | 2444.4 | 4204.2 | 296.5 | -5.4 | 39.2 | 168.0 | 2463.2 | 4219.1 | 290.4 | -11.2 | 36.6 | 165.5 | 2442.7 | 4205.7 | 298.6 | -7.4 | 37.5 | 162.4 | random |
| image 003 | r | 2442.8 | 4231.0 | 295.7 | 2.4 | 40.5 | 161.3 | 2386.8 | 4193.6 | 196.1 | -0.9 | 27.2 | 146.4 | 2444.9 | 4237.0 | 297.8 | 3.9 | 42.1 | 155.7 | random |
| image 004 | r | 2435.8 | 4297.2 | 300.5 | 3.4 | 40.2 | 161.9 | 2551.1 | 4401.7 | 306.9 | 14.1 | 31.5 | 178.2 | 2439.2 | 4307.6 | 303.7 | 4.3 | 42.2 | 158.1 | random |
| image 005 | r | 2420.6 | 4404.7 | 310.7 | -3.0 | 40.9 | 152.7 | 2419.7 | 4429.5 | 338.8 | -1.3 | 40.0 | 153.3 | 2422.3 | 4405.5 | 311.8 | -4.5 | 41.8 | 146.8 | few corners |
| image 006 | r | 2410.2 | 4538.7 | 316.7 | -2.4 | 39.3 | 154.0 | 2423.4 | 4540.7 | 329.2 | -0.8 | 34.9 | 153.0 | 2413.7 | 4543.3 | 316.9 | -2.2 | 39.6 | 150.9 | random |
| image 007 | r | 2403.9 | 4689.1 | 325.5 | -5.2 | 42.7 | 147.3 | 2388.5 | 4673.9 | 339.9 | -9.1 | 38.6 | 138.6 | 2409.2 | 4686.5 | 337.6 | -4.9 | 43.2 | 141.5 | no |
| image 008 | r | 2396.6 | 4855.7 | 331.2 | -1.9 | 39.9 | 154.3 | 2398.3 | 4849.4 | 323.1 | 4.7 | 35.6 | 149.7 | 2393.6 | 4861.8 | 328.0 | -1.9 | 40.0 | 147.4 | random |
| image 009 | r | 2389.2 | 5031.2 | 338.0 | -4.2 | 41.7 | 151.7 | 2389.1 | 5026.0 | 339.9 | -6.3 | 39.4 | 148.3 | 2388.1 | 5025.6 | 328.3 | -5.2 | 38.2 | 148.8 | no |
| image 010 | r | 2376.9 | 5212.2 | 342.7 | -0.2 | 33.7 | 161.3 | 2383.2 | 5214.5 | 342.1 | 0.5 | 32.6 | 157.7 | 2385.1 | 5228.2 | 348.9 | 0.9 | 33.7 | 159.1 | no |
| image 011 | r | 2364.3 | 5401.7 | 348.0 | -2.7 | 36.1 | 161.1 | 2382.2 | 5433.5 | 341.1 | -2.4 | 36.4 | 163.0 | 2354.0 | 5384.2 | 357.9 | -4.0 | 37.2 | 152.1 | few corners |
| image 012 | r | 2342.1 | 5592.3 | 354.5 | -9.2 | 43.7 | 166.3 | 2358.8 | 5582.8 | 337.9 | -13.1 | 39.6 | 162.4 | 2345.1 | 5597.2 | 360.2 | -9.3 | 45.8 | 163.8 | few corners |
| image 013 | r | 2247.6 | 5744.9 | 359.7 | -9.7 | 48.4 | -157.0 | 2252.8 | 5754.1 | 288.6 | -6.6 | 32.7 | -158.3 | 2251.0 | 5749.8 | 357.0 | -7.7 | 46.3 | -155.3 | random |
| image 014 | r | 2077.9 | 5810.0 | 357.5 | -4.9 | 43.1 | -115.4 | 2123.0 | 5763.6 | 352.5 | -18.9 | 42.3 | -123.7 | 2086.5 | 5806.7 | 353.1 | -6.6 | 40.3 | -119.8 | few corners |
| image 015 | r | 1884.8 | 5783.6 | 351.7 | 1.4 | 33.7 | -94.4 | 1885.8 | 5789.1 | 372.3 | -5.0 | 38.1 | -100.7 | 1905.3 | 5816.3 | 315.8 | -12.3 | 28.8 | -98.6 | few corners |
| image 016 | r | 1727.6 | 5680.8 | 351.5 | -14.6 | 40.7 | -49.3 | 1743.1 | 5687.2 | 321.0 | -29.4 | 42.8 | -49.8 | 1733.2 | 5673.9 | 352.2 | -15.0 | 42.7 | -44.2 | few corners |
| image 017 | x | 1795.7 | 5565.3 | 340.5 | -11.8 | 40.2 | 30.6 | 1798.1 | 5555.8 | 300.5 | -7.7 | 35.8 | 27.8 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 018 | x | 1966.8 | 5547.0 | 323.7 | 16.0 | -2.7 | 18.9 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 019 | x | 2039.3 | 5450.3 | 319.5 | 5.7 | 15.6 | -28.4 | 2036.9 | 5444.5 | 317.9 | 4.3 | 17.0 | -29.8 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 020 | r | 2050.5 | 5341.7 | 309.7 | 0.2 | 42.4 | -33.3 | 2051.6 | 5337.6 | 312.7 | -1.2 | 43.5 | -34.4 | 2053.2 | 5340.6 | 309.2 | -1.7 | 43.1 | -35.1 | no |
| image 021 | r | 2061.1 | 5172.4 | 301.7 | -3.1 | 43.9 | -36.6 | 2061.5 | 5174.7 | 298.8 | -3.9 | 43.2 | -36.8 | 2063.5 | 5175.2 | 301.0 | -4.6 | 43.3 | -37.3 | no |
| image 022 | r | 2071.1 | 4990.7 | 290.2 | -3.9 | 40.0 | -39.0 | 2070.1 | 4989.5 | 292.5 | -3.1 | 40.4 | -35.6 | 2068.6 | 4985.2 | 283.8 | -1.5 | 39.3 | -34.8 | no |
| image 023 | r | 2079.2 | 4806.8 | 278.2 | -2.2 | 39.9 | -36.4 | 2098.9 | 4804.2 | 250.6 | -11.5 | 33.8 | -45.1 | 2073.8 | 4807.2 | 272.8 | 0.5 | 38.1 | -33.8 | random |
| image 024 | r | 2088.8 | 4615.9 | 267.5 | -0.9 | 41.0 | -28.5 | 2088.7 | 4614.0 | 266.8 | 0.3 | 40.6 | -26.4 | 2092.1 | 4620.0 | 265.5 | -1.2 | 40.4 | -28.2 | no |
| image 025 | r | 2099.7 | 4421.6 | 257.5 | -0.7 | 40.0 | -27.1 | 2098.8 | 4422.8 | 262.8 | 0.5 | 41.3 | -25.7 | 2097.3 | 4424.2 | 254.5 | -0.5 | 39.7 | -25.8 | no |
| image 026 | r | 2111.9 | 4226.6 | 250.2 | -2.2 | 41.1 | -30.9 | 2113.0 | 4229.0 | 249.5 | -1.8 | 41.3 | -27.1 | 2117.7 | 4237.8 | 246.7 | -3.5 | 40.7 | -29.3 | no |
| image 027 | r | 2124.9 | 4037.3 | 240.0 | 1.5 | 39.0 | -25.7 | 2124.4 | 4036.6 | 240.8 | 2.4 | 38.0 | -20.6 | 2124.5 | 4042.5 | 234.4 | 1.4 | 36.7 | -22.1 | no |
| image 028 | r | 2137.6 | 3851.2 | 231.5 | 1.0 | 32.6 | -21.4 | 2140.0 | 3856.2 | 229.3 | 0.9 | 31.5 | -19.0 | 2137.7 | 3856.2 | 228.5 | 1.5 | 31.1 | -19.0 | no |
| image 029 | x | 2152.7 | 3669.4 | 225.7 | 1.3 | 32.8 | -30.8 | 2180.3 | 3676.0 | 258.6 | -1.3 | 37.2 | -34.0 | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 030 | x | 2117.0 | 3528.2 | 228.0 | 5.7 | 16.8 | -76.3 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 031 | x | 2039.7 | 3533.0 | 230.0 | 16.0 | 13.1 | -164.8 | 2034.5 | 3486.9 | 225.6 | 2.4 | 11.0 | -172.4 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 032 | d | 2051.8 | 3662.2 | 225.7 | 4.8 | 38.3 | 153.8 | 2050.9 | 3660.6 | 225.3 | 1.1 | 37.7 | 151.2 | 2057.9 | 3663.5 | 223.8 | 2.5 | 37.0 | 152.8 | no |
| image 033 | d | 2078.5 | 3822.9 | 226.7 | -1.5 | 39.2 | 149.9 | 2080.0 | 3822.3 | 228.3 | -1.5 | 38.4 | 149.0 | 2079.6 | 3821.0 | 227.1 | -2.2 | 38.3 | 148.7 | no |
| image 034 | d | 2088.7 | 4003.5 | 229.2 | -2.0 | 39.6 | 154.7 | 2091.3 | 4002.1 | 231.3 | -3.7 | 39.1 | 154.5 | 2091.9 | 4004.9 | 230.9 | -3.0 | 39.1 | 155.4 | no |
| image 035 | c | 2079.4 | 4200.2 | 229.2 | -1.0 | 35.1 | 155.5 | 2085.7 | 4203.2 | 229.6 | -0.8 | 35.8 | 154.8 | 2082.9 | 4197.1 | 233.8 | -2.6 | 36.7 | 153.0 | no |
| image 036 | c | 2070.9 | 4406.4 | 228.5 | -5.2 | 37.7 | 153.2 | 2070.9 | 4394.5 | 222.8 | -7.3 | 35.0 | 150.0 | 2074.4 | 4410.1 | 226.5 | -4.9 | 36.3 | 154.3 | no |
| image 037 | c | 2060.9 | 4614.1 | 231.0 | -5.9 | 36.2 | 156.1 | 2068.7 | 4613.4 | 227.2 | -9.2 | 34.0 | 157.0 | 2065.5 | 4607.7 | 234.1 | -5.9 | 35.7 | 154.2 | random |
| image 038 | c | 2049.7 | 4819.6 | 234.0 | -3.7 | 35.6 | 155.5 | 2055.3 | 4819.4 | 237.2 | -4.2 | 36.4 | 154.5 | 2056.2 | 4811.2 | 247.2 | -6.0 | 37.6 | 152.5 | no |
| image 039 | c | 2040.4 | 5028.5 | 235.0 | -4.6 | 36.1 | 158.7 | 2060.3 | 5041.7 | 231.1 | 0.8 | 34.2 | 162.2 | 2048.9 | 5031.0 | 234.8 | -4.6 | 36.0 | 157.8 | random |
| image 040 | c | 2029.6 | 5240.1 | 237.2 | -3.0 | 33.7 | 163.6 | 2022.3 | 5222.1 | 214.0 | -16.7 | 30.9 | 161.7 | 2033.1 | 5236.3 | 241.7 | -4.3 | 34.0 | 160.0 | no |
| image 041 | c | 2022.7 | 5445.7 | 247.5 | -6.3 | 32.0 | 156.5 | 2014.1 | 5449.7 | 247.6 | -8.1 | 35.2 | 156.9 | 2027.7 | 5444.2 | 253.4 | -6.5 | 31.8 | 153.0 | few corners |
| image 042 | c | 2017.2 | 5633.3 | 260.5 | -7.4 | 27.9 | 164.7 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 043 | c | 1983.7 | 5764.0 | 274.0 | -15.9 | 40.3 | -151.8 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 044 | r | 1919.2 | 5771.9 | 270.7 | -0.7 | 43.0 | -69.0 | 1959.9 | 5779.0 | 263.5 | -19.9 | 40.0 | -80.9 | 1920.2 | 5768.4 | 268.2 | -1.9 | 44.7 | -62.0 | few corners |
| image 045 | r | 1896.1 | 5682.8 | 261.2 | 5.1 | 36.7 | -24.9 | 1892.6 | 5691.0 | 254.3 | 0.4 | 31.6 | -27.0 | 1898.7 | 5684.7 | 266.6 | 1.5 | 38.2 | -29.0 | few corners |
| image 046 | r | 1901.8 | 5529.5 | 252.7 | 2.6 | 39.1 | -31.9 | 1911.6 | 5544.9 | 251.7 | 0.9 | 34.4 | -40.8 | 1906.7 | 5521.9 | 234.3 | 4.1 | 33.3 | -34.8 | few corners |
| image 047 | r | 1907.9 | 5355.7 | 249.5 | 0.1 | 37.7 | -34.7 | 1909.0 | 5356.5 | 252.8 | -1.1 | 37.8 | -35.1 | 1910.2 | 5356.1 | 252.0 | -1.4 | 37.8 | -35.3 | no |
| image 048 | r | 1912.9 | 5171.6 | 245.0 | -0.7 | 36.8 | -29.4 | 1911.2 | 5168.5 | 246.7 | 0.1 | 37.8 | -29.4 | 1911.6 | 5175.1 | 245.2 | -1.0 | 36.9 | -31.3 | no |
| image 049 | r | 1916.0 | 4981.4 | 239.7 | -1.8 | 37.3 | -30.8 | 1914.0 | 4983.1 | 239.9 | -1.9 | 36.4 | -31.4 | 1924.0 | 4993.5 | 248.2 | -4.7 | 38.2 | -35.1 | no |
| image 050 | r | 1919.7 | 4786.1 | 239.2 | -2.0 | 38.5 | -30.5 | 1921.5 | 4782.5 | 224.6 | -4.5 | 32.8 | -34.6 | 1915.7 | 4786.8 | 240.3 | -1.3 | 38.7 | -29.4 | random |
| image 051 | r | 1926.3 | 4580.8 | 237.0 | -3.7 | 40.9 | -30.7 | 1927.4 | 4584.8 | 238.8 | -2.8 | 40.2 | -29.7 | 1925.8 | 4584.1 | 237.9 | -2.3 | 40.1 | -29.3 | no |
| image 052 | r | 1938.2 | 4373.9 | 236.2 | -6.4 | 38.1 | -32.4 | 1938.4 | 4372.8 | 238.3 | -3.7 | 40.8 | -29.2 | 1938.3 | 4377.1 | 232.6 | -4.3 | 39.7 | -30.1 | no |
| image 053 | r | 1953.2 | 4170.5 | 234.5 | -3.4 | 39.0 | -29.9 | 1952.4 | 4173.5 | 244.0 | -3.1 | 40.4 | -27.4 | 1952.8 | 4172.1 | 234.6 | -3.2 | 39.1 | -27.5 | no |
| image 054 | r | 1974.1 | 3964.4 | 232.0 | -3.9 | 40.4 | -29.6 | 1977.6 | 3967.4 | 234.3 | -3.8 | 39.6 | -26.7 | 1973.2 | 3966.4 | 229.8 | -3.2 | 38.4 | -26.0 | no |
| image 055 | x | 1996.8 | 3758.3 | 229.7 | -7.1 | 37.4 | -30.6 | 1987.2 | 3765.8 | 226.3 | -7.8 | 36.6 | -27.2 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 056 | x | 1984.7 | 3587.7 | 234.7 | 3.4 | 20.6 | -59.1 | 2327.2 | 3772.7 | 258.9 | -22.6 | 8.7 | -113.1 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 057 | x | 1896.9 | 3556.0 | 241.5 | 16.2 | 14.9 | -146.4 | 1898.9 | 3546.0 | 271.9 | 10.5 | 24.6 | -143.2 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 058 | d | 1884.1 | 3682.5 | 236.7 | 1.1 | 36.7 | 156.5 | 1885.5 | 3688.0 | 239.8 | -0.3 | 37.8 | 154.3 | 1886.0 | 3687.8 | 240.0 | -0.6 | 37.6 | 154.3 | no |
| image 059 | d | 1896.4 | 3840.7 | 238.0 | -7.4 | 38.5 | 153.5 | 1898.7 | 3838.8 | 239.9 | -8.7 | 38.2 | 152.1 | 1897.8 | 3844.7 | 247.9 | -8.0 | 39.8 | 153.4 | no |
| image 060 | d | 1894.1 | 4016.0 | 243.7 | -5.4 | 36.7 | 157.6 | 1894.1 | 4016.3 | 245.6 | -5.3 | 36.0 | 153.5 | 1894.4 | 4015.7 | 247.2 | -5.3 | 36.3 | 153.5 | no |
| image 061 | d | 1883.2 | 4209.9 | 252.2 | -8.8 | 37.8 | 150.5 | 1883.9 | 4211.6 | 253.2 | -9.5 | 37.5 | 148.1 | 1883.5 | 4212.2 | 257.9 | -8.9 | 38.6 | 148.1 | no |
| image 062 | c | 1871.7 | 4414.2 | 261.2 | -9.9 | 39.8 | 150.6 | 1894.4 | 4419.9 | 194.8 | -0.7 | 23.5 | 153.2 | 1881.7 | 4417.8 | 261.8 | -7.8 | 38.3 | 151.8 | few corners |
| image 063 | c | 1855.2 | 4624.2 | 266.2 | -8.1 | 38.2 | 154.8 | 1856.2 | 4619.3 | 268.5 | -9.7 | 39.4 | 152.4 | 1861.1 | 4620.4 | 273.2 | -8.8 | 39.8 | 153.2 | no |
| image 064 | c | 1837.6 | 4842.8 | 268.5 | -0.1 | 32.3 | 168.6 | 1834.8 | 4847.3 | 270.4 | -4.7 | 31.0 | 172.9 | 1840.3 | 4836.5 | 272.6 | -0.7 | 31.0 | 166.7 | few corners |
| image 065 | c | 1819.6 | 5062.0 | 270.7 | -6.5 | 38.3 | 158.5 | 1816.4 | 5061.2 | 265.9 | -8.8 | 37.5 | 153.6 | 1805.6 | 5056.2 | 259.5 | -8.2 | 36.9 | 152.8 | no |
| image 066 | c | 1803.2 | 5282.2 | 275.2 | -7.2 | 33.8 | 154.1 | 1794.5 | 5230.2 | 261.4 | -15.7 | 26.4 | 136.4 | 1807.1 | 5271.5 | 277.9 | -8.5 | 33.4 | 149.7 | few corners |
| image 067 | x | 1790.3 | 5497.8 | 282.7 | -9.4 | 33.6 | 150.7 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 068 | x | 1762.3 | 5680.9 | 291.2 | -26.3 | 39.5 | 178.3 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 069 | x | 1683.6 | 5709.6 | 290.5 | -8.1 | 51.6 | -76.4 | 1702.2 | 5737.7 | 276.4 | -1.3 | 40.4 | -80.4 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 070 | c | 1664.1 | 5577.2 | 273.5 | 2.8 | 39.3 | -26.2 | 1695.9 | 5584.0 | 286.2 | -12.0 | 47.4 | -30.2 | 1671.1 | 5578.4 | 270.6 | -0.4 | 40.6 | -28.9 | few corners |
| image 071 | c | 1675.0 | 5401.5 | 267.2 | -3.0 | 40.2 | -33.2 | 1670.8 | 5398.8 | 270.2 | -2.1 | 40.2 | -32.9 | 1670.0 | 5400.1 | 272.7 | -2.2 | 40.5 | -33.3 | no |
| image 072 | c | 1688.9 | 5210.6 | 257.2 | -2.8 | 40.4 | -35.4 | 1689.4 | 5208.3 | 257.8 | -3.1 | 40.8 | -35.1 | 1685.5 | 5210.1 | 257.8 | -2.6 | 40.2 | -34.9 | no |
| image 073 | c | 1698.6 | 5012.6 | 250.0 | -5.9 | 39.7 | -33.3 | 1690.0 | 5006.8 | 245.9 | -3.8 | 38.1 | -28.8 | 1695.3 | 5010.5 | 248.9 | -5.3 | 39.1 | -30.5 | no |
| image 074 | c | 1709.8 | 4811.0 | 241.5 | -4.1 | 42.4 | -33.5 | 1730.9 | 4803.9 | 226.4 | -11.9 | 39.1 | -39.8 | 1706.0 | 4809.0 | 238.1 | -5.1 | 41.5 | -34.1 | few corners |
| image 075 | c | 1727.2 | 4603.4 | 233.7 | -4.0 | 40.9 | -34.9 | 1728.4 | 4600.5 | 232.3 | -4.5 | 40.8 | -35.6 | 1725.9 | 4593.7 | 233.5 | -3.0 | 41.3 | -33.6 | no |
| image 076 | c | 1740.4 | 4394.4 | 228.5 | -4.1 | 38.9 | -33.6 | 1734.0 | 4387.3 | 229.3 | -2.2 | 38.4 | -29.2 | 1739.5 | 4396.5 | 227.5 | -3.3 | 38.1 | -31.4 | no |

*c: campus
r: residential area
d: downtown
x: erroneous data acquisition

**x: erroneous data acquisition
no: correct pose

| | | NAV420CA Reading | | | | | | Corner Correspondence | | | | | | Manual Correspondence | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Area* | X | Y | Z | roll | pitch | yaw | X | Y | Z | roll | pitch | yaw | X | Y | Z | roll | pitch | yaw | source of error** |
| | | (m) | (m) | (m) | (˚) | (˚) | (˚) | (m) | (m) | (m) | (˚) | (˚) | (˚) | (m) | (m) | (m) | (˚) | (˚) | (˚) | |
| image 077 | d | 1751.4 | 4190.2 | 221.2 | -2.9 | 39.8 | -30.9 | 1750.9 | 4191.6 | 223.7 | -2.6 | 41.3 | -29.1 | 1750.2 | 4193.6 | 217.9 | -3.2 | 40.1 | -29.6 | no |
| image 078 | d | 1769.4 | 3985.4 | 213.7 | -3.7 | 36.2 | -30.6 | 1770.5 | 3987.8 | 213.1 | -3.0 | 36.4 | -28.6 | 1772.8 | 3996.3 | 203.2 | -6.9 | 35.1 | -31.1 | no |
| image 079 | d | 1787.3 | 3786.6 | 207.7 | -6.7 | 35.5 | -34.1 | 1785.3 | 3787.6 | 210.8 | -5.0 | 35.7 | -29.0 | 1786.4 | 3785.6 | 217.1 | -3.6 | 37.0 | -28.9 | no |
| image 080 | x | 1791.3 | 3626.1 | 207.7 | -0.2 | 12.4 | -52.8 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 081 | x | 1744.8 | 3561.7 | 213.2 | 7.4 | 6.8 | -123.5 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 082 | d | 1723.2 | 3599.7 | 213.2 | 11.4 | 37.4 | 164.3 | 1727.8 | 3607.2 | 208.6 | 12.2 | 39.0 | 164.3 | 1724.2 | 3612.1 | 211.7 | 12.6 | 40.1 | 165.0 | no |
| image 083 | d | 1721.0 | 3711.7 | 213.7 | -1.9 | 42.0 | 145.9 | 1721.7 | 3716.7 | 220.0 | -1.8 | 43.0 | 147.2 | 1722.5 | 3717.0 | 213.9 | -1.6 | 41.9 | 147.4 | no |
| image 084 | d | 1710.6 | 3863.5 | 215.0 | -0.1 | 42.3 | 147.7 | 1711.9 | 3864.3 | 218.1 | -1.4 | 42.7 | 146.2 | 1710.0 | 3862.4 | 213.0 | -2.8 | 41.8 | 145.3 | no |
| image 085 | d | 1696.4 | 4035.2 | 219.0 | 1.7 | 39.3 | 151.1 | 1698.0 | 4036.5 | 222.2 | -0.1 | 40.1 | 150.3 | 1698.8 | 4036.1 | 222.3 | 0.2 | 40.1 | 150.5 | no |
| image 086 | d | 1677.6 | 4218.2 | 226.5 | 4.8 | 35.7 | 152.3 | 1682.0 | 4214.0 | 238.9 | 2.5 | 37.6 | 150.9 | 1672.6 | 4211.7 | 225.6 | 0.1 | 36.1 | 150.0 | no |
| image 087 | c | 1669.4 | 4420.3 | 229.5 | -3.2 | 38.7 | 149.6 | 1663.7 | 4426.9 | 214.4 | -4.7 | 30.9 | 143.5 | 1676.0 | 4421.5 | 233.6 | -1.8 | 37.2 | 151.0 | few corners |
| image 088 | c | 1654.1 | 4629.4 | 235.7 | -1.7 | 36.0 | 151.7 | 1649.9 | 4619.6 | 197.9 | -11.5 | 30.8 | 145.6 | 1655.5 | 4635.5 | 235.6 | -1.8 | 37.7 | 152.3 | few corners |
| image 089 | c | 1631.6 | 4838.9 | 241.7 | -3.0 | 38.3 | 152.2 | 1622.5 | 4826.0 | 232.6 | -0.1 | 34.5 | 143.1 | 1634.1 | 4837.4 | 248.4 | -4.3 | 39.5 | 150.5 | random |
| image 090 | c | 1610.2 | 5042.7 | 246.0 | -1.3 | 41.3 | 157.4 | 1613.7 | 5047.3 | 250.5 | -0.4 | 40.0 | 154.1 | 1616.5 | 5043.4 | 247.8 | -1.5 | 41.7 | 156.3 | few corners |
| image 091 | c | 1589.2 | 5241.9 | 249.2 | -1.2 | 39.9 | 158.1 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 092 | x | 1571.4 | 5427.0 | 258.5 | -17.3 | 24.9 | 162.6 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 093 | c | 1500.8 | 5505.9 | 271.7 | -17.8 | 44.3 | -132.8 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 094 | x | 1417.0 | 5425.0 | 263.0 | -9.6 | 41.0 | -56.1 | 1419.0 | 5437.4 | 241.5 | -13.1 | 33.8 | -54.4 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 095 | x | 1423.0 | 5271.8 | 245.2 | -2.3 | 31.2 | -34.9 | 1416.7 | 5268.9 | 252.8 | -2.4 | 30.3 | -33.8 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 096 | c | 1451.3 | 5109.6 | 234.5 | 4.2 | 37.5 | -25.3 | 1450.1 | 5109.7 | 225.0 | 5.3 | 33.1 | -26.9 | 1454.8 | 5112.1 | 237.5 | 4.5 | 38.7 | -26.4 | few corners |
| image 097 | c | 1473.3 | 4962.1 | 236.0 | 3.6 | 38.8 | -25.5 | 1463.3 | 4982.6 | 245.8 | -5.5 | 35.1 | -24.8 | 1474.7 | 4962.3 | 239.2 | 3.0 | 38.1 | -25.6 | random |
| image 098 | c | 1489.6 | 4830.5 | 239.7 | 0.6 | 40.6 | -28.8 | 1466.3 | 4806.4 | 265.3 | 1.2 | 37.7 | -24.3 | 1491.1 | 4833.2 | 240.8 | 0.9 | 39.9 | -30.0 | random |
| image 099 | c | 1509.6 | 4683.2 | 240.2 | 2.2 | 42.6 | -27.5 | 1509.7 | 4683.3 | 246.6 | 3.0 | 41.9 | -25.3 | 1510.4 | 4682.8 | 245.1 | 2.8 | 41.8 | -25.3 | no |
| image 100 | c | 1532.2 | 4518.8 | 242.0 | 4.5 | 37.9 | -25.3 | 1529.1 | 4522.5 | 254.4 | 3.5 | 36.5 | -22.3 | 1533.3 | 4514.4 | 251.1 | 5.1 | 36.4 | -20.8 | no |
| image 101 | d | 1549.2 | 4346.4 | 241.7 | 1.8 | 39.4 | -28.3 | 1547.7 | 4347.7 | 249.0 | 2.9 | 39.8 | -25.5 | 1548.4 | 4351.7 | 239.4 | 0.8 | 38.2 | -26.8 | no |
| image 102 | d | 1569.2 | 4169.7 | 238.0 | -1.7 | 37.8 | -28.3 | 1570.0 | 4169.7 | 241.1 | -1.1 | 38.6 | -27.2 | 1570.5 | 4170.5 | 241.1 | -1.2 | 38.6 | -27.4 | no |
| image 103 | d | 1587.4 | 3992.1 | 230.5 | -0.2 | 37.4 | -29.1 | 1587.3 | 3994.0 | 231.6 | 0.6 | 38.5 | -26.2 | 1587.6 | 3993.9 | 233.7 | 0.2 | 38.7 | -26.4 | no |
| image 104 | d | 1604.2 | 3812.9 | 224.7 | -3.0 | 40.7 | -29.7 | 1599.3 | 3809.5 | 225.4 | -0.9 | 40.2 | -24.5 | 1606.1 | 3816.0 | 227.8 | -4.1 | 40.9 | -27.0 | no |
| image 105 | x | 1616.5 | 3644.0 | 222.2 | 1.9 | 23.2 | -44.5 | 1598.6 | 3632.4 | 224.6 | 3.8 | 18.5 | -41.9 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 106 | x | 1557.2 | 3566.7 | 225.0 | 18.0 | 13.5 | -125.1 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 107 | d | 1513.2 | 3653.5 | 218.2 | 5.3 | 41.3 | 156.9 | 1517.3 | 3656.4 | 219.1 | 3.7 | 39.9 | 155.3 | 1519.1 | 3651.0 | 215.2 | 2.1 | 38.9 | 154.1 | no |
| image 108 | d | 1502.8 | 3790.7 | 224.5 | -2.1 | 40.8 | 148.1 | 1504.3 | 3792.1 | 227.9 | -4.3 | 40.4 | 148.2 | 1506.5 | 3791.0 | 226.4 | -4.1 | 39.8 | 148.3 | no |
| image 109 | d | 1492.2 | 3943.9 | 229.7 | -0.4 | 40.7 | 148.7 | 1493.1 | 3950.5 | 225.6 | -4.8 | 36.4 | 149.3 | 1493.7 | 3941.1 | 229.6 | -1.6 | 40.3 | 148.3 | random |
| image 110 | d | 1477.8 | 4112.0 | 234.0 | -3.4 | 39.1 | 149.6 | 1483.2 | 4111.0 | 235.7 | -2.5 | 39.2 | 150.4 | 1482.7 | 4111.7 | 241.5 | -2.3 | 40.4 | 150.3 | no |
| image 111 | c | 1458.1 | 4283.6 | 239.7 | -2.3 | 41.4 | 153.5 | 1469.8 | 4294.8 | 236.4 | 4.3 | 39.9 | 158.5 | 1463.1 | 4285.0 | 246.7 | 0.3 | 42.6 | 155.3 | no |
| image 112 | c | 1431.6 | 4464.9 | 243.2 | 0.4 | 39.2 | 152.6 | 1418.6 | 4455.1 | 262.7 | -6.1 | 43.7 | 146.3 | 1433.3 | 4460.9 | 246.1 | -2.0 | 38.5 | 151.2 | random |
| image 113 | c | 1410.5 | 4652.6 | 246.7 | -2.7 | 39.8 | 157.2 | 1410.1 | 4648.2 | 256.4 | -4.9 | 41.5 | 154.9 | 1413.8 | 4655.8 | 250.5 | -2.0 | 40.8 | 157.7 | few corners |
| image 114 | c | 1376.8 | 4838.4 | 248.0 | -2.2 | 34.2 | 158.7 | 1378.9 | 4832.9 | 249.8 | -3.3 | 34.5 | 156.6 | 1380.6 | 4841.7 | 247.4 | -1.9 | 34.3 | 158.8 | no |
| image 115 | c | 1341.9 | 5018.7 | 251.0 | -1.6 | 33.9 | 162.2 | 1346.7 | 5026.6 | 243.0 | 0.5 | 31.7 | 163.8 | 1346.2 | 5017.8 | 251.4 | -0.6 | 32.9 | 161.3 | no |
| image 116 | c | 1311.5 | 5185.8 | 257.7 | -6.0 | 31.1 | 162.9 | 1332.1 | 5173.2 | 236.3 | -3.3 | 28.3 | 156.7 | 1314.6 | 5178.4 | 255.8 | -8.3 | 30.1 | 159.3 | few corners |
| image 117 | c | 1276.4 | 5298.6 | 261.5 | -18.7 | 44.9 | -171.6 | 1272.8 | 5271.4 | 239.3 | -37.1 | 39.8 | 176.3 | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 118 | c | 1218.6 | 5309.5 | 260.7 | -1.7 | 56.0 | -85.5 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 119 | c | 1188.4 | 5220.2 | 253.0 | 4.1 | 41.4 | -20.5 | 1180.4 | 5215.8 | 249.1 | 2.4 | 39.1 | -21.2 | 1195.5 | 5225.7 | 244.7 | 3.1 | 37.9 | -22.1 | few corners |
| image 120 | c | 1199.3 | 5076.4 | 246.2 | 3.0 | 33.7 | -13.7 | 1202.4 | 5079.0 | 246.2 | 3.4 | 34.0 | -17.3 | 1198.5 | 5075.8 | 249.4 | 4.3 | 34.4 | -16.2 | no |
| image 121 | c | 1216.3 | 4916.1 | 240.2 | 1.2 | 34.4 | -15.4 | 1217.1 | 4916.4 | 242.0 | 1.7 | 34.3 | -17.1 | 1215.8 | 4917.8 | 242.6 | 1.3 | 34.3 | -17.4 | no |
| image 122 | c | 1227.2 | 4744.6 | 236.0 | -3.3 | 36.2 | -21.0 | 1224.2 | 4734.4 | 241.5 | 0.0 | 36.6 | -19.3 | 1229.1 | 4745.4 | 239.6 | -4.2 | 36.6 | -22.7 | few corners |
| image 123 | c | 1234.0 | 4573.4 | 238.0 | -3.4 | 34.1 | -21.9 | 1239.3 | 4575.7 | 229.6 | -4.0 | 33.3 | -21.1 | 1235.5 | 4576.0 | 238.2 | -4.0 | 34.5 | -20.5 | random |
| image 124 | c | 1241.8 | 4404.4 | 234.7 | -3.9 | 38.4 | -20.3 | 1240.8 | 4405.4 | 234.4 | -3.8 | 39.5 | -19.1 | 1239.8 | 4405.3 | 234.6 | -3.9 | 39.4 | -19.1 | no |
| image 125 | d | 1254.6 | 4229.4 | 230.2 | -4.4 | 38.7 | -18.9 | 1254.4 | 4233.2 | 228.8 | -4.7 | 40.3 | -17.5 | 1261.8 | 4235.0 | 234.5 | -4.9 | 41.9 | -18.4 | no |
| image 126 | d | 1269.4 | 4048.7 | 224.0 | -2.1 | 37.8 | -19.4 | 1270.8 | 4050.7 | 225.2 | -3.7 | 38.8 | -17.4 | 1269.9 | 4051.1 | 224.7 | -3.8 | 38.7 | -17.4 | no |
| image 127 | d | 1287.1 | 3869.7 | 219.0 | -5.4 | 38.5 | -20.9 | 1286.1 | 3873.0 | 224.4 | -5.7 | 40.2 | -19.8 | 1287.6 | 3873.8 | 224.1 | -5.9 | 40.0 | -20.3 | no |
| image 128 | x | 1302.2 | 3708.3 | 220.0 | -5.0 | 28.7 | -28.9 | 1295.7 | 3709.4 | 220.1 | -2.3 | 24.1 | -27.5 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 129 | x | 1261.5 | 3600.5 | 228.7 | -7.4 | 14.7 | -74.9 | 1340.9 | 3589.4 | 150.3 | -15.7 | 10.6 | -87.3 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 130 | x | 1205.1 | 3618.1 | 235.7 | 0.4 | 27.0 | -172.9 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 131 | r | 1202.6 | 3724.6 | 236.5 | 0.1 | 36.3 | 160.2 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 132 | r | 1199.2 | 3865.8 | 237.5 | -2.5 | 36.5 | 159.1 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 133 | r | 1196.0 | 4025.6 | 237.2 | -8.8 | 34.5 | 156.2 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 134 | r | 1188.2 | 4188.1 | 242.2 | -8.9 | 37.8 | 161.6 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 135 | r | 1173.1 | 4358.8 | 249.5 | -7.1 | 37.0 | 161.1 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 136 | r | 1154.5 | 4534.5 | 252.0 | -6.0 | 36.3 | 163.3 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 137 | r | 1137.5 | 4715.4 | 253.5 | -9.2 | 40.0 | 161.8 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 138 | r | 1116.9 | 4884.8 | 256.5 | -5.5 | 35.0 | 164.4 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 139 | r | 1102.3 | 5036.1 | 261.7 | -5.4 | 35.5 | 161.5 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 140 | r | 1082.1 | 5189.5 | 274.2 | -18.0 | 36.3 | 163.4 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 141 | r | 1034.3 | 5270.4 | 283.2 | -29.6 | 45.4 | -136.5 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 142 | r | 1000.6 | 5202.2 | 274.2 | 3.2 | 45.6 | -33.7 | 983.8 | 5192.9 | 278.6 | 4.3 | 41.3 | -26.0 | 1002.9 | 5200.8 | 274.8 | 2.6 | 44.9 | -33.1 | random |
| image 143 | c | 1006.1 | 5064.4 | 269.5 | -1.1 | 35.4 | -30.5 | 1006.5 | 5066.7 | 272.3 | -2.5 | 35.4 | -31.2 | 1008.2 | 5072.7 | 267.0 | -5.4 | 34.4 | -32.8 | no |
| image 144 | c | 1015.8 | 4907.1 | 264.7 | -2.3 | 33.5 | -32.4 | 1018.6 | 4909.0 | 263.5 | -0.4 | 31.5 | -32.3 | 1018.9 | 4908.6 | 269.8 | -0.8 | 32.2 | -32.6 | no |
| image 145 | c | 1024.5 | 4736.7 | 261.5 | 3.8 | 33.8 | -30.9 | 1020.9 | 4742.3 | 268.6 | 2.4 | 33.3 | -30.9 | 1023.8 | 4738.7 | 262.4 | 2.3 | 32.8 | -30.8 | no |
| image 146 | c | 1033.2 | 4551.3 | 255.0 | 0.5 | 36.0 | -29.2 | 1035.0 | 4553.1 | 258.9 | -0.1 | 36.4 | -29.7 | 1036.7 | 4557.2 | 262.2 | -1.6 | 36.8 | -30.8 | no |
| image 147 | x | 1041.5 | 4360.3 | 249.2 | -6.7 | 25.9 | -42.8 | 1040.8 | 4356.1 | 250.3 | -4.9 | 26.5 | -39.5 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 148 | x | 1056.6 | 4176.1 | 246.2 | -1.5 | 22.8 | -35.9 | 1057.8 | 4176.2 | 249.3 | 0.0 | 23.6 | -37.0 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 149 | x | 1066.6 | 3997.1 | 240.0 | -0.8 | 20.7 | -39.7 | 1073.7 | 3994.1 | 239.5 | 0.5 | 24.5 | -37.4 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 150 | r | 1076.0 | 3828.0 | 237.5 | -1.5 | 37.2 | -28.4 | 1075.0 | 3845.1 | 240.6 | -4.2 | 36.0 | -29.7 | 1078.7 | 3830.9 | 243.3 | -1.2 | 38.0 | -26.9 | no |
| image 151 | r | 1085.4 | 3666.9 | 236.7 | -3.6 | 38.4 | -26.9 | 1088.2 | 3669.2 | 240.6 | -4.0 | 38.0 | -26.7 | 1084.6 | 3676.1 | 233.0 | -6.4 | 36.1 | -27.9 | no |
| image 152 | r | 1096.8 | 3534.9 | 245.7 | -12.0 | 34.5 | -8.7 | 1119.4 | 3502.7 | 229.1 | -19.0 | 27.9 | 0.0 | 1096.4 | 3548.3 | 241.0 | -13.3 | 33.3 | -6.1 | random |

*c: campus
r: residential area
d: downtown
x: erroneous data acquisition

**x: erroneous data acquisition
no: correct pose

| | | NAV420CA Reading | | | | | | Corner Correspondence | | | | | | Manual Correspondence | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Area* | X | Y | Z | roll | pitch | yaw | X | Y | Z | roll | pitch | yaw | X | Y | Z | roll | pitch | yaw | source of error** |
| | | (m) | (m) | (m) | (˚) | (˚) | (˚) | (m) | (m) | (m) | (˚) | (˚) | (˚) | (m) | (m) | (m) | (˚) | (˚) | (˚) | |
| image 153 | r | 1123.3 | 3484.8 | 255.0 | -5.5 | 38.9 | 54.6 | 1117.4 | 3494.7 | 251.7 | -4.4 | 35.5 | 56.4 | 1135.2 | 3487.1 | 257.9 | -2.5 | 36.5 | 59.7 | few corners |
| image 154 | r | 1153.6 | 3491.2 | 258.0 | 5.4 | 30.8 | 73.7 | 1164.3 | 3493.1 | 255.9 | 9.8 | 31.7 | 74.7 | 1169.6 | 3493.4 | 256.2 | 9.7 | 31.8 | 75.1 | no |
| image 155 | r | 1208.2 | 3493.4 | 260.0 | -6.9 | 35.4 | 59.3 | 1214.5 | 3497.5 | 262.6 | -2.1 | 35.8 | 63.4 | 1211.9 | 3497.6 | 262.2 | -2.8 | 35.7 | 62.9 | no |
| image 156 | r | 1286.1 | 3504.6 | 264.5 | 3.4 | 32.3 | 70.7 | 1290.0 | 3509.2 | 265.5 | 2.5 | 31.1 | 75.5 | 1287.7 | 3504.2 | 265.1 | 2.4 | 30.6 | 75.3 | no |
| image 157 | c | 1350.0 | 3540.2 | 267.2 | -9.5 | 32.5 | 138.2 | 1350.9 | 3544.6 | 266.8 | -8.7 | 29.9 | 144.5 | N/A | N/A | N/A | N/A | N/A | N/A | no |
| image 158 | c | 1337.2 | 3570.7 | 263.2 | -4.6 | 40.9 | 152.7 | 1342.3 | 3629.8 | 294.6 | 3.1 | 43.2 | 169.1 | N/A | N/A | N/A | N/A | N/A | N/A | random |
| image 159 | x | 1283.8 | 3567.4 | 261.7 | 3.9 | 27.7 | 153.4 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 160 | c | 1229.6 | 3565.0 | 264.0 | 3.8 | 38.0 | 154.3 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 161 | c | 1186.6 | 3573.4 | 264.5 | 5.6 | 30.2 | 165.1 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 162 | d | 1170.8 | 3595.7 | 262.2 | 8.3 | 34.4 | 62.5 | 1117.8 | 3618.6 | 243.0 | 4.5 | 32.6 | 50.6 | 1178.5 | 3605.7 | 259.0 | 9.8 | 36.3 | 61.5 | random |
| image 163 | d | 1211.5 | 3602.8 | 256.5 | 5.1 | 38.2 | 57.6 | 1219.0 | 3606.0 | 264.3 | 3.1 | 39.8 | 56.0 | 1223.3 | 3605.7 | 254.3 | 4.2 | 38.6 | 56.7 | no |
| image 164 | d | 1307.7 | 3600.7 | 258.2 | 1.8 | 38.0 | 55.7 | 1312.2 | 3604.1 | 263.6 | 2.9 | 37.5 | 57.7 | 1313.2 | 3602.7 | 261.8 | 3.2 | 37.2 | 58.0 | no |
| image 165 | d | 1439.8 | 3600.4 | 256.7 | -0.5 | 30.7 | 56.7 | 1444.4 | 3602.5 | 258.8 | -1.0 | 30.9 | 57.4 | 1449.8 | 3603.4 | 261.2 | -1.1 | 31.3 | 57.8 | no |
| image 166 | d | 1591.3 | 3601.4 | 253.7 | 3.6 | 30.7 | 58.7 | 1593.5 | 3602.9 | 257.4 | 4.2 | 31.2 | 58.7 | 1593.0 | 3603.7 | 254.1 | 3.9 | 30.9 | 58.6 | no |
| image 167 | d | 1757.1 | 3600.3 | 250.0 | 3.1 | 26.9 | 60.5 | 1757.6 | 3603.9 | 253.4 | 4.0 | 26.6 | 57.5 | 1760.3 | 3603.2 | 249.8 | 4.5 | 26.3 | 58.0 | no |
| image 168 | x | 1928.5 | 3596.8 | 247.5 | 2.8 | 22.0 | 57.6 | 1931.3 | 3598.2 | 246.4 | 2.5 | 21.6 | 58.9 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 169 | x | 2077.5 | 3600.7 | 247.7 | -4.7 | 26.6 | 66.9 | 2072.0 | 3606.3 | 249.5 | -9.3 | 31.3 | 68.1 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 170 | d | 2177.7 | 3649.6 | 251.5 | 3.3 | 34.0 | 118.9 | 2176.1 | 3650.2 | 255.6 | 3.4 | 33.9 | 122.7 | 2170.0 | 3649.7 | 244.2 | 3.7 | 32.8 | 122.1 | no |
| image 171 | d | 2228.2 | 3747.4 | 250.7 | -0.5 | 28.7 | 160.2 | 2228.0 | 3747.7 | 251.7 | 0.3 | 28.5 | 165.3 | 2227.6 | 3747.9 | 248.2 | 0.6 | 28.2 | 165.4 | no |
| image 172 | r | 2189.3 | 3834.5 | 245.0 | 0.6 | 42.5 | -135.8 | 2199.1 | 3836.4 | 223.9 | -0.1 | 38.1 | -133.5 | 2191.5 | 3830.8 | 251.0 | 1.7 | 42.9 | -132.6 | no |
| image 173 | r | 2068.9 | 3847.4 | 241.2 | 1.4 | 37.3 | -96.3 | 2073.2 | 3847.7 | 243.4 | 2.2 | 35.6 | -97.8 | 2074.1 | 3848.3 | 243.3 | 1.6 | 35.6 | -98.0 | few corners |
| image 174 | r | 1917.4 | 3820.9 | 241.7 | -0.9 | 33.6 | -108.7 | 1931.8 | 3836.5 | 249.0 | -5.8 | 31.7 | -108.8 | 1917.2 | 3824.0 | 245.2 | 0.8 | 32.7 | -108.1 | no |
| image 175 | r | 1753.1 | 3798.4 | 244.0 | -4.5 | 37.0 | -117.1 | 1757.5 | 3801.5 | 251.7 | -3.1 | 36.7 | -118.8 | 1760.4 | 3796.0 | 249.0 | -4.9 | 36.9 | -120.1 | no |
| image 176 | r | 1580.4 | 3780.7 | 245.7 | -4.6 | 39.0 | -117.9 | 1595.8 | 3784.7 | 239.1 | -5.1 | 35.7 | -120.8 | 1592.0 | 3782.3 | 245.9 | -5.7 | 37.5 | -120.1 | few corners |
| image 177 | r | 1402.8 | 3764.5 | 246.5 | -3.1 | 36.2 | -116.4 | 1405.1 | 3770.3 | 253.7 | -1.0 | 33.4 | -120.8 | 1407.1 | 3768.8 | 250.5 | -3.6 | 36.4 | -117.6 | few corners |
| image 178 | r | 1229.6 | 3749.3 | 246.5 | -3.3 | 38.3 | -111.6 | 1242.5 | 3751.3 | 285.6 | -7.3 | 35.8 | -112.0 | 1235.0 | 3751.6 | 252.5 | -1.9 | 36.4 | -115.4 | few corners |
| image 179 | x | 1083.0 | 3763.7 | 252.5 | 3.9 | 17.1 | -164.5 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 180 | x | 1080.7 | 3826.8 | 253.2 | 10.1 | 18.1 | 92.7 | 1168.3 | 3817.8 | 243.7 | 31.9 | 10.1 | 105.0 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 181 | r | 1181.0 | 3820.0 | 244.7 | 0.6 | 42.6 | 67.8 | 1176.6 | 3826.6 | 244.3 | -3.4 | 42.2 | 61.6 | 1190.8 | 3824.6 | 260.5 | -0.4 | 45.2 | 65.0 | no |
| image 182 | d | 1311.5 | 3822.2 | 247.0 | -3.1 | 37.4 | 66.7 | 1314.8 | 3820.1 | 248.2 | -2.7 | 37.1 | 63.7 | 1313.8 | 3824.7 | 250.9 | -3.4 | 37.7 | 63.0 | no |
| image 183 | d | 1461.8 | 3822.9 | 246.2 | -3.6 | 42.1 | 63.2 | 1464.9 | 3820.3 | 247.3 | -2.8 | 39.4 | 61.2 | 1465.1 | 3828.8 | 251.0 | -3.8 | 40.7 | 60.2 | no |
| image 184 | d | 1633.0 | 3823.1 | 244.2 | -4.4 | 37.7 | 63.5 | 1639.7 | 3824.8 | 248.5 | -6.2 | 40.7 | 61.3 | 1630.9 | 3828.8 | 246.7 | -7.5 | 40.4 | 58.7 | no |
| image 185 | d | 1822.0 | 3824.4 | 237.7 | -4.8 | 36.2 | 64.9 | 1823.8 | 3826.0 | 240.3 | -5.9 | 39.2 | 61.0 | 1818.5 | 3827.3 | 238.1 | -7.1 | 38.7 | 59.7 | no |
| image 186 | d | 2013.3 | 3823.7 | 234.2 | -10.7 | 39.6 | 60.5 | 2014.1 | 3824.3 | 237.4 | -10.3 | 39.0 | 58.2 | 2010.9 | 3822.6 | 239.9 | -11.8 | 39.0 | 57.4 | no |
| image 187 | d | 2176.0 | 3834.1 | 234.7 | -26.2 | 45.8 | 75.1 | 2177.3 | 3835.4 | 239.7 | -29.7 | 47.3 | 74.1 | 2180.7 | 3833.7 | 242.7 | -28.7 | 47.6 | 75.4 | no |
| image 188 | d | 2226.4 | 3903.9 | 237.0 | -26.2 | 47.1 | 143.3 | 2226.3 | 3889.1 | 182.3 | -24.3 | 36.9 | 144.8 | 2230.6 | 3908.4 | 239.0 | -22.0 | 49.0 | 151.6 | random |
| image 189 | d | 2145.9 | 3949.7 | 227.2 | -2.5 | 43.2 | -116.8 | 2146.2 | 3949.0 | 231.7 | -2.2 | 41.7 | -116.1 | 2147.4 | 3951.2 | 233.5 | -2.4 | 41.7 | -116.0 | no |
| image 190 | d | 2021.3 | 3955.1 | 224.0 | -5.3 | 37.0 | -114.7 | 2027.5 | 3956.7 | 229.8 | -5.7 | 37.3 | -115.3 | 2027.5 | 3957.7 | 225.8 | -5.4 | 36.6 | -115.2 | no |
| image 191 | d | 1897.7 | 3957.7 | 221.7 | -5.0 | 35.2 | -111.4 | 1902.6 | 3960.7 | 226.0 | -5.5 | 37.4 | -109.9 | 1905.3 | 3961.2 | 218.1 | -5.8 | 36.0 | -110.5 | no |
| image 192 | d | 1750.1 | 3955.5 | 221.0 | -6.2 | 36.9 | -112.4 | 1714.9 | 3972.2 | 255.7 | 8.1 | 43.9 | -102.8 | 1754.7 | 3958.3 | 224.5 | -6.0 | 36.9 | -111.5 | random |
| image 193 | d | 1582.3 | 3953.2 | 222.7 | -9.0 | 40.9 | -111.2 | 1590.8 | 3963.4 | 204.7 | -9.7 | 36.0 | -111.7 | 1588.4 | 3955.7 | 223.1 | -9.8 | 39.6 | -111.9 | no |
| image 194 | d | 1415.8 | 3949.3 | 227.2 | -7.8 | 38.8 | -114.8 | 1420.6 | 3948.4 | 232.3 | -7.3 | 39.7 | -116.6 | 1419.4 | 3953.3 | 240.1 | -6.9 | 40.4 | -115.6 | no |
| image 195 | r | 1249.6 | 3948.2 | 229.5 | -12.6 | 40.0 | -116.7 | 1258.2 | 3954.6 | 223.5 | -17.3 | 37.6 | -120.3 | 1255.8 | 3947.1 | 226.7 | -16.6 | 39.3 | -120.3 | no |
| image 196 | x | 1103.7 | 3950.3 | 230.0 | -10.8 | 28.8 | -136.7 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 197 | x | 1056.5 | 4001.6 | 234.0 | 10.6 | 24.5 | 120.6 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 198 | d | 1149.0 | 4047.7 | 229.0 | 2.7 | 36.5 | 70.3 | 1142.1 | 4032.4 | 223.5 | -5.8 | 29.7 | 66.2 | 1152.4 | 4046.0 | 232.5 | -0.4 | 36.3 | 67.1 | random |
| image 199 | d | 1290.0 | 4059.9 | 232.7 | -5.6 | 38.4 | 56.0 | 1294.2 | 4056.0 | 226.4 | -5.6 | 36.3 | 55.7 | 1290.8 | 4059.8 | 236.2 | -7.3 | 38.4 | 54.4 | no |
| image 200 | d | 1446.2 | 4061.1 | 238.0 | -7.8 | 36.4 | 52.1 | 1448.3 | 4060.3 | 243.1 | -8.6 | 37.4 | 51.4 | 1447.8 | 4060.4 | 240.6 | -8.8 | 37.1 | 51.3 | no |
| image 201 | d | 1616.1 | 4061.9 | 243.5 | -9.3 | 35.4 | 51.8 | 1621.0 | 4059.2 | 248.4 | -7.9 | 35.9 | 51.0 | 1617.7 | 4059.2 | 249.4 | -8.7 | 35.9 | 50.5 | no |
| image 202 | d | 1795.4 | 4060.4 | 244.2 | -10.4 | 37.1 | 52.7 | 1795.9 | 4061.7 | 251.3 | -10.5 | 38.9 | 50.6 | 1796.6 | 4058.5 | 247.4 | -10.3 | 38.0 | 51.1 | no |
| image 203 | r | 1975.2 | 4058.4 | 246.2 | -9.9 | 37.3 | 53.9 | 1977.3 | 4058.0 | 248.3 | -9.0 | 37.4 | 51.9 | 1975.7 | 4047.2 | 256.2 | -8.2 | 37.6 | 53.2 | no |
| image 204 | r | 2145.7 | 4060.2 | 251.0 | -24.3 | 37.0 | 64.5 | 2143.7 | 4064.7 | 255.9 | -26.3 | 39.4 | 65.3 | 2154.6 | 4052.1 | 255.1 | -23.4 | 38.3 | 69.2 | no |
| image 205 | r | 2221.5 | 4134.4 | 261.0 | -30.4 | 48.3 | 137.0 | 2226.7 | 4140.4 | 267.9 | -28.9 | 46.5 | 142.2 | 2224.6 | 4137.4 | 259.7 | -30.1 | 45.5 | 141.2 | no |
| image 206 | r | 2142.3 | 4182.9 | 253.0 | 0.6 | 41.7 | -116.8 | 2143.4 | 4182.7 | 253.7 | -0.2 | 40.5 | -116.1 | 2145.9 | 4181.1 | 253.5 | -1.2 | 40.6 | -117.0 | no |
| image 207 | d | 1999.0 | 4183.5 | 250.0 | -6.8 | 34.9 | -120.1 | 2005.0 | 4183.5 | 252.5 | -6.2 | 34.3 | -119.8 | 2004.2 | 4181.8 | 249.2 | -6.5 | 34.0 | -119.8 | no |
| image 208 | d | 1837.3 | 4183.4 | 252.7 | -8.2 | 37.1 | -122.6 | 1840.6 | 4186.1 | 256.9 | -7.0 | 37.6 | -121.9 | 1839.7 | 4186.2 | 260.5 | -5.5 | 38.0 | -121.4 | no |
| image 209 | d | 1668.7 | 4181.9 | 253.0 | -10.5 | 37.2 | -120.1 | 1676.4 | 4183.9 | 257.4 | -9.7 | 37.0 | -121.1 | 1672.0 | 4183.3 | 256.7 | -9.1 | 37.2 | -120.2 | no |
| image 210 | r | 1494.2 | 4178.6 | 253.0 | -8.4 | 39.6 | -122.5 | 1487.2 | 4188.3 | 226.0 | -10.2 | 30.7 | -125.8 | 1497.5 | 4179.6 | 255.5 | -9.7 | 39.4 | -125.5 | random |
| image 211 | r | 1309.5 | 4171.5 | 253.2 | -8.8 | 36.3 | -123.6 | 1318.7 | 4171.2 | 261.0 | -11.0 | 35.9 | -126.3 | 1321.3 | 4175.0 | 255.6 | -12.0 | 34.5 | -126.2 | no |
| image 212 | r | 1132.7 | 4165.9 | 256.2 | -12.4 | 35.8 | -127.1 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 213 | x | 1008.5 | 4194.9 | 262.5 | -3.9 | 17.7 | -178.3 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 214 | r | 1023.2 | 4251.2 | 266.2 | 9.3 | 41.2 | 84.4 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 215 | c | 1139.0 | 4273.1 | 265.7 | 1.2 | 39.1 | 55.2 | 1144.7 | 4267.3 | 269.0 | 1.1 | 38.7 | 55.4 | 1140.9 | 4273.8 | 273.0 | -0.2 | 39.8 | 53.6 | no |
| image 216 | c | 1291.4 | 4272.7 | 266.7 | 0.9 | 43.0 | 58.6 | 1295.3 | 4271.9 | 272.8 | 2.7 | 42.8 | 57.5 | 1301.1 | 4264.4 | 267.8 | 4.4 | 41.7 | 60.1 | no |
| image 217 | c | 1464.4 | 4270.1 | 266.0 | 3.7 | 33.9 | 59.6 | 1441.0 | 4258.0 | 237.4 | -6.9 | 27.2 | 58.0 | 1468.8 | 4274.3 | 268.2 | 2.7 | 34.9 | 59.4 | few corners |
| image 218 | c | 1646.9 | 4267.1 | 262.2 | 5.7 | 30.5 | 62.0 | 1645.8 | 4267.9 | 265.9 | 5.3 | 30.8 | 62.4 | 1647.4 | 4271.3 | 267.4 | 5.6 | 31.1 | 62.4 | no |
| image 219 | r | 1837.7 | 4268.6 | 258.7 | -2.0 | 35.7 | 53.4 | 1843.4 | 4266.3 | 264.2 | -0.2 | 34.8 | 53.9 | 1844.2 | 4264.8 | 266.9 | 1.4 | 34.9 | 54.2 | no |
| image 220 | r | 2026.1 | 4266.4 | 256.2 | -6.6 | 35.8 | 54.3 | 2030.4 | 4267.3 | 251.8 | -8.1 | 35.9 | 52.8 | 2025.9 | 4267.7 | 258.4 | -9.0 | 36.6 | 52.1 | no |
| image 221 | r | 2169.3 | 4304.7 | 262.0 | -25.1 | 50.9 | 99.8 | 2214.0 | 4300.7 | 219.0 | -4.8 | 39.8 | 114.5 | 2166.3 | 4305.4 | 262.3 | -22.7 | 46.3 | 107.6 | random |
| image 222 | r | 2177.6 | 4382.6 | 261.0 | -7.9 | 45.8 | -173.0 | 2183.3 | 4370.8 | 254.3 | -10.3 | 46.0 | -171.0 | 2179.6 | 4380.8 | 264.3 | -6.4 | 47.8 | -166.8 | random |
| image 223 | d | 2081.7 | 4411.0 | 252.2 | 6.5 | 37.0 | -106.7 | 2086.1 | 4409.8 | 256.3 | 3.0 | 37.2 | -107.4 | 2083.3 | 4411.6 | 260.1 | 3.9 | 37.5 | -106.5 | no |
| image 224 | d | 1933.1 | 4409.5 | 249.0 | 0.9 | 33.4 | -112.8 | 1937.1 | 4409.1 | 254.1 | 0.0 | 33.6 | -114.9 | 1938.0 | 4407.6 | 250.9 | -0.3 | 33.3 | -115.1 | no |
| image 225 | d | 1761.7 | 4410.4 | 246.0 | -8.9 | 38.9 | -117.7 | 1767.3 | 4411.3 | 251.3 | -9.8 | 39.0 | -119.7 | 1765.7 | 4412.5 | 251.1 | -9.0 | 39.0 | -119.3 | no |
| image 226 | d | 1590.6 | 4409.6 | 247.7 | -8.4 | 39.4 | -116.1 | 1597.3 | 4410.9 | 255.4 | -9.1 | 40.1 | -116.5 | 1597.3 | 4409.6 | 250.7 | -9.4 | 39.7 | -116.5 | no |
| image 227 | d | 1415.8 | 4411.9 | 245.5 | -10.2 | 41.8 | -118.5 | 1421.2 | 4410.9 | 243.1 | -9.5 | 40.7 | -118.7 | 1422.2 | 4413.2 | 241.7 | -9.0 | 40.4 | -118.6 | no |
| image 228 | r | 1246.0 | 4409.9 | 242.2 | -0.2 | 39.8 | -114.7 | 1259.2 | 4406.9 | 250.0 | -2.7 | 41.3 | -119.5 | 1258.2 | 4407.9 | 250.5 | -2.8 | 41.3 | -119.0 | no |

*c: campus
r: residential area
d: downtown
x: erroneous data acquisition

**x: erroneous data acquisition
no: correct pose

|  |  | NAV420CA Reading |  |  |  |  |  | Corner Correspondence |  |  |  |  |  | Manual Correspondence |  |  |  |  |  |  |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
|  | Area* | X | Y | Z | roll | pitch | yaw | X | Y | Z | roll | pitch | yaw | X | Y | Z | roll | pitch | yaw | source of error** |
|  |  | (m) | (m) | (m) | (°) | (°) | (°) | (m) | (m) | (m) | (°) | (°) | (°) | (m) | (m) | (m) | (°) | (°) | (°) |  |
| image 229 | r | 1085.8 | 4397.8 | 239.7 | -4.5 | 38.4 | -120.9 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 230 | x | 973.9 | 4412.4 | 240.5 | 4.9 | 19.9 | -175.9 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 231 | r | 977.9 | 4447.9 | 239.2 | 13.2 | 40.1 | 75.4 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 232 | r | 1052.1 | 4440.9 | 237.5 | 8.2 | 39.3 | 68.8 | 1002.8 | 4545.6 | 303.5 | -27.5 | 58.7 | 27.3 | 1052.4 | 4445.5 | 226.5 | 4.6 | 37.1 | 66.8 | few corners |
| image 233 | c | 1171.6 | 4436.0 | 235.7 | 0.0 | 28.1 | 72.5 | 1166.3 | 4440.8 | 258.1 | -2.7 | 31.7 | 67.9 | 1175.0 | 4444.8 | 239.8 | -0.5 | 28.0 | 71.0 | few corners |
| image 234 | c | 1313.8 | 4438.6 | 235.7 | 9.6 | 35.7 | 67.4 | 1311.9 | 4439.3 | 235.2 | 7.9 | 34.5 | 66.6 | 1315.1 | 4437.7 | 238.5 | 9.1 | 35.2 | 67.6 | no |
| image 235 | c | 1474.4 | 4443.8 | 237.0 | 1.9 | 33.4 | 61.3 | 1473.3 | 4443.6 | 241.2 | 1.3 | 33.7 | 59.2 | 1473.5 | 4450.2 | 239.7 | 0.9 | 33.8 | 58.7 | no |
| image 236 | c | 1642.2 | 4447.1 | 236.7 | 0.1 | 31.5 | 58.2 | 1646.1 | 4444.8 | 242.3 | 0.1 | 32.5 | 60.2 | 1643.9 | 4447.1 | 239.8 | 0.0 | 32.1 | 59.7 | no |
| image 237 | c | 1813.9 | 4447.7 | 238.0 | -2.8 | 32.1 | 61.1 | 1812.5 | 4448.2 | 241.6 | -3.7 | 31.6 | 59.0 | 1816.0 | 4447.5 | 242.6 | -3.3 | 32.0 | 60.0 | no |
| image 238 | r | 1981.5 | 4458.2 | 241.5 | -1.9 | 34.4 | 64.8 | 1962.8 | 4505.1 | 198.2 | -5.8 | 29.0 | 63.5 | 1983.3 | 4455.4 | 245.9 | -2.6 | 36.4 | 65.1 | no |
| image 239 | x | 2136.4 | 4467.0 | 246.2 | 0.5 | 20.8 | 53.9 | 2076.0 | 4481.1 | 222.9 | -2.3 | 11.5 | 50.3 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 240 | x | 2190.5 | 4397.2 | 252.5 | 9.8 | 25.6 | -19.4 | 2191.2 | 4394.7 | 251.9 | 10.5 | 30.9 | -23.4 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 241 | r | 2169.0 | 4284.1 | 245.2 | 1.2 | 34.7 | -19.4 | 2227.0 | 4357.4 | 189.6 | -13.0 | 23.4 | -43.4 | 2181.8 | 4277.5 | 268.6 | 0.1 | 36.8 | -21.1 | few corners |
| image 242 | x | 2167.1 | 4203.2 | 245.7 | -12.8 | 16.7 | -24.8 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 243 | x | 2139.4 | 4214.2 | 245.0 | -2.5 | 23.0 | -27.4 | 2152.5 | 4210.3 | 266.6 | -6.2 | 27.0 | -32.8 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 244 | x | 2089.3 | 4284.8 | 243.0 | 2.0 | 28.6 | -47.0 | 2110.1 | 4295.4 | 237.0 | -1.2 | 26.8 | -51.5 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 245 | x | 2047.2 | 4389.3 | 243.2 | 9.3 | 30.4 | -34.4 | 2057.0 | 4384.2 | 243.7 | 8.8 | 30.1 | -32.5 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 246 | r | 2003.7 | 4497.7 | 249.2 | 9.7 | 35.8 | -49.8 | 2009.3 | 4489.8 | 253.7 | 11.7 | 38.0 | -46.6 | 2008.7 | 4490.1 | 249.4 | 11.9 | 37.2 | -46.7 | no |
| image 247 | r | 1971.5 | 4560.2 | 255.7 | 24.0 | 38.8 | -20.5 | 1962.4 | 4532.7 | 271.8 | 34.0 | 34.4 | -11.0 | 1978.8 | 4559.1 | 254.0 | 26.0 | 38.3 | -17.7 | random |
| image 248 | r | 1950.5 | 4528.4 | 252.5 | 8.8 | 36.5 | -25.7 | 1930.5 | 4571.1 | 221.8 | 14.9 | 25.0 | -35.0 | N/A | N/A | N/A | N/A | N/A | N/A | random |
| image 249 | r | 1953.4 | 4432.2 | 248.0 | 7.3 | 33.7 | -23.0 | 1942.1 | 4415.4 | 262.9 | 10.5 | 34.3 | -15.8 | 1955.5 | 4431.5 | 250.1 | 9.2 | 32.9 | -21.1 | random |
| image 250 | x | 1954.0 | 4301.5 | 249.5 | 6.7 | 18.5 | -44.0 | 1936.6 | 4320.2 | 324.5 | 0.1 | 17.6 | -48.2 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 251 | d | 1858.2 | 4255.0 | 250.2 | 7.6 | 19.6 | -96.2 | 1859.3 | 4260.2 | 257.2 | 5.8 | 24.0 | -96.4 | 1862.3 | 4252.5 | 250.5 | 5.2 | 23.8 | -97.1 | no |
| image 252 | d | 1725.2 | 4289.8 | 243.7 | -0.3 | 41.6 | -112.9 | 1732.2 | 4278.4 | 244.5 | -5.4 | 43.4 | -109.3 | 1730.5 | 4291.3 | 240.4 | -0.1 | 41.9 | -107.0 | random |
| image 253 | d | 1597.4 | 4297.4 | 242.7 | -2.0 | 42.8 | -109.2 | 1601.0 | 4298.4 | 244.2 | -1.7 | 42.3 | -109.5 | 1599.3 | 4298.3 | 244.4 | -1.3 | 42.6 | -109.0 | no |
| image 254 | d | 1438.5 | 4300.6 | 242.2 | -1.9 | 43.1 | -112.9 | 1437.9 | 4302.9 | 244.3 | 2.7 | 42.0 | -109.4 | 1445.4 | 4302.4 | 240.8 | -0.1 | 41.0 | -111.5 | no |
| image 255 | r | 1265.9 | 4304.1 | 243.2 | -1.6 | 42.1 | -112.2 | 1268.2 | 4308.4 | 243.5 | -0.8 | 42.3 | -109.8 | 1276.2 | 4304.5 | 248.6 | -3.0 | 43.3 | -112.3 | no |
| image 256 | r | 1103.1 | 4304.5 | 243.2 | -3.8 | 41.6 | -102.1 | 1124.8 | 4323.9 | 278.8 | -13.9 | 36.2 | -102.0 | 1095.3 | 4296.1 | 253.6 | 0.9 | 47.0 | -95.6 | few corners |
| image 257 | r | 995.8 | 4254.1 | 246.5 | -6.0 | 48.7 | -39.0 | 994.7 | 4326.9 | 264.6 | -38.2 | 41.8 | -53.0 | 1003.4 | 4253.6 | 243.6 | -7.5 | 49.9 | -32.5 | few corners |
| image 258 | r | 1039.4 | 4195.3 | 245.5 | 4.0 | 39.3 | 56.4 | 1043.8 | 4197.0 | 248.9 | 10.7 | 37.0 | 61.9 | 1047.0 | 4200.0 | 238.6 | 7.3 | 36.2 | 63.1 | random |
| image 259 | c | 1157.7 | 4188.5 | 240.0 | 5.2 | 27.8 | 63.3 | 1162.4 | 4189.2 | 240.7 | 6.4 | 27.2 | 66.1 | 1171.8 | 4176.3 | 243.5 | 8.0 | 26.9 | 68.7 | no |
| image 260 | x | 1266.4 | 4181.2 | 235.2 | 8.4 | 7.8 | 73.1 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 261 | x | 1401.2 | 4172.7 | 228.5 | 8.0 | 5.3 | 71.4 | 1327.6 | 4238.1 | 120.1 | 12.1 | -4.5 | 60.7 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 262 | x | 1545.5 | 4163.5 | 228.2 | 9.2 | 12.3 | 65.0 | 4897.5 | 2986.6 | 2219.9 | 160.9 | 42.4 | 146.5 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 263 | x | 1697.1 | 4167.5 | 234.2 | 11.3 | 10.4 | 63.9 | 1692.3 | 4169.0 | 246.6 | 12.7 | 9.8 | 66.6 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 264 | x | 1852.5 | 4169.5 | 240.0 | 5.2 | 6.7 | 58.6 | 1943.3 | 4116.2 | 212.5 | 10.4 | 6.5 | 70.5 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 265 | r | 1982.5 | 4168.0 | 246.2 | -0.7 | 30.5 | 59.6 | 1987.8 | 4169.4 | 250.5 | -1.7 | 30.1 | 61.2 | 1984.7 | 4169.6 | 249.9 | -2.0 | 29.9 | 60.7 | no |
| image 266 | r | 2083.1 | 4164.1 | 251.2 | 0.1 | 29.2 | 78.5 | 2055.4 | 4164.8 | 232.4 | -6.4 | 25.5 | 74.9 | 2086.3 | 4166.0 | 256.2 | -0.5 | 29.8 | 84.8 | random |
| image 267 | r | 2170.3 | 4165.5 | 254.7 | 2.1 | 28.3 | 75.8 | 2144.8 | 4170.3 | 193.8 | -4.3 | 14.7 | 74.7 | 2169.7 | 4172.3 | 252.6 | 1.2 | 29.7 | 75.2 | random |
| image 268 | r | 2255.4 | 4168.9 | 255.5 | 1.7 | 37.1 | 116.4 | 2281.4 | 4173.0 | 250.4 | 4.3 | 33.7 | 121.5 | 2256.5 | 4171.4 | 258.3 | 2.1 | 39.1 | 124.2 | random |
| image 269 | r | 2287.5 | 4222.1 | 254.5 | 6.9 | 37.6 | 161.9 | 2290.4 | 4225.7 | 251.5 | 8.1 | 36.9 | 164.9 | 2289.5 | 4228.3 | 257.9 | 8.4 | 37.9 | 165.4 | no |
| image 270 | r | 2283.3 | 4315.9 | 254.2 | 2.2 | 32.5 | 161.9 | 2285.4 | 4316.9 | 256.4 | 2.1 | 32.1 | 161.8 | 2286.1 | 4316.2 | 256.2 | 2.2 | 31.9 | 161.7 | no |
| image 271 | r | 2281.1 | 4434.1 | 257.2 | 1.6 | 33.6 | 163.2 | 2285.6 | 4439.6 | 260.9 | 2.7 | 33.7 | 161.9 | 2280.7 | 4436.2 | 264.4 | 1.5 | 34.3 | 160.8 | no |
| image 272 | r | 2281.6 | 4571.4 | 261.0 | -7.7 | 33.8 | 154.6 | 2285.7 | 4574.6 | 252.7 | -5.0 | 35.3 | 154.8 | 2286.4 | 4570.2 | 260.8 | -6.8 | 36.3 | 153.6 | no |
| image 273 | r | 2275.1 | 4722.3 | 267.0 | -3.8 | 34.8 | 161.3 | 2279.1 | 4708.4 | 269.7 | -4.6 | 35.7 | 155.1 | 2278.8 | 4718.7 | 270.1 | -3.0 | 36.2 | 157.1 | no |
| image 274 | r | 2264.3 | 4886.9 | 274.2 | -5.7 | 33.3 | 154.4 | 2268.8 | 4888.1 | 278.9 | -8.8 | 34.7 | 152.2 | 2265.5 | 4891.0 | 280.6 | -8.3 | 35.1 | 152.4 | no |
| image 275 | r | 2263.2 | 5057.5 | 279.7 | -8.6 | 35.3 | 156.9 | 2264.4 | 5051.5 | 280.9 | -8.7 | 35.2 | 152.2 | 2263.7 | 5056.2 | 286.0 | -7.7 | 36.3 | 153.1 | no |
| image 276 | r | 2259.8 | 5233.8 | 289.0 | -7.4 | 32.6 | 152.8 | 2256.9 | 5233.2 | 289.8 | -9.9 | 32.0 | 151.5 | 2266.9 | 5241.5 | 293.5 | -8.1 | 32.1 | 153.1 | random |
| image 277 | r | 2258.0 | 5405.8 | 294.2 | -11.6 | 33.8 | 156.9 | 2260.0 | 5399.7 | 295.2 | -13.7 | 33.7 | 152.6 | 2255.3 | 5395.9 | 299.2 | -13.7 | 34.5 | 151.9 | no |
| image 278 | r | 2254.0 | 5568.4 | 299.2 | -15.8 | 35.0 | 161.3 | 2255.9 | 5564.0 | 303.6 | -17.9 | 36.6 | 158.3 | 2253.1 | 5561.7 | 295.1 | -17.8 | 35.7 | 158.1 | no |
| image 279 | r | 2213.3 | 5668.1 | 305.2 | -24.8 | 39.8 | -141.7 | 2213.9 | 5637.6 | 263.8 | -27.6 | 33.3 | -148.0 | 2205.6 | 5660.3 | 303.8 | -20.8 | 39.9 | -136.4 | random |
| image 280 | r | 2153.5 | 5630.5 | 297.5 | 5.0 | 31.5 | -46.8 | 2159.1 | 5626.9 | 298.5 | 6.9 | 31.8 | -46.3 | 2150.4 | 5621.7 | 293.6 | 8.2 | 30.9 | -44.9 | no |
| image 281 | r | 2114.6 | 5510.1 | 289.5 | -4.0 | 37.7 | -33.1 | 2118.9 | 5510.7 | 292.1 | -5.2 | 38.2 | -32.1 | 2116.7 | 5506.3 | 289.3 | -4.7 | 37.7 | -31.3 | no |
| image 282 | r | 2116.4 | 5353.2 | 280.7 | -7.4 | 35.4 | -32.4 | 2123.1 | 5355.9 | 284.1 | -8.3 | 35.3 | -35.7 | 2123.2 | 5357.7 | 279.4 | -8.2 | 34.2 | -35.9 | no |
| image 283 | r | 2118.7 | 5177.6 | 270.7 | -6.9 | 34.7 | -34.8 | 2122.1 | 5180.0 | 273.2 | -7.7 | 35.2 | -35.3 | 2122.1 | 5179.1 | 264.4 | -5.9 | 34.1 | -34.7 | no |
| image 284 | r | 2118.9 | 4996.4 | 264.0 | -8.6 | 35.5 | -35.0 | 2095.8 | 5005.0 | 265.3 | -13.5 | 35.1 | -38.1 | 2120.8 | 4996.2 | 263.8 | -8.5 | 35.1 | -36.2 | few corners |
| image 285 | r | 2125.3 | 4818.5 | 259.2 | -8.2 | 34.9 | -34.0 | 2138.2 | 4832.6 | 293.7 | -5.6 | 38.3 | -36.3 | 2125.6 | 4815.9 | 255.2 | -4.9 | 33.1 | -32.7 | no |
| image 286 | r | 2133.8 | 4638.6 | 254.7 | -6.9 | 36.5 | -35.9 | 2128.4 | 4624.5 | 260.8 | -0.5 | 36.6 | -28.8 | 2126.1 | 4635.1 | 255.4 | -1.4 | 34.8 | -31.0 | no |
| image 287 | r | 2141.1 | 4453.8 | 243.5 | -4.3 | 34.7 | -32.9 | 2152.5 | 4461.1 | 248.9 | -7.9 | 37.2 | -33.4 | 2139.8 | 4454.9 | 238.4 | -5.8 | 34.6 | -30.5 | no |
| image 288 | r | 2147.8 | 4267.6 | 236.5 | -5.5 | 36.4 | -30.7 | 2164.5 | 4275.0 | 247.7 | -9.0 | 39.3 | -32.4 | 2155.2 | 4275.8 | 231.9 | -7.1 | 35.9 | -31.3 | no |
| image 289 | r | 2160.6 | 4083.7 | 228.2 | -10.5 | 33.8 | -32.7 | 2155.3 | 4083.8 | 229.3 | -8.2 | 32.8 | -29.0 | 2162.9 | 4086.7 | 231.6 | -9.6 | 33.8 | -30.6 | no |
| image 290 | r | 2174.7 | 3915.8 | 224.2 | -10.6 | 32.7 | -33.6 | 2177.8 | 3918.8 | 222.9 | -12.0 | 32.6 | -32.7 | 2181.5 | 3922.8 | 225.4 | -12.4 | 33.0 | -33.8 | no |
| image 291 | x | 2182.6 | 3781.6 | 223.2 | -6.9 | 13.9 | -56.0 | 2145.7 | 3856.9 | 207.2 | -3.7 | 6.4 | -54.8 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 292 | d | 2140.0 | 3760.8 | 226.0 | 6.7 | 26.2 | -154.9 | 2113.6 | 3773.4 | 247.8 | 14.6 | 30.7 | -147.7 | 2137.6 | 3767.2 | 223.9 | 8.6 | 30.5 | -159.4 | random |
| image 293 | d | 2109.9 | 3850.6 | 227.2 | -3.7 | 35.4 | 159.8 | 2111.6 | 3853.4 | 229.2 | -3.7 | 34.7 | 158.0 | 2108.2 | 3853.3 | 227.5 | -4.2 | 34.7 | 157.8 | no |
| image 294 | d | 2099.4 | 3976.7 | 228.7 | -7.0 | 36.7 | 148.6 | 2105.6 | 3978.7 | 232.6 | -7.0 | 36.6 | 151.2 | 2102.9 | 3973.9 | 234.4 | -8.1 | 36.9 | 150.0 | no |
| image 295 | d | 2094.9 | 4114.8 | 235.0 | -8.9 | 33.3 | 148.6 | 2101.6 | 4114.5 | 236.4 | -8.6 | 32.6 | 149.8 | 2099.6 | 4113.6 | 241.0 | -9.5 | 33.7 | 149.6 | no |
| image 296 | c | 2094.2 | 4259.3 | 242.2 | -9.1 | 34.8 | 152.0 | 2096.9 | 4256.5 | 246.9 | -10.3 | 35.0 | 152.6 | 2100.8 | 4256.2 | 240.8 | -9.8 | 33.7 | 152.7 | no |
| image 297 | c | 2088.6 | 4410.1 | 249.7 | -8.1 | 33.8 | 156.9 | 2099.9 | 4418.3 | 246.6 | -5.6 | 32.2 | 158.9 | 2096.4 | 4410.1 | 249.0 | -7.9 | 32.9 | 156.9 | no |
| image 298 | c | 2076.8 | 4569.1 | 260.5 | -7.7 | 32.5 | 152.0 | 2080.1 | 4568.1 | 262.1 | -8.3 | 33.1 | 150.4 | 2079.8 | 4568.8 | 265.5 | -7.9 | 33.6 | 150.6 | no |
| image 299 | c | 2061.9 | 4732.7 | 262.7 | -8.2 | 31.8 | 150.9 | 2072.4 | 4733.3 | 272.2 | -6.8 | 32.6 | 151.8 | 2059.3 | 4725.4 | 263.2 | -8.4 | 31.9 | 149.1 | no |
| image 300 | c | 2054.6 | 4901.0 | 261.7 | -7.9 | 32.0 | 149.9 | 2061.3 | 4903.5 | 263.9 | -6.4 | 32.2 | 149.5 | 2067.4 | 4900.2 | 250.1 | -5.3 | 29.5 | 149.7 | no |
| image 301 | c | 2043.6 | 5069.7 | 263.5 | -9.1 | 32.7 | 152.1 | 2048.4 | 5072.4 | 269.3 | -8.6 | 33.6 | 152.8 | 2047.6 | 5066.5 | 264.8 | -9.6 | 32.9 | 151.6 | no |
| image 302 | c | 2028.9 | 5240.4 | 270.2 | -7.5 | 31.2 | 152.1 | 2036.7 | 5239.1 | 275.4 | -6.0 | 31.0 | 150.5 | 2037.9 | 5242.1 | 274.8 | -6.3 | 31.0 | 151.2 | no |
| image 303 | c | 2023.7 | 5411.0 | 278.2 | -13.9 | 28.2 | 149.7 | 2030.5 | 5386.3 | 272.6 | -17.1 | 25.6 | 144.6 | 2031.1 | 5411.3 | 279.7 | -13.4 | 27.9 | 149.6 | few corners |
| image 304 | c | 2013.4 | 5544.4 | 282.2 | -23.9 | 38.8 | 178.2 | 2000.6 | 5543.6 | 280.8 | -30.5 | 33.7 | 175.6 | 2018.8 | 5537.8 | 286.6 | -26.8 | 39.8 | 179.5 | few corners |

*c: campus
r: residential area
d: downtown
x: erroneous data acquisition

**x: erroneous data acquisition
no: correct pose

| | | NAV420CA Reading | | | | | | Corner Correspondence | | | | | | Manual Correspondence | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Area* | X | Y | Z | roll | pitch | yaw | X | Y | Z | roll | pitch | yaw | X | Y | Z | roll | pitch | yaw | source of error** |
| | | (m) | (m) | (m) | (˚) | (˚) | (˚) | (m) | (m) | (m) | (˚) | (˚) | (˚) | (m) | (m) | (m) | (˚) | (˚) | (˚) | |
| image 305 | r | 1963.8 | 5574.5 | 280.0 | -6.0 | 42.9 | -91.8 | 2083.8 | 5548.0 | 325.2 | -28.4 | 41.9 | -114.6 | 1968.2 | 5569.2 | 273.1 | -3.4 | 39.3 | -85.5 | random |
| image 306 | r | 1923.2 | 5479.9 | 272.2 | -0.7 | 35.5 | -33.1 | 1928.1 | 5480.3 | 271.2 | -4.2 | 37.4 | -32.8 | 1931.8 | 5494.8 | 277.2 | -6.3 | 38.1 | -35.7 | no |
| image 307 | r | 1917.9 | 5342.9 | 262.5 | -7.7 | 35.2 | -35.1 | 1923.6 | 5343.4 | 265.3 | -7.5 | 35.2 | -37.5 | 1919.6 | 5355.6 | 277.9 | -10.8 | 35.6 | -40.3 | no |
| image 308 | r | 1922.9 | 5189.5 | 258.2 | -7.0 | 34.3 | -34.7 | 1924.8 | 5187.0 | 260.4 | -5.7 | 34.8 | -34.9 | 1919.7 | 5192.2 | 257.9 | -6.4 | 32.8 | -35.3 | no |
| image 309 | r | 1923.1 | 5027.4 | 253.7 | -6.6 | 38.7 | -34.7 | 1926.3 | 5028.8 | 255.1 | -7.0 | 37.8 | -34.8 | 1929.0 | 5024.0 | 261.2 | -6.2 | 39.2 | -34.3 | no |
| image 310 | r | 1929.9 | 4869.1 | 251.7 | -4.3 | 37.1 | -28.5 | 1931.7 | 4867.7 | 255.1 | -2.3 | 36.9 | -28.3 | 1933.8 | 4871.0 | 255.9 | -3.2 | 37.1 | -29.3 | no |
| image 311 | r | 1935.8 | 4713.6 | 248.2 | -5.8 | 36.3 | -33.6 | 1941.9 | 4713.4 | 249.9 | -5.9 | 37.1 | -33.6 | 1937.5 | 4714.5 | 253.0 | -6.0 | 37.0 | -33.2 | no |
| image 312 | r | 1940.2 | 4560.9 | 247.2 | -6.3 | 35.6 | -32.3 | 1942.8 | 4556.2 | 243.3 | -4.5 | 34.6 | -30.5 | 1942.5 | 4559.0 | 251.8 | -5.6 | 35.8 | -30.7 | no |
| image 313 | r | 1943.5 | 4410.1 | 245.0 | -7.0 | 35.9 | -34.0 | 1944.8 | 4413.6 | 262.9 | -7.8 | 38.5 | -30.9 | 1946.8 | 4408.0 | 240.0 | -6.0 | 35.6 | -29.4 | no |
| image 314 | r | 1957.7 | 4262.3 | 243.7 | -5.6 | 35.4 | -32.1 | 1959.8 | 4261.2 | 246.4 | -5.6 | 36.4 | -29.2 | 1954.9 | 4253.3 | 234.0 | -2.5 | 35.5 | -27.3 | no |
| image 315 | r | 1965.4 | 4107.6 | 242.2 | -4.8 | 38.1 | -30.9 | 1967.8 | 4109.2 | 245.7 | -4.7 | 38.7 | -27.9 | 1970.2 | 4106.5 | 252.1 | -3.8 | 39.9 | -27.6 | no |
| image 316 | r | 1971.3 | 3955.8 | 242.5 | -9.7 | 37.9 | -33.2 | 1974.0 | 3956.5 | 245.3 | -9.5 | 38.3 | -29.6 | 1973.7 | 3957.2 | 246.3 | -9.9 | 38.3 | -29.7 | no |
| image 317 | x | 1968.8 | 3828.0 | 245.7 | -6.5 | 19.2 | -63.2 | 1854.2 | 3875.9 | 291.5 | 0.9 | 14.2 | -58.3 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 318 | d | 1925.2 | 3827.3 | 248.2 | -0.7 | 31.0 | -172.7 | 1967.5 | 3871.7 | 336.0 | 8.7 | 36.9 | -169.1 | 1926.2 | 3832.2 | 250.9 | -3.6 | 34.0 | -177.4 | random |
| image 319 | d | 1910.2 | 3922.8 | 246.7 | -4.8 | 41.0 | 159.6 | 1911.6 | 3923.5 | 248.1 | -4.5 | 37.8 | 161.6 | 1915.4 | 3923.9 | 253.9 | -4.0 | 38.4 | 162.0 | no |
| image 320 | d | 1898.7 | 4034.1 | 247.2 | -2.6 | 37.2 | 157.7 | 1901.0 | 4038.0 | 249.7 | -2.5 | 37.6 | 158.9 | 1900.9 | 4040.0 | 249.5 | -1.8 | 37.8 | 159.6 | no |
| image 321 | d | 1888.7 | 4174.6 | 252.0 | -6.7 | 36.8 | 155.3 | 1891.8 | 4171.6 | 259.8 | -8.3 | 36.6 | 154.6 | 1893.2 | 4173.3 | 262.1 | -7.2 | 37.0 | 155.3 | no |
| image 322 | c | 1881.2 | 4334.3 | 260.0 | -10.3 | 35.0 | 152.7 | 1916.6 | 4324.8 | 322.6 | -8.3 | 37.2 | 153.7 | 1883.4 | 4334.0 | 264.5 | -9.4 | 35.2 | 152.5 | few corners |
| image 323 | c | 1872.2 | 4514.4 | 265.0 | -9.5 | 33.1 | 153.6 | 1877.8 | 4516.2 | 268.3 | -10.0 | 34.5 | 154.6 | 1877.8 | 4513.1 | 268.8 | -10.1 | 34.5 | 154.1 | no |
| image 324 | c | 1862.6 | 4700.7 | 272.0 | -8.7 | 35.1 | 155.5 | 1908.3 | 4702.5 | 266.4 | -4.5 | 26.2 | 163.2 | 1863.5 | 4689.7 | 274.1 | -10.6 | 34.1 | 153.7 | random |
| image 325 | c | 1849.4 | 4885.2 | 278.7 | -7.3 | 32.8 | 158.7 | 1851.2 | 4886.3 | 276.3 | -6.1 | 31.4 | 158.9 | 1854.8 | 4881.1 | 286.7 | -6.8 | 32.4 | 158.2 | no |
| image 326 | c | 1839.1 | 5073.1 | 285.2 | -10.2 | 34.0 | 158.6 | 1871.6 | 5105.1 | 291.4 | -3.5 | 32.7 | 159.6 | 1835.4 | 5062.6 | 279.6 | -12.1 | 32.7 | 154.8 | random |
| image 327 | c | 1824.2 | 5257.1 | 288.5 | -13.3 | 31.7 | 155.8 | 1842.0 | 5291.3 | 250.7 | -10.0 | 22.7 | 165.1 | 1829.8 | 5251.5 | 295.0 | -14.6 | 31.1 | 152.9 | few corners |
| image 328 | c | 1820.3 | 5413.8 | 297.2 | -20.8 | 31.2 | 158.7 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 329 | c | 1785.6 | 5480.7 | 304.2 | -25.7 | 47.2 | -137.9 | 1803.0 | 5523.1 | 270.1 | -32.0 | 37.9 | -132.0 | 1786.9 | 5480.2 | 305.0 | -19.9 | 48.0 | -123.9 | few corners |
| image 330 | r | 1738.7 | 5423.7 | 300.5 | 3.7 | 33.1 | -30.0 | 1719.8 | 5412.3 | 283.8 | 0.2 | 26.5 | -31.0 | 1748.4 | 5423.1 | 299.7 | 3.1 | 32.9 | -28.7 | random |
| image 331 | r | 1724.0 | 5318.1 | 293.7 | -2.0 | 35.5 | -24.9 | 1723.8 | 5308.8 | 285.2 | -2.5 | 34.6 | -24.9 | 1727.3 | 5319.5 | 292.6 | -3.7 | 35.6 | -27.3 | no |
| image 332 | r | 1725.2 | 5175.0 | 287.2 | -4.9 | 32.9 | -23.7 | 1729.3 | 5177.7 | 291.5 | -5.5 | 32.5 | -23.9 | 1729.0 | 5175.4 | 285.2 | -5.6 | 31.7 | -23.5 | no |
| image 333 | r | 1727.6 | 5003.6 | 281.5 | -4.6 | 37.4 | -23.5 | 1729.8 | 5004.3 | 284.2 | -4.6 | 37.3 | -22.1 | 1730.7 | 5004.3 | 286.7 | -4.2 | 37.7 | -22.1 | no |
| image 334 | r | 1735.2 | 4818.4 | 273.7 | -7.5 | 37.0 | -25.9 | 1720.5 | 4823.8 | 272.6 | -9.6 | 32.0 | -27.1 | 1739.5 | 4824.9 | 277.0 | -8.7 | 37.8 | -26.0 | random |
| image 335 | r | 1747.0 | 4631.4 | 268.7 | -8.1 | 34.3 | -27.1 | 1747.1 | 4635.0 | 270.6 | -7.3 | 34.3 | -26.0 | 1752.4 | 4634.9 | 266.9 | -6.8 | 34.2 | -26.3 | no |
| image 336 | r | 1756.2 | 4451.8 | 263.0 | -8.0 | 37.7 | -30.5 | 1759.3 | 4457.7 | 254.3 | -8.1 | 35.7 | -29.8 | 1760.0 | 4453.2 | 265.3 | -8.6 | 37.6 | -29.1 | no |
| image 337 | d | 1765.5 | 4270.9 | 255.2 | -9.2 | 39.1 | -32.7 | 1766.5 | 4272.2 | 259.6 | -7.3 | 39.4 | -28.5 | 1766.1 | 4271.8 | 264.1 | -6.2 | 40.0 | -28.3 | no |
| image 338 | d | 1780.6 | 4097.1 | 248.5 | -11.5 | 39.6 | -34.9 | 1782.3 | 4096.7 | 252.9 | -9.9 | 39.5 | -30.4 | 1783.2 | 4096.0 | 254.2 | -10.2 | 39.7 | -30.4 | no |
| image 339 | d | 1796.9 | 3939.2 | 244.7 | -11.2 | 38.4 | -32.1 | 1799.3 | 3940.9 | 252.1 | -12.7 | 39.4 | -29.0 | 1800.9 | 3941.0 | 250.3 | -13.0 | 39.2 | -29.2 | no |
| image 340 | x | 1805.9 | 3804.5 | 245.7 | -5.5 | 26.0 | -47.9 | 1807.4 | 3788.6 | 206.5 | -17.5 | 14.7 | -50.7 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 341 | x | 1760.6 | 3756.3 | 251.5 | 7.8 | 18.9 | -134.2 | 1825.0 | 3718.4 | 228.0 | 2.8 | 20.0 | -148.3 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 342 | d | 1723.6 | 3844.6 | 252.0 | -2.5 | 37.6 | 164.6 | 1727.0 | 3845.1 | 253.5 | -4.0 | 37.5 | 165.0 | 1726.7 | 3841.0 | 253.4 | -5.3 | 37.3 | 164.1 | no |
| image 343 | d | 1713.3 | 3969.1 | 251.5 | -6.7 | 42.5 | 155.7 | 1715.6 | 3970.3 | 253.5 | -8.5 | 42.2 | 155.2 | 1718.3 | 3969.6 | 258.7 | -8.3 | 42.7 | 155.5 | no |
| image 344 | d | 1698.8 | 4122.4 | 249.7 | -11.0 | 39.7 | 153.2 | 1702.6 | 4120.7 | 254.3 | -11.9 | 37.1 | 153.4 | 1709.5 | 4127.3 | 249.4 | -9.2 | 36.2 | 155.5 | no |
| image 345 | c | 1684.6 | 4288.1 | 256.2 | -11.4 | 38.0 | 150.9 | 1692.8 | 4281.4 | 264.3 | -13.5 | 38.5 | 149.9 | 1685.8 | 4292.1 | 260.5 | -12.7 | 38.8 | 151.3 | no |
| image 346 | c | 1671.7 | 4460.2 | 265.0 | 1.5 | 35.9 | 147.4 | 1668.4 | 4445.0 | 290.4 | 0.7 | 39.1 | 144.3 | 1669.6 | 4459.2 | 268.5 | 0.9 | 36.7 | 146.6 | few corners |
| image 347 | c | 1658.6 | 4632.6 | 270.0 | 1.0 | 35.1 | 148.2 | 1661.3 | 4634.0 | 265.8 | 1.4 | 35.1 | 148.9 | 1664.3 | 4629.2 | 273.0 | 1.9 | 35.6 | 148.8 | no |
| image 348 | c | 1649.7 | 4806.0 | 273.0 | -0.1 | 36.0 | 151.3 | 1642.7 | 4808.3 | 282.6 | 0.7 | 38.5 | 149.4 | 1653.7 | 4805.3 | 275.0 | 0.9 | 36.7 | 149.8 | random |
| image 349 | c | 1641.0 | 4986.3 | 276.7 | -0.5 | 36.1 | 151.6 | 1649.9 | 4992.8 | 275.8 | 3.3 | 35.5 | 153.6 | 1643.7 | 4982.6 | 279.3 | -1.5 | 36.5 | 150.6 | random |
| image 350 | c | 1625.0 | 5168.0 | 279.2 | -3.1 | 33.1 | 155.8 | 1626.1 | 5207.6 | 307.0 | 7.1 | 36.6 | 163.2 | 1633.7 | 5165.4 | 275.9 | -3.5 | 31.6 | 155.4 | few corners |
| image 351 | c | 1606.4 | 5334.9 | 282.5 | -15.6 | 39.0 | 164.3 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 352 | c | 1539.3 | 5411.4 | 287.7 | -25.3 | 47.9 | -135.3 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 353 | c | 1487.5 | 5346.9 | 284.7 | 2.0 | 36.8 | -29.8 | 1493.3 | 5353.9 | 282.2 | -1.7 | 36.4 | -31.2 | 1496.5 | 5346.2 | 286.0 | 0.2 | 37.8 | -29.6 | random |
| image 354 | c | 1489.7 | 5223.4 | 277.7 | 0.4 | 34.1 | -18.5 | 1497.1 | 5226.5 | 275.2 | 0.0 | 32.3 | -19.6 | 1493.3 | 5215.8 | 277.5 | 0.6 | 32.8 | -18.2 | no |
| image 355 | c | 1494.8 | 5106.9 | 271.7 | -3.5 | 35.0 | -19.1 | 1500.7 | 5109.5 | 278.0 | -1.3 | 36.4 | -18.4 | 1498.1 | 5107.9 | 274.2 | -1.1 | 35.7 | -17.9 | random |
| image 356 | c | 1499.3 | 4996.9 | 270.2 | -4.4 | 34.7 | -24.3 | 1480.5 | 4997.4 | 318.1 | -12.0 | 40.6 | -23.3 | 1504.9 | 5000.2 | 274.6 | -5.1 | 35.7 | -24.1 | random |
| image 357 | c | 1508.1 | 4856.4 | 270.0 | -4.4 | 34.9 | -25.1 | 1521.6 | 4853.0 | 242.2 | -9.3 | 30.5 | -23.8 | 1512.6 | 4855.6 | 269.2 | -3.6 | 34.0 | -24.8 | random |
| image 358 | c | 1519.3 | 4703.9 | 268.0 | 0.3 | 35.4 | -20.6 | 1515.5 | 4696.2 | 273.7 | 1.8 | 35.2 | -18.3 | 1523.1 | 4703.0 | 269.6 | 1.3 | 34.9 | -19.6 | no |
| image 359 | c | 1526.7 | 4543.1 | 265.5 | -1.8 | 36.2 | -25.4 | 1528.1 | 4546.2 | 268.3 | -2.6 | 37.8 | -24.8 | 1528.7 | 4542.2 | 270.2 | -2.2 | 38.3 | -23.9 | no |
| image 360 | d | 1535.6 | 4373.6 | 263.5 | -1.9 | 35.3 | -27.1 | 1535.9 | 4373.2 | 264.8 | -0.5 | 35.7 | -22.5 | 1540.3 | 4371.6 | 266.6 | 0.1 | 36.1 | -22.5 | no |
| image 361 | d | 1547.7 | 4201.4 | 257.2 | -2.0 | 38.7 | -24.8 | 1551.7 | 4206.0 | 259.0 | -3.3 | 39.0 | -22.0 | 1549.2 | 4206.2 | 259.4 | -3.4 | 38.9 | -22.0 | no |
| image 362 | d | 1559.0 | 4025.2 | 251.2 | -5.8 | 39.0 | -28.4 | 1559.2 | 4023.8 | 252.1 | -4.5 | 38.8 | -25.6 | 1558.2 | 4028.1 | 257.1 | -5.3 | 39.1 | -26.5 | no |
| image 363 | d | 1572.7 | 3863.0 | 250.7 | -7.6 | 33.2 | -26.9 | 1575.2 | 3867.1 | 256.7 | -9.2 | 31.5 | -25.3 | 1576.8 | 3866.2 | 251.6 | -9.0 | 30.9 | -25.2 | no |
| image 364 | x | 1550.0 | 3761.2 | 256.0 | 1.3 | 15.5 | -90.1 | 1487.4 | 3694.6 | 203.5 | 2.3 | 14.8 | -76.5 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 365 | r | 1508.5 | 3806.5 | 258.2 | 6.1 | 37.0 | 172.2 | 1512.3 | 3809.6 | 264.1 | 3.6 | 36.9 | 170.6 | 1512.0 | 3805.8 | 263.3 | 2.7 | 36.9 | 169.8 | no |
| image 366 | d | 1500.5 | 3935.4 | 262.5 | -3.7 | 40.1 | 154.6 | 1504.9 | 3942.3 | 265.6 | -6.2 | 40.8 | 153.7 | 1504.9 | 3946.1 | 261.2 | -4.8 | 40.2 | 154.4 | no |
| image 367 | d | 1488.9 | 4093.2 | 267.0 | -6.5 | 38.1 | 150.2 | 1492.1 | 4094.6 | 274.7 | -7.7 | 39.0 | 151.2 | 1494.8 | 4091.6 | 264.3 | -7.7 | 37.2 | 151.0 | no |
| image 368 | c | 1480.3 | 4267.6 | 269.5 | -7.4 | 40.1 | 149.9 | 1483.9 | 4266.5 | 273.5 | -5.3 | 41.0 | 150.5 | 1500.8 | 4268.5 | 261.5 | -1.8 | 37.5 | 153.4 | no |
| image 369 | c | 1464.8 | 4453.4 | 272.2 | -4.4 | 38.5 | 153.0 | 1477.9 | 4460.8 | 268.1 | -0.9 | 36.1 | 157.8 | 1472.9 | 4446.3 | 273.5 | -5.0 | 36.7 | 153.6 | few corners |
| image 370 | c | 1447.3 | 4644.1 | 278.5 | -5.4 | 36.0 | 155.9 | 1450.6 | 4643.5 | 279.8 | -4.7 | 36.3 | 156.5 | 1452.9 | 4646.3 | 284.1 | -4.4 | 36.9 | 157.4 | no |
| image 371 | c | 1427.2 | 4839.1 | 279.2 | -4.8 | 34.5 | 158.3 | 1432.2 | 4836.2 | 282.3 | -6.5 | 33.2 | 155.9 | 1436.0 | 4838.4 | 276.7 | -7.2 | 32.2 | 156.7 | no |
| image 372 | c | 1413.1 | 5034.4 | 281.0 | -8.2 | 33.6 | 156.9 | 1420.7 | 5032.0 | 283.4 | -7.8 | 33.9 | 156.9 | 1418.9 | 5029.0 | 281.3 | -8.8 | 33.6 | 156.3 | no |
| image 373 | c | 1392.0 | 5200.4 | 282.7 | -13.9 | 37.9 | 165.0 | 1401.0 | 5183.6 | 258.7 | -24.5 | 32.8 | 161.7 | 1392.8 | 5197.7 | 292.9 | -15.7 | 41.0 | 165.8 | few corners |
| image 374 | c | 1335.6 | 5283.8 | 289.0 | -16.9 | 47.3 | -133.9 | 1321.4 | 5298.6 | 270.6 | -5.1 | 43.0 | -126.3 | 1343.1 | 5282.4 | 289.5 | -12.2 | 48.1 | -121.3 | few corners |
| image 375 | c | 1274.7 | 5233.2 | 282.7 | -2.0 | 46.3 | -33.3 | 1274.9 | 5224.8 | 283.4 | -0.7 | 42.1 | -27.3 | 1279.2 | 5229.0 | 280.9 | -1.3 | 41.9 | -28.9 | no |
| image 376 | c | 1284.7 | 5104.6 | 275.2 | 3.7 | 32.3 | -19.5 | 1276.7 | 5105.9 | 285.5 | 5.5 | 33.3 | -18.4 | 1284.6 | 5102.5 | 275.5 | 4.0 | 33.1 | -18.5 | no |
| image 377 | c | 1295.2 | 4951.7 | 271.2 | 0.8 | 32.8 | -25.5 | 1296.3 | 4957.5 | 271.4 | 0.2 | 32.1 | -23.8 | 1297.9 | 4957.8 | 275.6 | 0.4 | 32.8 | -24.0 | no |
| image 378 | c | 1303.9 | 4785.8 | 267.2 | -1.3 | 33.5 | -24.1 | 1306.8 | 4791.9 | 264.7 | -1.9 | 34.1 | -23.4 | 1306.6 | 4785.2 | 271.3 | -1.2 | 35.1 | -22.0 | no |
| image 379 | c | 1314.8 | 4615.5 | 264.0 | -4.9 | 34.5 | -25.4 | 1315.7 | 4616.0 | 263.7 | -3.8 | 33.8 | -22.6 | 1315.8 | 4617.0 | 253.5 | -3.7 | 32.2 | -22.4 | no |
| image 380 | d | 1326.9 | 4442.5 | 265.5 | -1.4 | 33.6 | -25.0 | 1329.6 | 4444.5 | 271.5 | -1.9 | 35.1 | -23.8 | 1331.5 | 4444.1 | 262.5 | -2.8 | 34.2 | -24.2 | no |

*c: campus
r: residential area
d: downtown
x: erroneous data acquisition

**x: erroneous data acquisition
no: correct pose

| | | NAV420CA Reading | | | | | | Corner Correspondence | | | | | | Manual Correspondence | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Area* | X | Y | Z | roll | pitch | yaw | X | Y | Z | roll | pitch | yaw | X | Y | Z | roll | pitch | yaw | source of error** |
| | | (m) | (m) | (m) | (˚) | (˚) | (˚) | (m) | (m) | (m) | (˚) | (˚) | (˚) | (m) | (m) | (m) | (˚) | (˚) | (˚) | |
| image 381 | d | 1338.4 | 4267.4 | 267.5 | -3.9 | 34.3 | -28.5 | 1341.3 | 4268.3 | 270.8 | -4.1 | 35.0 | -27.7 | 1342.3 | 4269.4 | 271.5 | -4.4 | 35.1 | -28.0 | no |
| image 382 | d | 1351.6 | 4095.2 | 273.2 | -5.8 | 38.7 | -31.3 | 1353.0 | 4096.8 | 276.1 | -4.4 | 37.8 | -28.5 | 1353.6 | 4096.0 | 266.8 | -3.0 | 37.0 | -28.0 | no |
| image 383 | d | 1361.3 | 3930.3 | 276.7 | -2.0 | 30.2 | -40.1 | 1363.4 | 3932.2 | 277.9 | -0.1 | 29.3 | -38.9 | 1371.6 | 3931.1 | 274.4 | 0.0 | 29.4 | -39.2 | no |
| image 384 | x | 1313.2 | 3827.1 | 285.7 | 4.9 | 18.2 | -96.4 | 1162.7 | 3875.1 | 199.1 | 10.7 | 5.8 | -75.5 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 385 | r | 1254.0 | 3876.8 | 287.5 | 7.3 | 37.5 | 173.6 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 386 | r | 1240.4 | 4003.7 | 286.2 | -1.5 | 40.2 | 150.5 | | | | | | | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 387 | r | 1231.8 | 4158.9 | 288.7 | -2.9 | 35.2 | 151.9 | 1238.6 | 4162.9 | 257.9 | -2.7 | 31.4 | 155.4 | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 388 | r | 1223.2 | 4327.2 | 292.7 | -4.1 | 41.1 | 148.0 | 1223.7 | 4302.9 | 285.5 | -10.1 | 44.5 | 140.4 | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 389 | r | 1211.3 | 4508.7 | 294.5 | -5.1 | 40.2 | 147.8 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 390 | r | 1197.4 | 4696.7 | 293.7 | -5.2 | 39.7 | 149.1 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 391 | r | 1180.8 | 4886.3 | 298.7 | -5.2 | 35.3 | 152.4 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 392 | r | 1169.4 | 5070.1 | 300.7 | -4.7 | 38.8 | 150.7 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 393 | r | 1133.2 | 5209.8 | 308.0 | -27.6 | 51.2 | -171.6 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 394 | r | 1045.2 | 5201.2 | 305.5 | -3.8 | 55.7 | -65.6 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 395 | c | 1041.3 | 5074.8 | 299.2 | 0.7 | 27.6 | -28.1 | 1054.9 | 5082.8 | 307.4 | 4.6 | 33.7 | -28.5 | 1048.6 | 5074.9 | 298.7 | 2.3 | 32.7 | -22.7 | random |
| image 396 | c | 1056.2 | 4920.9 | 297.7 | 2.8 | 37.0 | -27.5 | 1053.0 | 4920.6 | 310.8 | 3.8 | 38.8 | -25.4 | 1061.7 | 4928.6 | 299.7 | 0.2 | 37.2 | -27.3 | no |
| image 397 | c | 1067.0 | 4743.7 | 288.7 | 1.1 | 35.3 | -28.4 | 1064.7 | 4737.0 | 292.4 | 5.1 | 37.0 | -24.0 | 1068.1 | 4743.4 | 289.6 | 4.1 | 36.8 | -25.3 | random |
| image 398 | c | 1077.4 | 4557.9 | 280.2 | -2.5 | 34.4 | -31.1 | 1076.9 | 4558.9 | 282.8 | -1.3 | 35.0 | -28.8 | 1080.8 | 4564.0 | 285.0 | -3.2 | 35.2 | -30.2 | no |
| image 399 | d | 1085.9 | 4375.5 | 274.2 | -1.4 | 42.2 | -29.4 | 1086.9 | 4375.3 | 284.9 | -3.0 | 44.0 | -28.4 | 1085.9 | 4375.1 | 273.3 | -1.6 | 42.4 | -27.9 | no |
| image 400 | d | 1093.5 | 4185.8 | 268.7 | -8.0 | 41.6 | -37.2 | 1092.6 | 4188.6 | 266.2 | -6.0 | 41.3 | -34.0 | 1102.8 | 4186.8 | 274.3 | -7.4 | 43.3 | -35.1 | no |
| image 401 | r | 1105.3 | 3998.5 | 262.5 | -8.9 | 38.8 | -40.7 | 1108.4 | 4002.2 | 264.3 | -8.0 | 39.0 | -39.8 | 1111.5 | 4007.0 | 267.0 | -9.4 | 39.1 | -41.0 | no |
| image 402 | x | 1108.3 | 3836.3 | 257.2 | -4.4 | 19.4 | -52.7 | 1073.3 | 3750.4 | 369.8 | 8.7 | 32.5 | -42.0 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 403 | x | 1055.6 | 3769.7 | 268.5 | 11.0 | 21.2 | -129.9 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 404 | r | 1022.6 | 3834.3 | 273.7 | 3.3 | 41.1 | 149.8 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 405 | r | 1019.0 | 3954.6 | 277.0 | -0.8 | 42.2 | 142.2 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 406 | r | 1013.9 | 4104.0 | 276.2 | -4.2 | 36.1 | 137.1 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 407 | x | 1014.6 | 4265.2 | 278.0 | -6.2 | 37.5 | 137.9 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 408 | x | 1009.8 | 4433.7 | 283.5 | -4.1 | 38.4 | 139.0 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 409 | r | 1006.1 | 4611.3 | 287.7 | -1.5 | 39.1 | 144.2 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 410 | r | 998.2 | 4798.4 | 290.0 | -2.4 | 39.0 | 147.7 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 411 | x | 982.6 | 4980.9 | 293.5 | -0.8 | 29.8 | 150.0 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 412 | x | 940.5 | 5121.3 | 305.5 | -22.1 | 40.1 | -175.9 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 413 | r | 858.9 | 5122.2 | 302.5 | -5.0 | 48.1 | -78.1 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 414 | x | 873.0 | 5018.7 | 296.2 | -3.0 | 44.4 | -6.7 | 963.7 | 5059.0 | 371.3 | -29.0 | 61.8 | -29.3 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 415 | x | 997.2 | 4918.0 | 291.2 | 6.7 | 34.1 | 26.6 | 998.3 | 4907.3 | 284.5 | 9.0 | 32.6 | 28.3 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 416 | x | 1169.4 | 4818.1 | 287.7 | 2.5 | 35.1 | 26.1 | 1167.4 | 4818.8 | 289.4 | 1.6 | 33.6 | 25.5 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 417 | x | 1371.5 | 4713.8 | 285.2 | 1.5 | 34.9 | 29.2 | 1368.4 | 4713.4 | 288.6 | 1.1 | 35.3 | 29.9 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 418 | x | 1589.3 | 4601.9 | 284.7 | 0.9 | 33.2 | 32.6 | 1585.3 | 4611.2 | 283.5 | -1.7 | 31.9 | 30.4 | N/A | N/A | N/A | N/A | N/A | N/A | x |
| image 419 | r | 1813.1 | 4484.7 | 280.2 | 2.6 | 34.4 | 30.7 | 1818.9 | 4475.9 | 284.1 | -2.5 | 29.7 | 26.0 | 1805.7 | 4490.5 | 283.2 | 0.9 | 34.6 | 31.6 | random |
| image 420 | r | 2038.7 | 4362.6 | 282.2 | -2.0 | 34.5 | 27.7 | 2018.7 | 4346.3 | 282.6 | -5.2 | 31.1 | 30.6 | 2034.1 | 4368.0 | 280.0 | -0.7 | 33.1 | 29.7 | no |
| image 421 | r | 2263.8 | 4251.1 | 285.0 | -0.1 | 34.2 | 32.5 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | few corners |
| image 422 | x | 2483.9 | 4137.5 | 285.5 | -1.1 | 12.1 | 9.0 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | x |

*c: campus
r: residential area
d: downtown
x: erroneous data acquisition

**x: erroneous data acquisition
no: correct pose