

Universität Stuttgart

Master Thesis

**Automatic 3D lane marking
reconstruction using multi-view
aerial imagery**

Chun-Yu Sheu

Course of Study: GEOENGINE

Examiner: apl. Prof. Dr.-Ing. Norbert Haala

Supervisor: Dr.-Ing Franz Kurz

Commenced: 3. Juli 2017

Completed: 1. February 2018

Abstract

... This paper exploits the use of linear regression in image space for 3D line reconstruction.
[... will be finished after finishing all the chapters]

keywords: line reconstruction, linear regression, least-squares adjustment

Contents

1	Introduction	11
1.1	Motivation	11
1.2	Autonomous Driving	11
1.3	3D Reconstruction	12
1.4	Automatic Line Detection and Line Matching	13
1.5	Related Work	14
1.6	Purpose	15
2	Methodologies	17
2.1	Lane markings Properties and Automatic Extraction	19
2.2	Imaging Properties of Aerial Photographs	21
2.3	Line Fitting	23
2.4	3D Line Reconstruction with Nonlinear LS Adjustment	25
2.5	Line Projection on the DSM (Determination of Initial Parameter Estimates) .	34
3	Experimental Results and Evaluations	35
3.1	Materials	35
3.2	Preprocessing	43
3.3	Simulation	43
3.4	True Data	50
4	Conclusion and Future Work	55
	Bibliography	59

List of Figures

2.1	The work flow	18
2.2	Line types of lane markings [source RMS Teil 1]	19
2.3	Lane markings Extraction. The extracted long lane-lines are marked in green and the dashed ones are in yellow. Note that both cases are reconstructed into 3D with the same framework; different colors here are only for illustration.	20
2.4	Before reconstruction, the back projection of initial approximate 3D line segment does not best-fit the extracted 2D lines in the covering images. After optimization, the back-projection of the reconstructed line segment should best fitting the extracted 2D lines in all the covering images.	26
2.5	A line is partially reconstructed through a sliding window.	27
2.6	Measurements collection in image space.	28
2.7	The red circle points out the reconsidered measurements in successive sliding windows.	28
3.1	Flight trajectory of DLR helicopter visualized on the Google Earth platform. The green polyline shows the flight trajectory. <i>Source: Google Earth 04/01/2017</i>	36
3.2	Part of the DSM in road area. It is noisy in the center of motorway.	39
3.3	Standard deviations of the height value of the DSM in road area. It has higher value in the center of motorway.	39
3.4	Distribution of stereo pairs used for DSM generation. Lighter color indicates more stereo pairs are used in that area. Maximum 27 stereo pairs are used for one pixel.	40
3.5	Original Image	42
3.6	Orthorectified Image	42
3.7	Masked Image	42
3.8	The added Gaussian random noise in the observations.	44
3.9	The reconstructed line segments and the true line segments in UTM coordinate system (in Zone 32N).	46
3.10	The reconstructed line segments and the unrefined DSM profile in UTM coordinate system (in Zone 32N).	46
3.11	XXX.	47
3.12	The relation between the added random Gaussian noise and the adjusted residuals.	48
3.13	The red area under the probability density function is the rejection region, which is 5%. Since none of the T_{obs} falls in the rejection region, the null hypothesis could not be rejected.	48
3.14	The relation between the variances of the estimated parameters and the amount of images.	49

3.15 The reconstructed line segments and the unrefined DSM profile in UTM coordinate system (in Zone 32N).	51
3.16 Histogram of the distances from the reconstructed line to the unrefined DSM profile.	51
3.17 XXX...The relationship between image amount, the resulting reconstructed line segments, and the redundancies and posterior standard deviation in LS adjustment.	52
3.18 The variances of the estimated object coordinates, in horizontal and vertical directions.	53
3.19 The estimated variances of the estimated object coordinates, in horizontal and vertical directions.	53

List of Tables

1.1	Summary of level of driving automation [Int14]. System refers to the driver assistance system, combination of driver assistance systems, or automated driving system.	12
2.1	Widths of lane markings [source RMS Teil 1]	19
2.2	Lengths of dashed lane markings with ratio 1:2 [source RMS Teil 1]	19
3.1	Properties of the oblique camera	36
3.2	Viewing geometry	36
3.3	Accuracies of Exterior Orientations	38
3.4	Interior Orientations and their accuracies	38

1 Introduction

1.1 Motivation

The availability of large-scale, accurate high resolution 3D information of roads with lane markings and road furniture plays an important role towards autonomous driving. Aerial imagery is a valuable database to derive 3D information of roads even in areas difficult to access, like on motorways. Compared to optical satellite data, acquiring large-scale 3D lane markings by optical aerial imagery is more efficient and has higher accuracy and spatial resolution. In view of the fact, that in Germany exists no area-wide, high resolution 3D information of the road surfaces including lane markings, new methods to derive this information are demanded.

The standard workflow with aerial images would be to project the images onto a DSM and to derive the information in the projected imagery, but the generation of Digital Surface Model (DSM) from stereo images is challenging in the regions with low textures. The lane markings, for example, are the most visible texture on asphalt roads useful for 3D reconstruction. Thus, it is desired to improve the quality of the DSM on the road surfaces by exploiting the line character of the lane markings.

1.2 Autonomous Driving

Autonomous driving is an important part of future transportation systems, and the execution of dynamic driving task is a key issue towards autonomous driving. According to the definition given by Society of Automotive Engineers (SAE) International, dynamic driving task includes the operational (e.g. steering, braking, accelerating, monitoring the vehicle and roadway) and tactical (responding to events, determining when to change lanes, turn, use signals, etc.) aspects. In the information report J3016 [Int14] proposed by International, driving automation is identified into 6 levels from “no automation” to “full automation” as expressed in Table 1.1.

For autonomous driving, redundant sources of information for the driving environment are useful for increasing the robustness and availability of the system [ARB+15]. It is especially important for level 3, 4 and 5 driving automations where the execution of dynamic driving tasks is totally conducted by the system. 3D lane markings, for example, provide global environment information and can be used on supporting lane-accurate localization.

1 Introduction

SAE level	Name	Execution of Steering and braking	Monitoring of driving environment	Fallback Performance of Dynamic Driving Task	System Capability
Human driver monitors the driving environment					
0	No Automation	Human	Human	Human	n.a.
1	Driver Assistance	Human & System	Human	Human	some driving modes
2	Partial Automation	System	Human	Human	some driving modes
Automated driving system monitors the driving environment					
3	Conditional Automation	System	System	Human	some driving modes
4	High Automation	System	System	System	some driving modes
5	Full Automation	System	System	System	all driving modes

Table 1.1: Summary of level of driving automation [Int14]. System refers to the driver assistance system, combination of driver assistance systems, or automated driving system.

1.3 3D Reconstruction

Generally, the procedure of 3D objects reconstruction consists of feature extraction in image space and depth information recovery in object space. To reconstruct the depth information at the exposure moment, either multiple rays spatial intersection or single ray intersection with an elevation model can be applied. Spatial intersection is usually applied in the cases that the correspondences of the extracted features among different views can be established. Alternatively, when Digital Elevation Model (DEM) is available, single extracted feature can be directly projected onto the DEM. In this case, the quality of DEM directly influences the result of 3D object reconstruction.

1.3.1 Feature Extraction and Matching

The aim of feature extraction is to gain the characteristics of the images, through which the stereo correspondence processes. As a result, the characteristics of the images closely link to the choice of matching methods.

Blob features have properties of being local intensity maximum or minimum in images. Edge features have image brightness discontinuities in the direction perpendicular to the line direction itself. Variance of algorithms have been proposed for different kinds of

features detection. The rotation-invariant Harris corner detector, for example, is commonly used to extract corners and infer features of an image.

The features are then matched among different views by comparison of the patches which center on the extracted features. Typically the similarity is measured by taking the Sum of Squared Differences (SSD) or Normalized Cross-Correlation (NCC) between the corresponding pixels of two patches.

Scale-Invariant Feature Transform (SIFT) is proposed by Lowe in 1999 [Low99]. Lowe's approach transforms an image into a large collection of local feature vectors, also known as keypoint descriptors. Each SIFT feature descriptor is invariant to image translation, scaling, and rotation, partially invariant to illumination changes and affine or 3D projection. These SIFT keypoints are then matched by identifying their nearest neighbors.

1.3.2 Dense Image Matching (DIM)

Dense image matching performs matching at the actual image resolution, i.e. pixel-wise correspondence between MVS is to be recovered. Depending on image texture, a per-pixel measure is generally ambiguous. Additional constraints, such as the assumption of a smooth surface, need to be introduced. Proposed by Hirschmüller et al. in 2008, Semi-Global Matching (SGM) approximates the two-dimensional global aggregation of matching cost by a number of one-dimensional cost path, where the matching step is casted into an energy minimization problem [HB08]. It not only achieves similar accuracy as truly global matching but also significantly reduces computational complexity.

1.4 Automatic Line Detection and Line Matching

Automatic detected lines may be used for 3D reconstruction by matching lines in the image space. Line matching is challenging for several reasons.

Firstly, line segments may be detected inexactly by automatic line detectors or obstructions appear in part of the lines. Consequently the end-points of a line segment often do not correspond to each other in different views.

Secondly, there is no strong disambiguating geometric constraint available [SZ97]. In the case of points, correspondences must satisfy the epipolar constraint. This strong disambiguating constraint helps to efficiently reduce the searching space from the whole image (2D) to a single line (1D) in matching processes. In the case of infinite line matching, however, there is no geometric constraint. For lines of finite length, there is only a weak overlap constraint arising from applying the epipolar constraint to its end-points.

Moreover, the corresponding neighborhoods may well have a very different shape and orientation in different views, or even totally different surroundings when dealing with wiry objects [HWB13].

In the cases of lane markings, they may be partly shaded by vehicles in aerial images. Besides, the continuous lane lines have no endpoints in the images. Worst of all, the asphalt road surface where the lane markings locate on is poorly textured. Therefore, line matching is even hardly applicable on lane markings.

1.5 Related Work

In the following, some works regarding 3D line reconstruction is presented.

First, appearance-based methods are described. For 3D line reconstruction, [BFG05; SZ97; WWH09] have tried to match line segments based on their appearances or some additional geometry constraints.

Schmid et al. exploit the epipolar geometry of line segments and the one-parameter family of homographies to provide point-wise correspondences, allowing cross-correlation of patches around line segments along the candidate lines in the epipolar-beam-region for matching scores evaluation [SZ97].

In the cases of poorly textured or shape-changing neighborhood of line segments in different views, line segments are barely comparable using classical correlation patches yet the color neighborhood along this line segment undergoes only slight changes. Based on color histogram rather than textures, Bay et al. exploit the appearance similarity of line segment pairs and their topological layout to iteratively increase the correct matches [BFG05]. If region matches are available, they are automatically integrated and exploited in combination. The final coplanar grouping stage allows to estimate the fundamental matrix even from line segments only. While color provides a very strong cue for discrimination, it may fail in the case where color feature is not distinctive, e.g. gray images. Besides, although matching groups of line segments takes more geometric information into account for disambiguation, the disadvantage is the increased computational complexity.

Without resorting to any other constraints or prior knowledge, Wang et al. propose a purely image content-based line descriptor MSLD for automatic line segments matching. Adapting SIFT-like strategy, MSLD is highly distinctive and robust against image rotation, illumination change, image blur, viewpoint change noise, JPEG compression and partial obstruction [WWH09]

The above appearance-based approaches demand either constant and rich neighboring textures or similar color profile of line segments, they are technically matching the surroundings instead of the lines themselves.

In order to create 3D models without the need of explicit line matching, Jain et al. generate all possible hypothetical straight 3D line segments by triangulating all the detected straight 2D line segments from different views [JKTS10]. They then keep the one whose back projection on the gradient images of neighboring views has the highest score, assuming that line features correspond to high gradient areas in images. Built upon the same principles whilst applying epipolar constraint on the end-points of line segments, Hofer et al. generate

less hypothetical 3D line segments and thus increase performance significantly while still creating accurate results [HWB13]. However, both approaches are barely possible in the case of infinite line reconstruction, where the detected 2D lines in different views do not exactly correspond to the same part of a 3D line.

Taylor et al. formulate the Structure from Motion (SfM) problem in terms of minimization of an objective function which measures the total squared distance in the image plane between the observed edge segments and the projections of the reconstructed lines [TK95]. By reconstructing the infinite straight line that supports the observed edge segments rather than the end-points of the line, the algorithm can be used even when multiple edges in a single image correspond to different portions of the same 3D line.

1.6 Purpose

In this thesis, I develop a framework to automatically detect the lane markings in the unprojected aerial imagery, and refine the 3D information of the road surface by exploiting the line character of the lane markings.

The unprojected aerial images with their bundle-adjusted orientations and the DSM are the inputs of my algorithm. I apply some standard pre-processing steps and a standard line detection algorithm for automatic lane marking detection in image space. By sliding a window of reasonable length and width through the curved long lane lines, I collect all line segments in all covering images assuming the lane markings to be straight in each sliding window.

I investigate the use of linear regression to optimize the 3D position of each line segments in object space so that its back projection would best fit the detected 2D line in all the covering views, i.e. the position and height of each 3D lane marking segments will be refined in one optimization step. Using the aerial image data set with special flight configuration at both sides of the motorway, the proposed approach addresses the problematic (quasi) infinite and curved properties of lane markings in the 3D reconstruction.

The framework will be tested on aerial imagery from the German highway A9.

2 Methodologies

The following sections introduce the principles of 3D lane marking reconstruction method of this work, with the work flow shown in Figure 2.1.

Section 2.1 describes the applied standard line detection algorithm for labeling the lane markings. To relate the object coordinates of a point with its image coordinates, Section 2.2 introduces the imaging properties of aerial images and their mathematical models, including the collinearity equation and lens distortion correction.

Section 2.3 presents the principle of line fitting and further derives the nonlinear Least-Squares (LS) model for line equations in two-point form. With the combination of the extended collinearity equation introduced in Section 2.2, Section 2.4 elaborates the usage of line fitting for 3D lane marking reconstruction.

In Section 2.5 the problem of acquiring initial values is described, as initial values of unknown quantities are required in nonlinear LS model. Section 2.7??? demonstrates how the corresponding measurements in image space are collected given the initial values in object space.

2 Methodologies

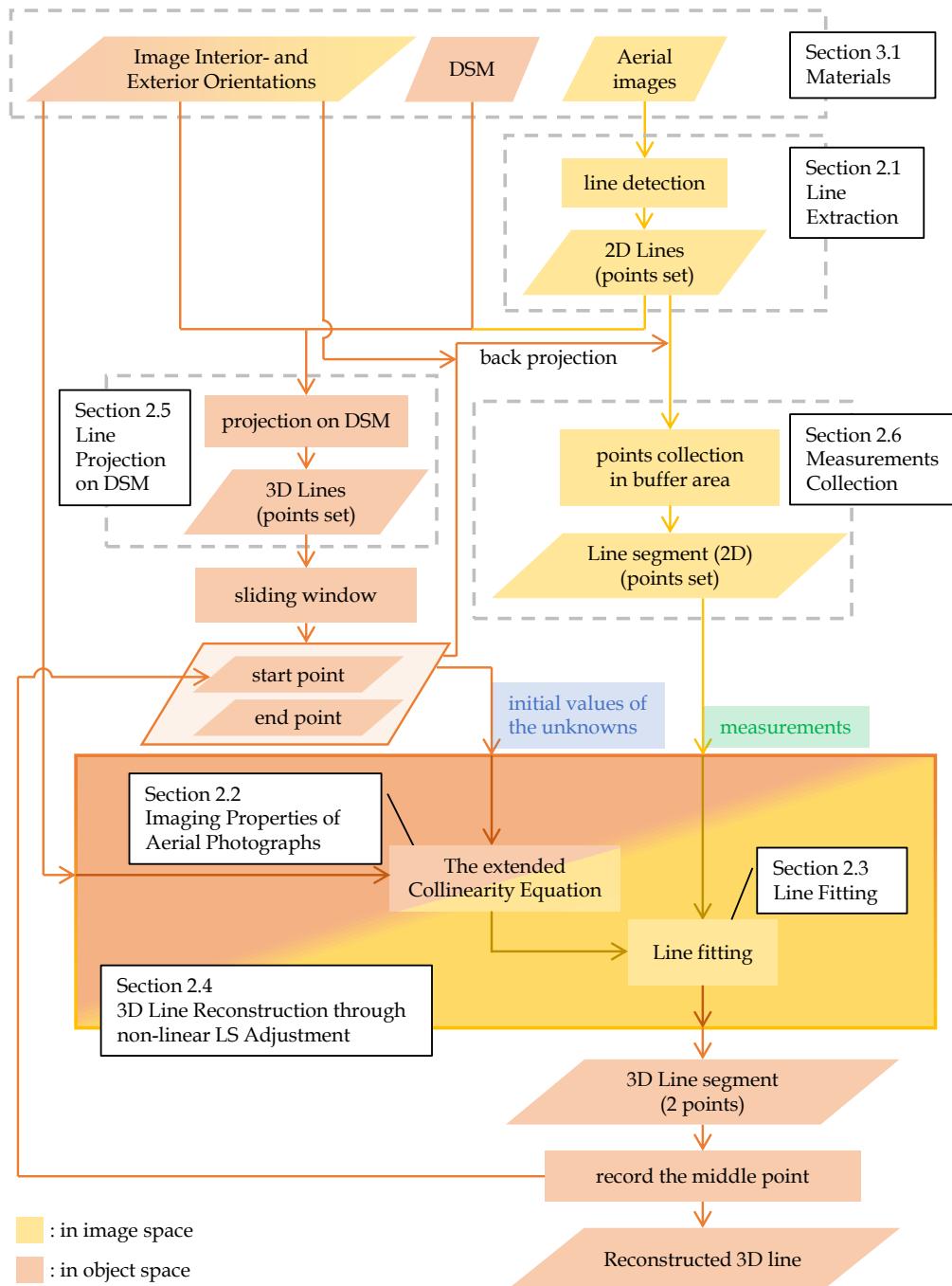


Figure 2.1: The work flow

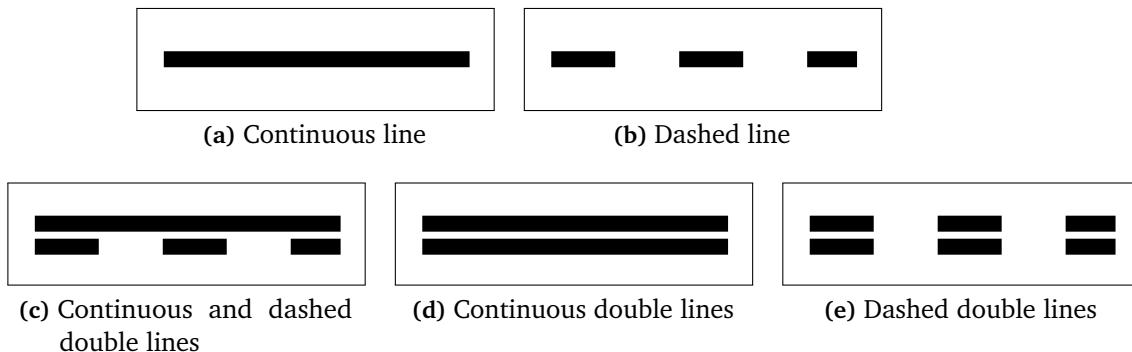


Figure 2.2: Line types of lane markings [source RMS Teil 1]

2.1 Lane markings Properties and Automatic Extraction

The appearance of lane markings on German roads including line type, color and width is specified depending on the road type. Different line types of lane markings are shown in Figure 2.2 and their line widths are defined in Table 2.1. As shown in Table 2.2, the dashed lane markings on motorways are of 6 meter length.

Because of the appearance, the problem of lane marking detection can be treated as a line detection problem. We restrict the proposed framework to lane markings with single white lines (dashed or continuous) of 0.3 meter width. Other types like in restricted zone, double lines, parking areas, temporal yellow lines in construction sites etc, are excluded.

	motorways ¹	other roads
narrow lines	0.15 [m]	0.12 [m]
wide lines	0.30 [m]	0.25 [m]

Table 2.1: Widths of lane markings [source RMS Teil 1]

	motorways ¹		other roads	
			in town	out of town
line / gap	6 [m] / 12 [m]	3 [m] / 6 [m]	4 [m] / 8 [m]	

Table 2.2: Lengths of dashed lane markings with ratio 1:2 [source RMS Teil 1]

¹and corresponding roads in the sense of the VwV-StVO to § 42 to mark 330 (motorway) II

2 Methodologies

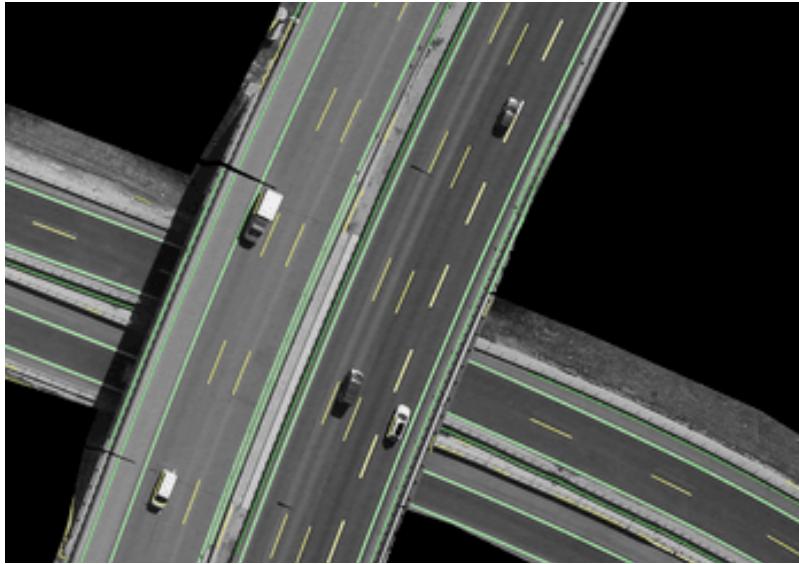


Figure 2.3: Lane markings Extraction. The extracted long lane-lines are marked in green and the dashed ones are in yellow. Note that both cases are reconstructed into 3D with the same framework; different colors here are only for illustration.

There are many algorithms for line detection. Prewitt line detector uses two orthogonal gradient operators, and the pixels in the operators are of same weights. Sobel detector also uses two orthogonal gradient operators, but the weights of pixel in operators are not equal—the closer the pixel to the center of operator, the higher weight it has. Canny edge detector searches local extrema of gradient to locate the positions of line features and is still a state-of-the-art edge detector. Edge-detectors that perform better than the Canny usually require higher computational complexities or a greater number of parameters. Edge drawing [TA12] first spots anchors along rows and columns by Sobel detector, and then joins these anchors to extract line features.

In this work, the principle to extract line features is to firstly derive the line direction for each pixel by using partial derivatives of a Gaussian smoothing kernel. Pixels that have a local maximum in the second directional derivative perpendicular to the line direction are marked as line points. By thresholding their second directional derivative values, the accepted line points are then linked and connected. The resultant connected points which compose a line are of sub-pixel precision. Figure 2.3 shows the extracted lines on part of the masked original image.

2.2 Imaging Properties of Aerial Photographs

This section describes the geometric model of the projection of 3D points into the image generated by a real camera. We first restrict the discussion in Section 2.2.1 to central perspective projection where the collinearity equation originate from. We then model deviations from this model, addressing real cameras with imperfect lenses, in Section 2.2.2.

2.2.1 Collinearity Equations

We assume frame photography, i.e. photographs exposed on a frame chip in one instant, and assume central projection model with cameras that have a single viewpoint and a planar sensor and being straight line-preserving. Collinearity indicates the condition that the image point (on the sensor plate of the camera), the observed point (in object space) and the projection center of the camera were aligned at the moment the picture was taken. Every measured point leads to two collinearity equations, describing transformations from object space to image coordinates:

$$\begin{aligned} x &= x_0 - c \frac{r_{11}(X - X_0) + r_{21}(Y - Y_0) + r_{31}(Z - Z_0)}{r_{13}(X - X_0) + r_{23}(Y - Y_0) + r_{33}(Z - Z_0)} \\ y &= y_0 - c \frac{r_{12}(X - X_0) + r_{22}(Y - Y_0) + r_{32}(Z - Z_0)}{r_{13}(X - X_0) + r_{23}(Y - Y_0) + r_{33}(Z - Z_0)} \end{aligned} \quad (2.1)$$

where

(x, y) : image coordinates of the point

(x_0, y_0) : image coordinates of principal point

c : principal distance; focal length

(X, Y, Z) : object coordinates of the point

(X_0, Y_0, Z_0) : object coordinates of projection center

r_{11}, \dots, r_{33} : elements of the rotation matrix R (orthogonal 3×3 -matrix from object space to image space, with 3 independent angles ω , ϕ and κ)

2.2.2 Lens Distortion Correction

An original image appears to have some degree of deviations from perspective mapping due to lens distortion, lens refraction or non-planarity of the sensor surface. There are several models to describe these perturbing effects and can be used to undistort the images, resulting in rectified images which are now straight line-preserving.

A subset of physical distortion model [Fra97] is chosen, with two radial symmetric distortion parameters A_1 and A_2 , two asymmetric parameters B_1 and B_2 , and a scaling C_1 and an affine shearing parameter C_2 . Assuming x and y to be the distorted image coordinates, the corrections Δx and Δy are then calculated by the following equations:

$$\begin{aligned} \Delta x &= x_p + A_1 x_* (r^2 - R_0^2) + A_2 x_* (r^4 - R_0^4) + B_1 (r^2 + 2x_*^2) + B_2 2x_* y + C_2 y \\ \Delta y &= y_p + A_1 y (r^2 - R_0^2) + A_2 y (r^4 - R_0^4) + B_1 (r^2 + 2y^2) + B_2 2x_* y \end{aligned} \quad (2.2)$$

2 Methodologies

with $r = \sqrt{x_*^2 + y^2}$, $x_* = \frac{x}{C_1}$ and radius² R_0 being set to a third of the sensor diagonal.

The undistorted image coordinates x' and y' are then calculated by

$$\begin{aligned} x' &= x + \Delta x \\ y' &= y + \Delta y \end{aligned} \quad (2.3)$$

2.2.3 Extended Collinearity Equation

As real cameras generally only approximate the perspective camera model, lens distortion correction can be additionally included in the collinearity model, attempting to correct the pixel position so that they obey the perspective model with sufficient accuracy.[W. Förstner et al. 2016]

By inserting (2.1) and (2.2) into (2.3) , the relationship between a 3D point $\mathbf{P}(X, Y, Z)$ and its corresponding distorted image coordinates $\mathbf{p}(x, y)$ can be described as

$$\begin{aligned} x &= x_0 - c \frac{r_{11}(X - X_0) + r_{21}(Y - Y_0) + r_{31}(Z - Z_0)}{r_{13}(X - X_0) + r_{23}(Y - Y_0) + r_{33}(Z - Z_0)} \\ &\quad - (x_p + A_1 x_*(r^2 - R_0^2) + A_2 x_*(r^4 - R_0^4) + B_1(r^2 + 2x_*^2) + B_2 2x_*y + C_2 y) \\ y &= y_0 - c \frac{r_{12}(X - X_0) + r_{22}(Y - Y_0) + r_{32}(Z - Z_0)}{r_{13}(X - X_0) + r_{23}(Y - Y_0) + r_{33}(Z - Z_0)} \\ &\quad - (y_p + A_1 y(r^2 - R_0^2) + A_2 y(r^4 - R_0^4) + B_1(r^2 + 2y^2) + B_2 2x_*y) \end{aligned} \quad (2.4)$$

To express (2.4) shortly, a function \mathcal{G} is defined as

$$\mathbf{p} = \mathcal{G}(\mathbf{q}, \mathbf{P}) \quad (2.5)$$

which takes the interior and exterior orientations as well as the lens distortion parameters of a camera $\mathbf{q}(x_0, y_0, c, X_0, Y_0, Z_0, r_{11}, \dots, r_{33}, A_1, A_2, B_1, B_2, C_1, C_2)$ and the position of a 3D point $\mathbf{P}(X, Y, Z)$, and returns the corresponding distorted image coordinates $\mathbf{p}(x, y)$.

²At the radius R_0 the radial symmetric distortion is zero by definition, which avoids too high distortion values at the edges and reduces the correlation with the focal length.

2.3 Line Fitting

Line fitting is the process of constructing a infinite straight line that has a best fit to a 2D dataset. One of the approaches is linear regression which attempts to find the linear function that "best" predicts the dependent variable values as a function of the independent variable. In this work, "best" predict will be understood as in the LS approach: minimization of the sum of squared residuals (differences between the measured and the estimated values of the dependent variable).

In the case of standard linear regression, the regressor x is assumed error free, inconsistencies³ are only for the dependent variable y . Geometrically it means that the vertical distances from observed data to the fitted line is minimized. To minimize the perpendicular distances from the data points to the regression line, a orthogonal regression model is derived in Section 2.3.1.

For a later combination with point-wise extended collinearity equation (2.5) in next section, the aim is to fit the line equation in two-point form to the observed dataset. For such nonlinear functional relation between variales, a nonlinear LS model is derived in Section 2.3.2.

A functional model is unsolvable when the assumed "dependent" variable is indeed not a function of the independent variables, i.e. the assumed functional relation does not really exist. Take an observed set of 2D points with their Cartesian coordinates $\{x_i, y_i\}_{i=1}^n$ on a vertical line $x = \text{constant}$ for example. Their y values have no dependency on their x values, i.e. knowledge of x tells nothing about y . Therefore, for this dataset, the functional model $y = f(x)$ is barely solvable. In such cases, however, x is a function of y (which is actually a constant function) and the equation system which models the dependent variable x being a function of the independent variable y becomes solvable. Regarding the dataset used in this work (which will be described with more details in Section 3.1) where the observed 2D points scatter mainly in column direction in image space, the functional relation between variables x and y will be setup as $x = f(y)$ to avoid weakly solvable equations system.

2.3.1 Orthogonal Regression

A linear regression model describes a dependent variable as a linear function of the regressor (an independent variable). Given a dataset $\{x_i, y_i\}_{i=1}^n$ of n points on a 2D plane, in the case when both dependent variable x_i and regressor y_i are measured with errors, a linear regression model takes the form:

$$x_i - e_{x_i} = a_0 + a_1 \bar{y}_i \quad (2.6)$$

³The word "inconsistencies" indicates the unobserved random errors, also called as measurement errors.

2 Methodologies

where the regression coefficients a_0 and a_1 are the unknown parameters to be estimated, \bar{y}_i denotes the true but unobserved regressor, and the error variable e_{x_i} is an unobserved random variable that adds noise to the linear relationship between the dependent variable x and true regressor \bar{y}_i . Whereas the true regressor \bar{y}_i is observed with an error e_{y_i} in the pseudo observation equation:

$$y_i - e_{y_i} = \bar{y}_i \quad (2.7)$$

Such models, as the combination of (2.6) and (2.7), that account the measurement errors in both dependent variable and regressor, are errors-in-variables models. Further more, for the case of equal error variances, i.e. when $\delta = \frac{\sigma_{e_x}}{\sigma_{e_y}} = 1$, it is a orthogonal regression model which minimizes the perpendicular distances from the data points to the regression line.

2.3.2 Orthogonal Regression in Two-point Form

The two-point form of a infinite line in the Cartesian plane passing through the points (x_1, y_1) and (x_2, y_2) is given by:

$$(x - x_1) = \frac{(x_2 - x_1)}{(y_2 - y_1)} \times y - y_1 \quad (2.8)$$

with $y_2 \neq y_1$, where (x, y) is any point on the line.

Let the unknown coordinates of two different points on a line in 2D space be (x_1, y_1) and (x_2, y_2) and the observed 2D points be $\{x_i, y_i\}_{i=1}^n$ with measurement errors e_{x_i} and e_{y_i} in both variables. The orthogonal regression model in two-point form is:

$$\begin{aligned} x_i - e_{x_i} &= (x_1 - \frac{(x_2 - x_1)}{(y_2 - y_1)} \times y_1) + \frac{(x_2 - x_1)}{(y_2 - y_1)} \times \bar{y}_i \\ y_i - e_{y_i} &= \bar{y}_i \end{aligned} \quad (2.7 \text{ revisited})$$

To express (2.9) and (2.7) shortly, a function \mathcal{F} is defined as

$$\hat{\mathbf{p}} = \mathcal{F}(\mathbf{p}_s, \mathbf{p}_e, y) \quad (2.10)$$

which takes 2D coordinates of a start-point $\mathbf{p}_s(x_s, y_s)$ and an end-point $\mathbf{p}_e(x_e, y_e)$ that define an infinite line, and takes the measured y-coordinate y of an image point $\mathbf{p}(x, y)$, and returns the estimated image coordinates $\hat{\mathbf{p}}(\hat{x}, \hat{y})$ which lies on the infinite line $\overline{\mathbf{p}_s \mathbf{p}_e}$.

Note that as a combination of (2.9) and (2.7), function \mathcal{F} is actually composed of

$$\begin{aligned} \hat{x} &= \mathcal{F}^x(\mathbf{p}_s, \mathbf{p}_e, y) \\ \hat{y} &= \mathcal{F}^y(\mathbf{p}_s, \mathbf{p}_e, y) \end{aligned} \quad (2.11)$$

2.4 3D Line Reconstruction with Nonlinear LS Adjustment

In this section, I describe the process of refining the position of a 3D line segment in the object space so that its back-projection in each image has a best-fit to the extracted line in the image space, as illustrated in Figure 2.4.

The non-linear LS adjustment model with constraints is introduced in Section 2.4.1. The observation equations for LS adjustment are set up in Section 2.4.2. They describe the fitting of a straight line to the measurements in all covering images, where the fitting lines on different images are transformed from a single 3D straight line segment through the extended collinearity equation (2.4).

Regarding the fact that the collinearity is a point-wise condition, a line segment is represented by its two endpoints whose object coordinates are the six unknown parameters in the LS model. Correspondingly, the observation equations are line equations in two-point form. A line equation has however a mathematical meaning of infinite length. Therefore, some constraints on unknowns are necessary to avoid arbitrary locations of the two points on the infinite reconstructed 3D line. The constraint equations are modeled in Section 2.4.3.

In Section 2.4.4 the non-linear model is linearized and LS estimated.

To simplify the problem, a long lane-marking segment is partially reconstructed through a sliding window in the object space. Each segment is approximated by a straight line, taking into account the maximum curvature of the highway.

In each sliding window, a segment is reconstructed, i.e. a complete non-linear LS adjustment is performed. Only the middle point of the reconstructed line segment is recorded. The sliding window then moves a stepsize forward, and the process of 3D reconstruction is performed again starting from the recorded middle point of the previous line segment. Another line segment is then reconstructed, with its middle point being recorded, and so on. These recorded middle points are in the end the nodes of the reconstructed line. This process is illustrated in Figure 2.5.

The measurements for each reconstruction process are collected correspondingly, as shown in Figure 2.6 —by back-projecting the initial line segment into image space and buffering 10 pixels width on each side, only the extracted 2D line segment in this region is collected. As shown in Figure 2.7, the reconsideration in overlapping region of successive sliding windows makes the reconstruction more robust.

2 Methodologies

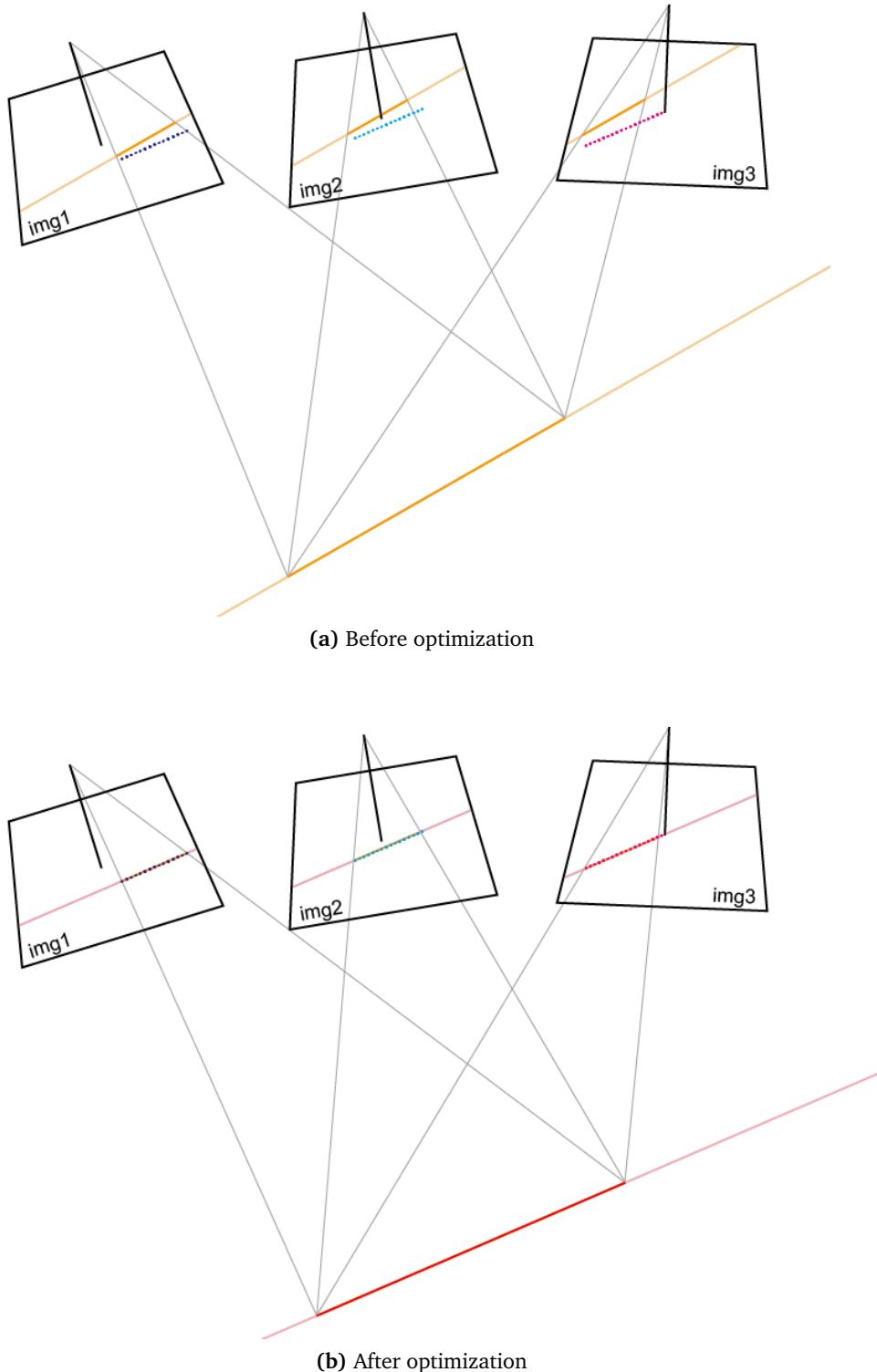


Figure 2.4: Before reconstruction, the back projection of initial approximate 3D line segment does not best-fit the extracted 2D lines in the covering images. After optimization, the back-projection of the reconstructed line segment should best fitting the extracted 2D lines in all the covering images.



(a) The first line segment of "sliding window length" is reconstructed, with its starting point and its middle point of "step size" from the starting point being recorded.



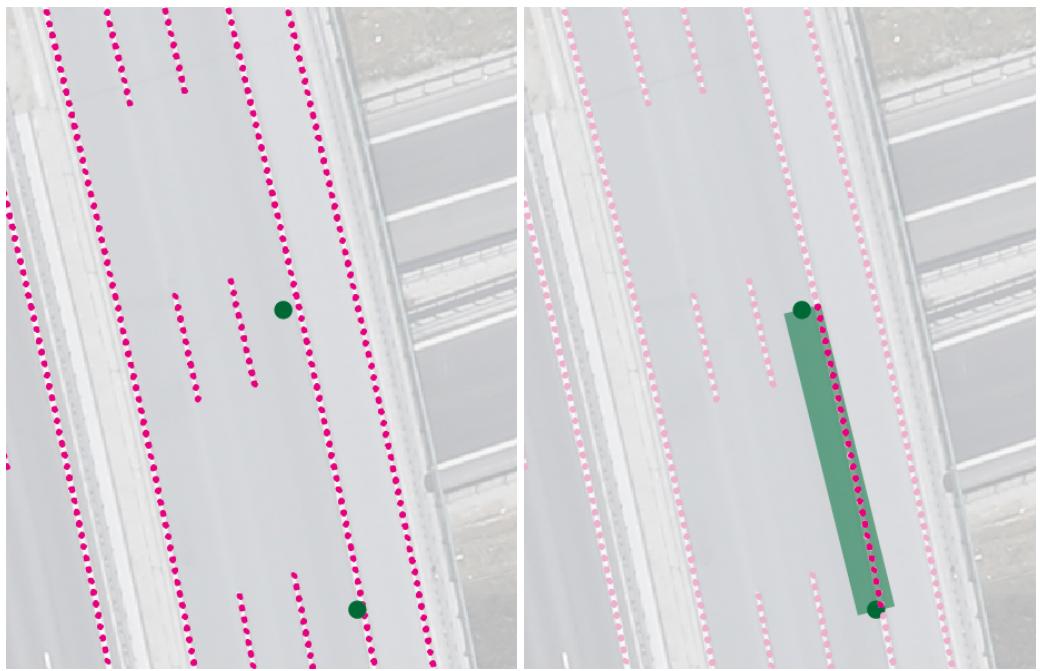
(b) Starting from the recorded node of last process, another line segment of "sliding window length" is again to be reconstructed. i.e. the sliding window has moved "step size" forward.



(c) The point of "step size" length from the starting point on the reconstructed segment is recorded.

Figure 2.5: A line is partially reconstructed through a sliding window.

2 Methodologies



(a) The pink points represent all the extracted lines (in the form of sets of points). The green points are the endpoints of the back projected initial approximate line segment.
(b) The points in the buffering area are collected as the measurements for LS adjustment.

Figure 2.6: Measurements collection in image space.



Figure 2.7: The red circle points out the reconsidered measurements in successive sliding windows.

2.4.1 The Nonlinear Gauss-Markov Model with Constraints

Given are the N observations $\mathbf{l} = [l_n]$, $n = 1, 2, \dots, N$, from which the U unknown parameters $\mathbf{x} = [x_u]$, $u = 1, 2, \dots, U$ are to be determined, with generally $U \leq N$.

The nonlinear Gauss-Markov model with N nonlinear functions $\mathbf{f}(\mathbf{x}) = [f_n(\mathbf{x})]$, $u = 1, 2, \dots, U$ and the H nonlinear constraints $\mathbf{h}(\hat{\mathbf{x}}) = [h_\eta(\hat{\mathbf{x}})]$, $\eta = 1, 2, \dots, H$ ($H < U$) between the unknowns can be written as:

$$\mathbf{l} + \hat{\mathbf{v}} = \mathbf{f}(\hat{\mathbf{x}}) \quad \text{or} \quad \hat{\mathbf{l}} = \underset{N \times 1}{\mathbf{f}}(\hat{\mathbf{x}}) \quad (2.12)$$

$$\underset{H \times 1}{\mathbf{h}}(\hat{\mathbf{x}}) = \mathbf{0} \quad (2.13)$$

where the observations \mathbf{l} are explicit functions of the unknowns \mathbf{x} , with the additive residuals \mathbf{v} introduced to the observations \mathbf{l} to achieve consistency.

Assuming that the deviations between the observed values \mathbf{l} and the true values $\hat{\mathbf{l}}$ are of random nature and have normal (or Gaussian) distribution, the uncertain observations \mathbf{l} are modeled with first and second moments:

$$\mathbf{l} \sim \mathcal{N}(\mathbf{f}(\hat{\mathbf{x}}), \Sigma_{ll}) \quad (2.14)$$

where Σ_{ll} is the variance-covariance matrix of observations \mathbf{l} , i.e. the observational errors.

The task is to minimize the weighted sum of residuals:

$$\Omega = \frac{1}{2}(\mathbf{l} - \mathbf{f}(\mathbf{x}))^T(\mathbf{l} - \mathbf{f}(\mathbf{x})) \quad \text{such that} \quad \mathbf{h}(\mathbf{x}) = \mathbf{0} \quad (2.15)$$

with the assumption of equal-weighted observations.

2.4.2 Observation Equations

Given a start-point $\mathbf{P}_s(X_s, Y_s, Z_s)$ and an end-point $\mathbf{P}_e(X_e, Y_e, Z_e)$ of a line segment L in the object space and the camera parameters \mathbf{q}^j of camera j . Consider the case where there are J images covering this line segment. With the expanded collinearity model (2.4), the start- and end-points of this line segment's back-projection in image j have the image coordinates $\mathbf{p}_s^j(x_s^j, y_s^j)$ and $\mathbf{p}_e^j(x_e^j, y_e^j)$:

$$\begin{aligned} \mathbf{p}_s^j &= \mathcal{G}(\mathbf{q}^j, \mathbf{P}_s) \\ \mathbf{p}_e^j &= \mathcal{G}(\mathbf{q}^j, \mathbf{P}_e) \end{aligned} \quad \forall j = 1, 2, \dots, J \quad (2.16)$$

Let l^j be the corresponding line segment of L being extracted (observed) on image j . Given a dataset $\{x_{l,i}^j, y_{l,i}^j\}_{i=1}^{N_l^j}$ of N_l^j points on line segment l^j , their estimated image coordinates $\hat{\mathbf{p}}_{l,i}^j(\hat{x}_{l,i}^j, \hat{y}_{l,i}^j)$ on the infinite line $\overline{\mathbf{p}_s^j, \mathbf{p}_e^j}$ computed from the orthogonal regression model (2.20) are:

$$\hat{\mathbf{p}}_{l,i}^j = \mathcal{F}(\mathbf{p}_s^j, \mathbf{p}_e^j, y_{l,i}^j) \quad \forall i = 1, 2, \dots, N_l^j \quad (2.17)$$

2 Methodologies

Combining (2.16) with (2.17) gives function \mathcal{H} :

$$\begin{aligned}\hat{\mathbf{p}}_{l,i}^j &= \mathcal{F}(\mathcal{G}(\mathbf{q}^j, \mathbf{P}_s), \mathcal{G}(\mathbf{q}^j, \mathbf{P}_e), y_{l,i}^j) \\ &= \mathcal{H}(\mathbf{q}^j, \mathbf{P}_s, \mathbf{P}_e, y_{l,i}^j) \quad \forall i = 1, 2, \dots, N_l^j, \quad \forall j = 1, 2, \dots, J\end{aligned}\tag{2.18}$$

which takes camera parameters $\mathbf{q}^j(x_0, y_0, c, X_0, Y_0, Z_0, R_{11}, \dots, R_{33}, A_1, A_2, B_1, B_2, C_1, C_2)$, object coordinates of \mathbf{P}_s and \mathbf{P}_e which define a line $\overline{\mathbf{P}_s, \mathbf{P}_e}$, and the observed y-coordinate of the point $\mathbf{p}_{l,i}^j$ in image space, and returns the estimated image coordinates $\hat{\mathbf{p}}_{l,i}^j$ on the back projected line of $\overline{\mathbf{P}_s, \mathbf{P}_e}$.

Corresponding to Equation (2.11), function \mathcal{H} is composed of

$$\begin{aligned}\hat{x}_{l,i}^j &= \mathcal{H}^x(\mathbf{q}^j, \mathbf{P}_s, \mathbf{P}_e, y_{l,i}^j) \quad \forall i = 1, 2, \dots, N_l^j, \quad \forall j = 1, 2, \dots, J \\ \hat{y}_{l,i}^j &= \mathcal{H}^y(\mathbf{q}^j, \mathbf{P}_s, \mathbf{P}_e, y_{l,i}^j)\end{aligned}\tag{2.19}$$

Equation (2.19)

Since the adjustment will be done "segment-wise"— for a pair of P_s and P_e , the measurements will be collected correspondingly. Thus the subscription l representing specific line segment will be left out in the followings.

2.4 3D Line Reconstruction with Nonlinear LS Adjustment

Each image gives $2 \times N^j$ observation equations⁴. These equations are often stacked together and written in vector form as:

$$\begin{bmatrix} x_1^j \\ x_2^j \\ \vdots \\ x_{N^j}^j \\ y_1^j \\ y_2^j \\ \vdots \\ y_{N^j}^j \end{bmatrix} \doteq \begin{bmatrix} \mathcal{H}^x(\mathbf{q}^j, \hat{\mathbf{P}}_s, \hat{\mathbf{P}}_e, y_1^j) \\ \mathcal{H}^x(\mathbf{q}^j, \hat{\mathbf{P}}_s, \hat{\mathbf{P}}_e, y_2^j) \\ \vdots \\ \mathcal{H}^x(\mathbf{q}^j, \hat{\mathbf{P}}_s, \hat{\mathbf{P}}_e, y_{N^j}^j) \\ \mathcal{H}^y(\mathbf{q}^j, \hat{\mathbf{P}}_s, \hat{\mathbf{P}}_e, y_1^j) \\ \mathcal{H}^y(\mathbf{q}^j, \hat{\mathbf{P}}_s, \hat{\mathbf{P}}_e, y_2^j) \\ \vdots \\ \mathcal{H}^y(\mathbf{q}^j, \hat{\mathbf{P}}_s, \hat{\mathbf{P}}_e, y_{N^j}^j) \end{bmatrix} \left\{ \begin{array}{l} N^j \\ N^j \end{array} \right\} \quad (2.20)$$

For all covering image $j = 1, 2, \dots, J$, there are $2 \times \sum_{j=1}^J N^j$ observation equations. Being written in the structure of the Gauss-Markov model, corresponding to Equation (2.12), they are expressed as:

$$l + \hat{v} = f(\hat{x}) : \begin{bmatrix} x_1^1 \\ \vdots \\ x_{N^1}^1 \\ y_1^1 \\ \vdots \\ y_{N^1}^1 \\ \vdots \\ y_{N^J}^J \\ \vdots \\ x_{N^J}^J \\ \vdots \\ y_1^J \\ \vdots \\ y_{N^J}^J \end{bmatrix} + \begin{bmatrix} \hat{v}_{x_1^1} \\ \vdots \\ \hat{v}_{x_{N^1}^1} \\ \hat{v}_{y_1^1} \\ \vdots \\ \hat{v}_{y_{N^1}^1} \\ \vdots \\ \hat{v}_{y_{N^J}^J} \\ \vdots \\ \hat{v}_{x_1^J} \\ \vdots \\ \hat{v}_{x_{N^J}^J} \\ \vdots \\ \hat{v}_{y_1^J} \\ \vdots \\ \hat{v}_{y_{N^J}^J} \end{bmatrix} = \begin{bmatrix} \mathcal{H}^x(\mathbf{q}^1, \hat{\mathbf{P}}_s, \hat{\mathbf{P}}_e, y_1^1) \\ \vdots \\ \mathcal{H}^x(\mathbf{q}^1, \hat{\mathbf{P}}_s, \hat{\mathbf{P}}_e, y_{N^1}^1) \\ \mathcal{H}^y(\mathbf{q}^1, \hat{\mathbf{P}}_s, \hat{\mathbf{P}}_e, y_1^1) \\ \vdots \\ \mathcal{H}^y(\mathbf{q}^1, \hat{\mathbf{P}}_s, \hat{\mathbf{P}}_e, y_{N^1}^1) \\ \vdots \\ \vdots \\ \vdots \\ \mathcal{H}^x(\mathbf{q}^J, \hat{\mathbf{P}}_s, \hat{\mathbf{P}}_e, y_1^J) \\ \vdots \\ \mathcal{H}^x(\mathbf{q}^J, \hat{\mathbf{P}}_s, \hat{\mathbf{P}}_e, y_{N^J}^J) \\ \mathcal{H}^y(\mathbf{q}^J, \hat{\mathbf{P}}_s, \hat{\mathbf{P}}_e, y_1^J) \\ \vdots \\ \vdots \\ \mathcal{H}^y(\mathbf{q}^J, \hat{\mathbf{P}}_s, \hat{\mathbf{P}}_e, y_{N^J}^J) \end{bmatrix} \left\{ \begin{array}{l} 2 \times N^1 \\ \vdots \\ 2 \times N^J \end{array} \right\} \quad (2.21)$$

⁴Dot equal indicates inconsistencies between the measured values, x_i^j and \hat{x}_i^j , and the computed values, $\mathcal{H}^x(q^j, P_s, P_e, y_i^j)$ and $\mathcal{H}^y(q^j, P_s, P_e, y_i^j)$.

2 Methodologies

2.4.3 Constraint Equations

There are three constraints on the unknown parameters used in this work:

- Fixing the X-, Y-coordinates of the start-point using the approximate values:

$$\hat{X}_s - X_s^0 = 0 \quad (2.22)$$

$$\hat{Y}_s - Y_s^0 = 0 \quad (2.23)$$

- Fixing the length of the line segment (i.e. constraining the relative location of the end-point):

$$\sqrt{(\hat{X}_s - \hat{X}_e)^2 + (\hat{Y}_s - \hat{Y}_e)^2 + (\hat{Z}_s - \hat{Z}_e)^2} - S = 0 \quad (2.24)$$

Only in the first line segment reconstruction of a long lane marking, the fixed X_s and Y_s values are from the initial parameter estimates derived in Section 2.5. Starting from the second line segment, the fixed values X_s^0 and Y_s^0 depend on the previously determined values.

The constraint equations (2.22), (2.23) and (2.24) can be stacked together and written in the structure of the Gauss-Markov model with constraints, corresponding to Equation (2.13):

$$h(\hat{x}) = \mathbf{0} : \begin{bmatrix} \hat{X}_s - X_s^0 \\ \hat{Y}_s - Y_s^0 \\ \sqrt{(\hat{X}_s - \hat{X}_e)^2 + (\hat{Y}_s - \hat{Y}_e)^2 + (\hat{Z}_s - \hat{Z}_e)^2} - S \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad (2.25)$$

2.4.4 Least-Squares Estimation of the non-linear Gauss-Markov Model with Constraints

2.4 3D Line Reconstruction with Nonlinear LS Adjustment

The nonlinear equation system is approximated to be locally linear with small step size of the unknown quantities. The linearized form is expressed as:

$$\widehat{\Delta l} = \Delta l + \widehat{v} = \underset{N \times U}{A} \widehat{\Delta x} \quad (2.26)$$

$$c_h = \underset{H \times U}{H^T} \widehat{\Delta x} \quad (2.27)$$

where

the $N \times U$ design matrix, which is the Jacobian of the function evaluated at the approximate values of the unknown parameters, is

$$A = \left. \frac{\partial f(x)}{\partial x} \right|_{x=\widehat{x}^a}$$

the $U \times H$ constraint matrix, which is the Jacobian of the constraints, is

$$H = \left. \left(\frac{\partial h(x)}{\partial x} \right)^T \right|_{x=\widehat{x}^a}$$

and the residual constraints are

$$c_h = -h(\widehat{x}^a)$$

with the corrections

$$\Delta l = l - f(\widehat{x}^a) =: \widehat{v}^a \quad (2.28)$$

$$\widehat{\Delta x} = \widehat{x} - \widehat{x}^a \quad (2.29)$$

where \widehat{x}^a is the approximate values for the estimates of the unknown parameters.

In the linearized substitute model as shown in (2.26) and (2.27), it turns to solve for the increments of unknowns Δx instead of the unknowns themselves. The LS estimation is applied iteratively until convergence is achieved.

The unknown parameters $\widehat{\Delta x}$ of the linearized model can be determined from the extended normal equation system

$$\begin{bmatrix} A^T A & H \\ H^T & 0 \end{bmatrix} \begin{bmatrix} \widehat{\Delta x} \\ \lambda \end{bmatrix} = \begin{bmatrix} A^T \Delta l \\ c_h \end{bmatrix} \quad (2.30)$$

The matrix A does not need to have full rank, but the block matrix $[A^T, H]$ must have full rank in order to guarantee the estimation problem has a unique solution.

The redundancy R of the problem is

$$R = N + H - U \quad (2.31)$$

Since the estimated variance-covariance matrix of the estimated parameters $\hat{\Sigma}_{\hat{X}\hat{X}}$ depends on both the design matrix A (i.e. the configuration) and the posterior standard deviation $\hat{\sigma}_0$ (i.e. the posterior measurements quality):

$$\hat{\Sigma}_{\hat{X}\hat{X}} = \hat{\sigma}_0^2 (A^T A)^{-1} = \begin{bmatrix} \hat{\sigma}_{\hat{X}}^2 & \hat{\sigma}_{\hat{X}\hat{Y}} & \hat{\sigma}_{\hat{X}\hat{Z}} \\ \hat{\sigma}_{\hat{Y}\hat{X}} & \hat{\sigma}_{\hat{Y}}^2 & \hat{\sigma}_{\hat{Y}\hat{Z}} \\ \hat{\sigma}_{\hat{Z}\hat{X}} & \hat{\sigma}_{\hat{Z}\hat{Y}} & \hat{\sigma}_{\hat{Z}}^2 \end{bmatrix} \quad (2.32)$$

2 Methodologies

we are not able to tell from the variances $\hat{\sigma}_{\hat{X}}, \hat{\sigma}_{\hat{Y}}, \hat{\sigma}_{\hat{Z}}$ whether the .

By setting a constant priori standard deviation value in all the LS adjustment processes (i.e. assuming the measurements are of same quality in each segment), the priori variance-covariance matrix of the estimated parameters $\Sigma_{\hat{X}\hat{X}}$ reflects the quality of the design matrix (i.e. the configuration strength) in each LS adjustment processes.

$$\Sigma_{\hat{X}\hat{X}} = \sigma_0^2 (A^T A)^{-1} \quad (2.33)$$

2.5 Line Projection on the DSM (Determination of Initial Parameter Estimates)

As the equation system in Section 2.4 may exhibit multiple local minimum, a "correct" initial approximation of the unknowns is required for convergence to the correct solution. To provide such initial 3D line segment, the extracted line features derived in Section 2.1 can be projected onto DSM based on the bundle adjusted exterior and interior orientations.

Given image coordinates $\mathbf{p}(x, y)$ of a point and (bundle-adjusted) image orientations \mathbf{q} , there is still one degree of freedom in extended collinearity equation (2.5) on solving object coordinates $\mathbf{P}(X, Y, Z)$. Combined with the usage of DSM, which provides the height information Z given a position (X, Y) , the corresponding object coordinates can be solved iteratively until the increment ΔZ small enough, i.e. convergence achieved.

Considering that the DSM is raster (discrete) whereas X and Y have continuous numerical values, the DSM height is bilinear interpolated during the iterative process.

3 Experimental Results and Evaluations

Section 3.1 provides information about input dataset and the applied preprocessing steps.

In Section 3.3, the correctness of the derived LS model for reconstruction is evaluated. Some other properties of the proposed reconstruction approach are also discussed.

3.1 Materials

Aerial Images For real-time mapping applications during disasters, mass events and traffic monitoring scenarios, the German Aerospace Center (DLR) has developed a new optical sensor system— the 4k system— on a helicopter from DLR. The oblique aerial images used in this work are acquired from a Canon EOS 1D-X camera, one of the three non-metric cameras in the 4k system, with an oblique viewing angle τ of 15°. The image data sets used in this work were acquired in around 500m flying height $H_{fliheight}$ above ground, which leads to a GSD of 7cm.

An example aerial image is shown in Figure 3.5. Table 3.1 lists the properties of this camera, and Table 3.2 provides the viewing geometry information.

The images used in this work are acquired with a special flight configuration at both sides of the motorway which guarantees a continuous stereo view perpendicular to the lane marking direction.¹ This is realized by flying at the right-hand side with respect to flying direction along the motorway, with the left oblique camera looking left-down to the motorway, in both forward and backward trip. The flight configuration is shown in Figure 3.1 on the Google Earth platform.

Besides, the forward overlap is around 70%, and all the lane markings are covered by both strips, whereas the side overlap depends on the distance of flight strips, which are a result of the pilots navigation ability and other influences, like wind. Nevertheless, the motorway in its entire width was covered by the two flight strips. Altogether, this results in approximately 8-image coverage in road areas.

¹The classical photogrammetric approach on flight planning is to have several straight flight lines which cover the whole motorway in a stereo view. This would be possible in this project yet would require more flight costs and would produce many more images.

3 Experimental Results and Evaluations

Canon EOS 1D-X	
Lenses	Zeiss Makro Planar 2/50
Sensor / Pixel size	Full frame CMOS / $6.944 \mu\text{m}$
Image size	5184×3456 pixel, ratio 3:2 (17.9 MPix)
ISO	100–204800
max. frame rate / max. images	14 fps/ 180 images
Exposure time	30 s – 1/8000 s
Data interface	LAN (EDSDK software interface)

Table 3.1: Properties of the oblique camera

RGB, 50mm lens	
Viewing directions	$\pm 15^\circ$
FOV	$\pm 34^\circ$ across strip, $\pm 13^\circ$ along strip
Coverage @500m	780 m \times 230 m
GSD @500m	6.9 cm (nadir)

Table 3.2: Viewing geometry



Figure 3.1: Flight trajectory of DLR helicopter visualized on the Google Earth platform. The green polyline shows the flight trajectory. *Source: Google Earth 04/01/2017*

Exterior and Interior Orientations The Exterior Orientation (EO) of the images are directly measured by a GNSS/Inertial system IGI IId. The EO parameters are then refined by a self-calibrating bundle adjustment. The accuracies of the exterior orientation (EO) parameters are shown in Table 3.3. The calibrated Interior Orientation (IO) parameters and their accuracies are shown in Table 3.4. To provide an overall quality on the interior orientations: from the calibration result of interior orientations (involving lens distortion), the residuals appear non-systematic and the biggest residual $r_{max,IO}$ is around 1 pixel.

To judge the influence of exterior and interior parameters on positioning accuracy in object space, the maximum values for each component based on the flight configuration was calculated. The quality of interior and exterior orientation parameters set would have a maximum impact in object space for around 16.5 [cm] in X,Y-direction:

- caused by inaccurate camera position:

$$\sqrt{\sigma_{north}^2 + \sigma_{east}^2} = \sqrt{0.055^2 + 0.035^2} \approx 0.065 \text{ [meter]}$$

- caused by inaccurate camera attitude:

$$\begin{aligned} & \tan(\sqrt{\sigma_{roll}^2 + \sigma_{pitch}^2}) \times H_{flightheight} \times \frac{1}{\cos^2 \tau} \\ &= \tan(\sqrt{0.002^2 + 0.002^2}) \times 500 \times \frac{1}{\cos^2 15^\circ} \approx 0.026 \text{ [meter]} \end{aligned}$$

- caused by inaccurate Interior Orientations:

$$r_{max,IO} \times GSD \times \frac{1}{\cos^2 \tau} = 1 \times 0.069 \times \frac{1}{\cos^2 15^\circ} \approx 0.074 \text{ [meter]}$$

and around 9.4 [cm] in Z-direction:

- caused by inaccurate camera position:

$$\sigma_{altitude} \approx 0.069 \text{ [meter]}$$

- caused by inaccurate camera attitude:

$$\begin{aligned} & \tan(\sqrt{\sigma_{roll}^2 + \sigma_{pitch}^2}) \times H_{flightheight} \times \frac{\sin \tau}{\cos \tau} \\ &= \tan(\sqrt{0.002^2 + 0.002^2}) \times 500 \times \frac{\sin 15^\circ}{\cos 15^\circ} \approx 0.007 \text{ [meter]} \end{aligned}$$

- caused by inaccurate Interior Orientations:

$$r_{max,IO} \times GSD \times \frac{\sin \tau}{\cos \tau} = 1 \times 0.069 \times \frac{\sin 15^\circ}{\cos 15^\circ} \approx 0.018 \text{ [meter]}$$

The above information tells the positioning accuracy in object space with measurements on a single image. With corresponding measurements from multiple stereo views, which allows the intersection of multiple rays, the positioning accuracy is expected to be improved for being overdetermined.

3 Experimental Results and Evaluations

position accuracies [meter]	attitude accuracies [degree]
σ_{north} 0.055	σ_{Roll} 0.002
σ_{east} 0.035	σ_{Pitch} 0.002
$\sigma_{altitude}$ 0.069	σ_{Yaw} 0.005

Table 3.3: Accuracies of Exterior Orientations

Interior Orientations	accuracies	unit
focal length c 0.051	σ_c 6.9e-7	[meter]
x coordinate of principal point pp_x -42.259	σ_{pp_x} 0.167	[μm]
x coordinate of principal point pp_y 115.384	σ_{pp_y} 0.799	[μm]

Table 3.4: Interior Orientations and their accuracies

Digital Surface Model (DSM) High resolution DSM can be generated by laser scanning or dense image matching. Compared to laser scanning, applying dense image matching to produce DSM is of lower cost on data acquisition.

For each pair of stereo images, a disparity map is generated using SGM algorithm. With disparity's property of being inversely proportional to depth, the disparity maps can be used to derive the DSM. In most of the cases, a point in object space is covered by more than two aerial images, resulting in more than one disparity maps. This leads to ambiguities on height value decision during DSM generations. The height value of each DSM pixel is determined by simply taking the median value derived from disparity maps in odd number of disparity maps cases, and the value just below the median in even number cases. Such disparity maps fusion approach may result in systematic errors of having lower height value in some parts of DSM.

The asphalt road surfaces where the lane markings locate on is homogeneous. Since SGM is a kind of appearance-based matching algorithm, such lowly textured area may have unstable, very different candidating height values derived from different disparity maps. The systematic errors arisen from disparity map fusion and some other random noises happen especially obviously in such areas.

However, such high-resolution DSM gives a good starting point for the lane marking refinement. In other words, the DSM will be used only for setting up the initial values of the work flow, and will not influence the final results of the 3D lane marking reconstruction.

The DSM has 20 cm grid spacing. Figure 3.2 shows a part of the DSM. Standard deviations of the height value in this part of the DSM is shown in Figure 3.3. The number of stereo image pairs used for each part on the DSM is shown in Figure 3.4.

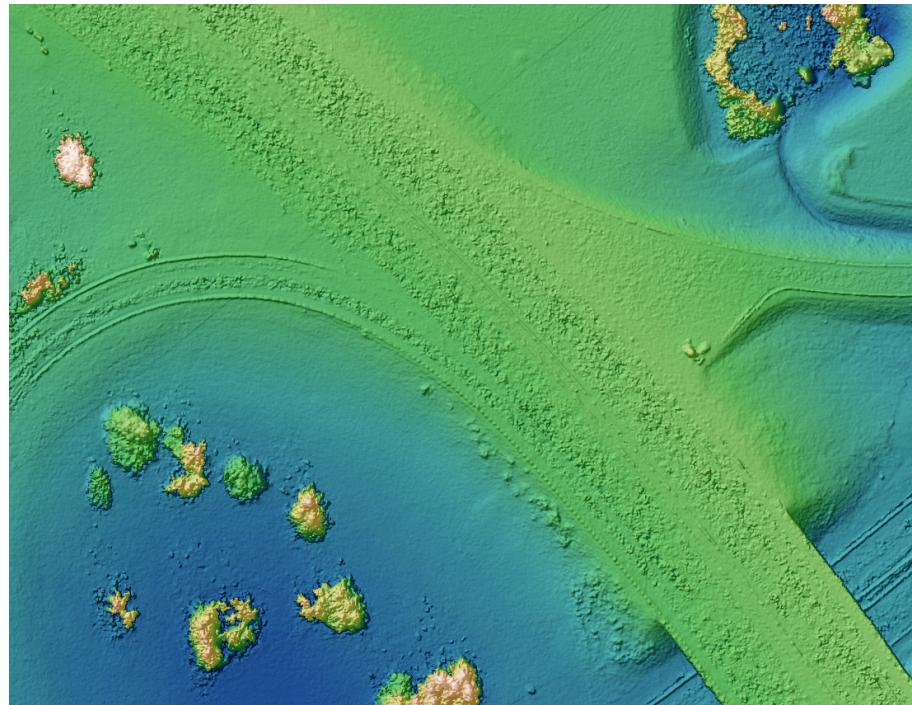


Figure 3.2: Part of the DSM in road area. It is noisy in the center of motorway.

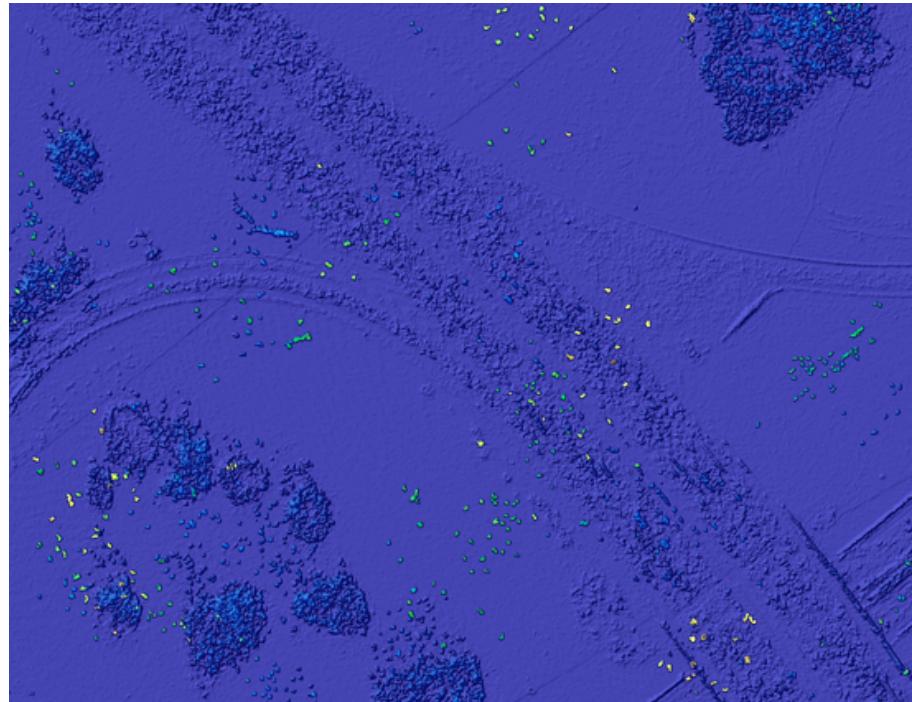


Figure 3.3: Standard deviations of the height value of the DSM in road area. It has higher value in the center of motorway.

3 Experimental Results and Evaluations

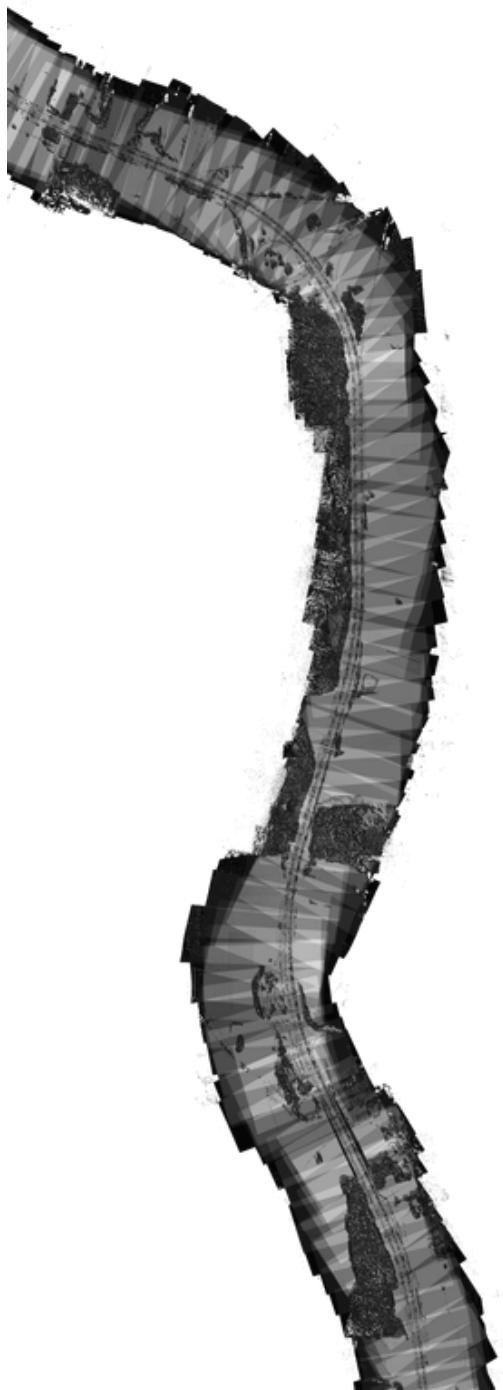


Figure 3.4: Distribution of stereo pairs used for DSM generation. Lighter color indicates more stereo pairs are used in that area. Maximum 27 stereo pairs are used for one pixel.

Orthorectified Images The orthorectified images are processed using the DSM and the interior and exterior orientations derived from the bundle adjustment. They are georeferenced and the scale is uniform. One of the orthorectified images is shown in Figure 3.6. The orthorectified images are only used for setting up initial values and used as intermediary step for processing the road masks, but do not influence the results of 3D lane marking reconstruction.

Road Masks Road segments are masked out from original images based on OpenStreetMap (OSM) data: Firstly, the rasterized road segments from OSM data are written with 25 meter buffer width around road axes into orthorectified images. By back-projecting the mask from orthorectified image to original image using the 3D information from the DSM, it can then be used to mask out the road regions on the original images, as shown in Figure 3.7.

3 Experimental Results and Evaluations



Figure 3.5: Original Image

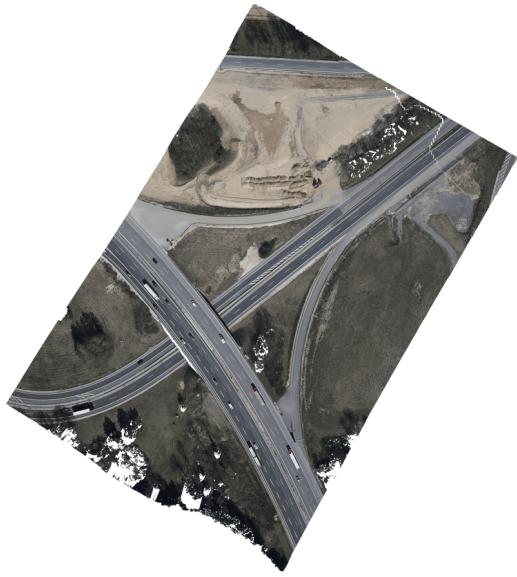


Figure 3.6: Orthorectified Image



Figure 3.7: Masked Image

3.2 Preprocessing

In lane marking extraction step, the σ value for Gaussian smoothing is set to be 1.8 to slightly suppress the noise in images. The extracted lines of length less than 70 pixels are rejected regarding the fact that a dashed lane-line is no longer than 6 meter which is correspondingly 87 pixels with GSD of 6.9 cm.

The length of the sliding window depends on the expected curvature, and for motorways it was fixed to 9m ...

3.3 Simulation

This section aims to verify the correctness of the derived LS model and to discover some characteristics of the reconstruction model using simulation data. The used materials are as described in Section 3.1. Only the measurements (the image coordinates of the extracted lines), the true and approximate values of the unknowns (the object coordinates of a line segment) in the non-linear LS model are simulated, as described in Section 3.3.1.

Section 3.3.2 firstly evaluate if the iteration scheme converges to the correct solution given imperfect initial values of the unknowns. The significant height differences between the approximate and the reconstructed line segments are also presented, indicating the refining ability of the proposed reconstruction approach.

The ability of the derived LS model on detecting the measurement errors is then evaluated. Furthermore, how the increase of covering images would influence the reconstruction result is elaborated.

3.3.1 Simulation Data

The true line segment in object space Firstly, the object coordinates of the endpoints of a 3D line segment are defined, with 151.8 meters length, locating on the road surface in the test area (German highway A9) with 3 to 7 aerial images coverage. By linear interpolating several points with 0.2 meter spaces (considering DSM grid of 0.2 meter) between the two endpoints, a 3D line segment in the form of a set of 3D points is generated. This 3D line segment serves as the ground truth in the experiments in Section 3.3.

The observed line segments in image spaces The observations in the LS model are simulated by back-projecting the true line segment into the covering images. Gaussian random noise $e \sim \mathcal{N}(0, 0.5^2)$ is added in the observations for each LS adjustment, as line extraction process is of sub-pixel accuracy. The added noise is plotted in Figure 3.8

3 Experimental Results and Evaluations

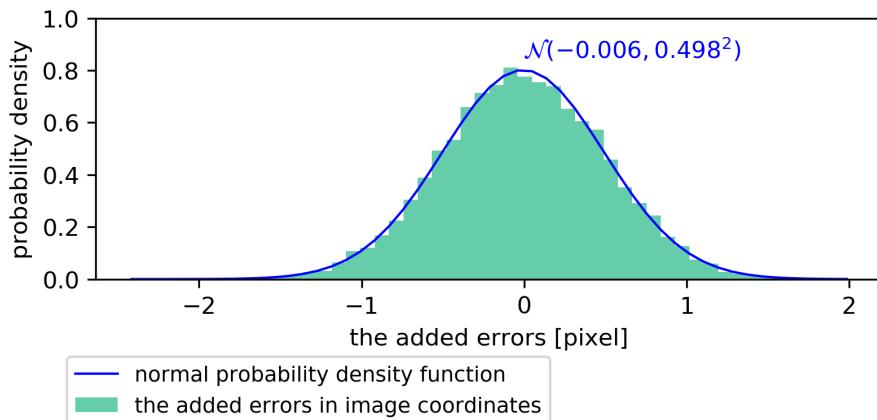


Figure 3.8: The added Gaussian random noise in the observations.

The approximate line segment in object space The initial estimates for non-linear LS adjustment is generated by projecting the observed line segments in image space onto the DSM.

3.3.2 Simulation Result

Figure 3.9 shows the reconstructed and the true line segments in UTM coordinate system (in Zone 32N). The distances from the reconstructed line nodes to the true line segment are computed and collected, resulting in sample size of 18. The sample mean is 0.008 [meter] and the sample variance is 0.101 [meter].

For such small sample size data, a two-tailed t-test is adopted to test if the population mean significantly not equals zero, i.e. **if the reconstructed line segments are significantly far from the true line segments**. Null hypothesis (H_0) and (two-tailed) alternative hypothesis (H_A) are stated as:

$$\begin{aligned} H_0 : \mu &= 0 \\ H_A : \mu &\neq 0 \end{aligned}$$

A significance level $\alpha = 0.05$ is selected, with degree of freedom being $18 - 1 = 17$, the two-tailed t-table value $T_{(0.975,17)}$ is

$$T_{(0.975,17)} = 2.110$$

which leads to the decision rule: if test statistic T_{obs} is less than $-T_{(0.975,17)} = -2.110$ or greater than $T_{(0.975,17)} = 2.110$, reject the null hypothesis.

With the sample mean $\bar{x} = 0.008$, the proposed population mean $\mu_0 = 0$, the sample standard deviation $\sigma = 0.101$, and sample size $n = 18$, the test statistic for One Sample T Test has the calculated value:

$$T_{obs} = \frac{\bar{x} - \mu_0}{\sigma/\sqrt{n}} = \frac{0.008 - 0}{0.101/\sqrt{18}} \approx 0.34$$

which is neither less than $-T_{(0.975,17)} = -2.110$ nor greater than $T_{(0.975,17)} = 2.110$, i.e. not in the rejection region. As a result, we fail to reject the null hypothesis. In other words, **we are not able to claim that the reconstructed line is significantly far away from the true line**. This indicates that the derived non-linear LS adjustment model for 3D reconstruction is correct.

Figure 3.10 shows the reconstructed line segments and the DSM profile which serves as the initial approximation for non-linear LS adjustment, in UTM coordinate system (in Zone 32N). The maximum distance between them is 1.97 meter, mainly in Z-direction. This tells that **the reconstruction model is at least able to refine the initial approximation with 2 meters bias in Z-direction**.

3 Experimental Results and Evaluations

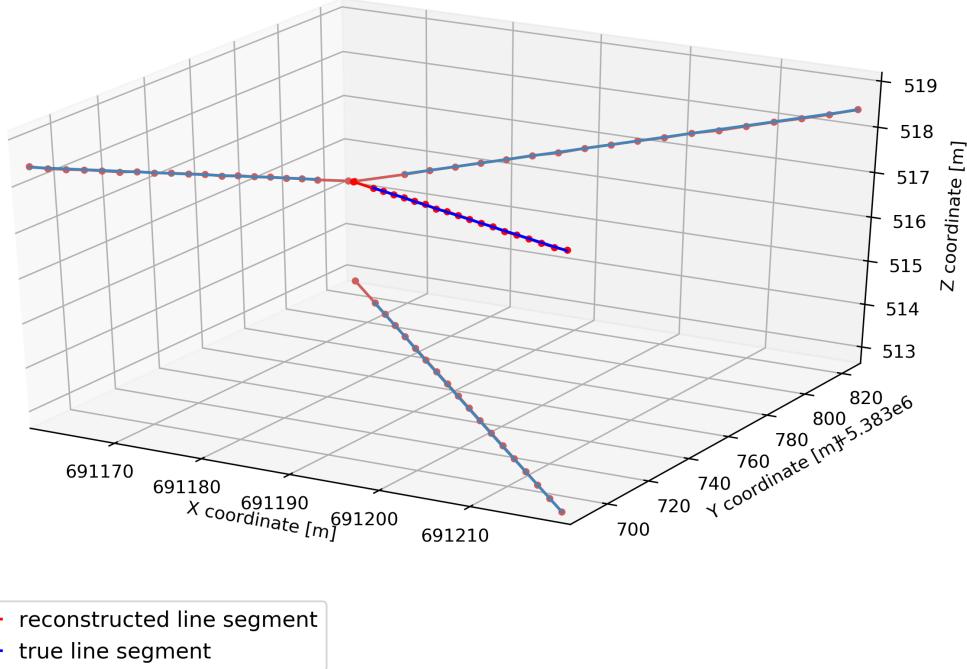


Figure 3.9: The reconstructed line segments and the true line segments in UTM coordinate system (in Zone 32N).

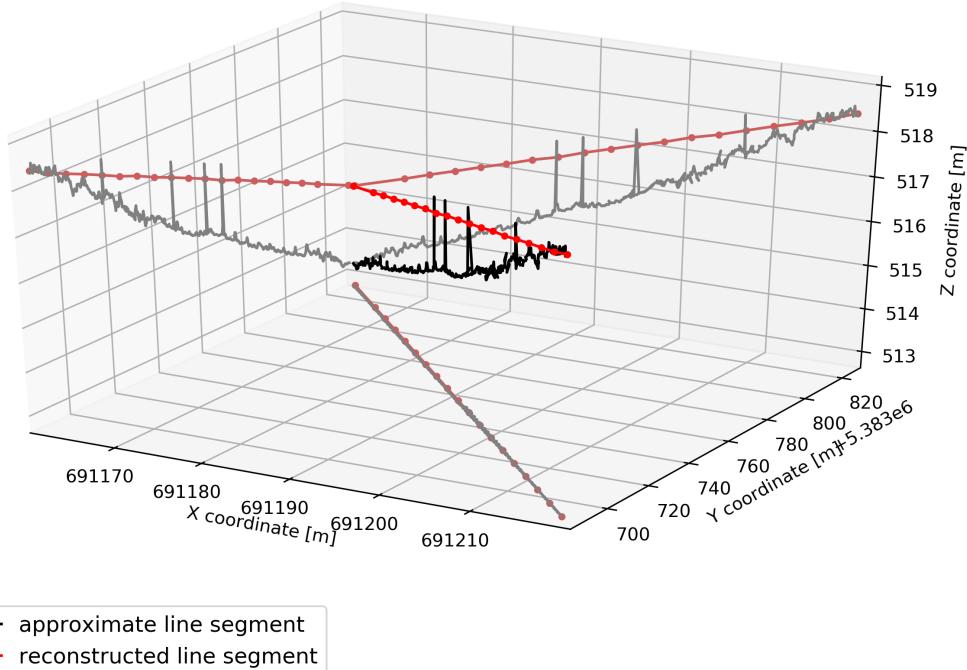


Figure 3.10: The reconstructed line segments and the unrefined DSM profile in UTM coordinate system (in Zone 32N).

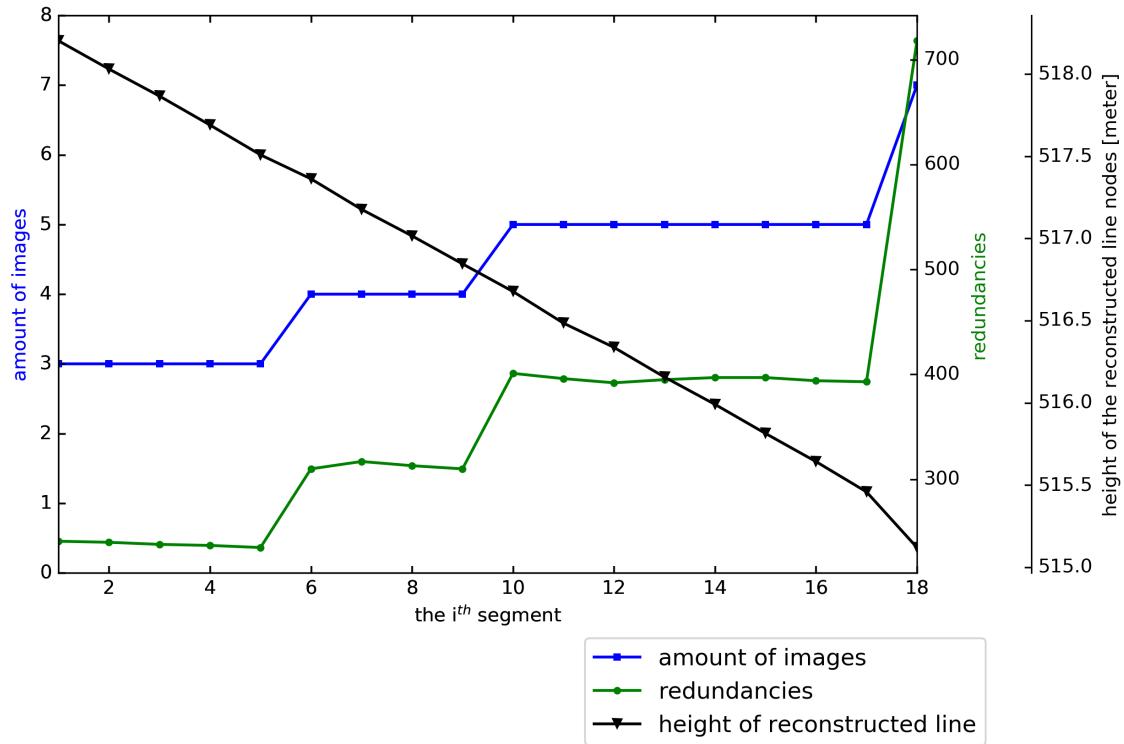


Figure 3.11: XXX.

Figure 3.11 gives the information on the amount of covering images, the redundancies and the height value of the reconstructed nodes, of each segment. Note that in each segment, LS adjustment is processed independently.

Figure 3.12 shows the mean and variance of the added random Gaussian noise and the adjusted residuals of each segment. This two samples are compared by applying a two-tailed two-sample T-test. Null hypothesis and alternative hypothesis (H_A) are stated as:

$$H_0 : \mu_1 - \mu_2 = 0$$

$$H_A : \mu_1 - \mu_2 \neq 0$$

With significance level $\alpha = 0.05$ and degree of freedom ≈ 300 , the t-score is

$$T_{0.975,300} = 1.968$$

As shown in Figure 3.13, all the test statistics T_{obs} of each segment are not less than $-T_{0.975,300} = -1.968$ or greater than $T_{0.975,300} = 1.968$, i.e. not in the rejection region, the null hypothesis could not be rejected. In other words, we fail to claim that the adjusted residuals are statistically different from the added random noise.

without the influence of imperfect camera parameters,

3 Experimental Results and Evaluations

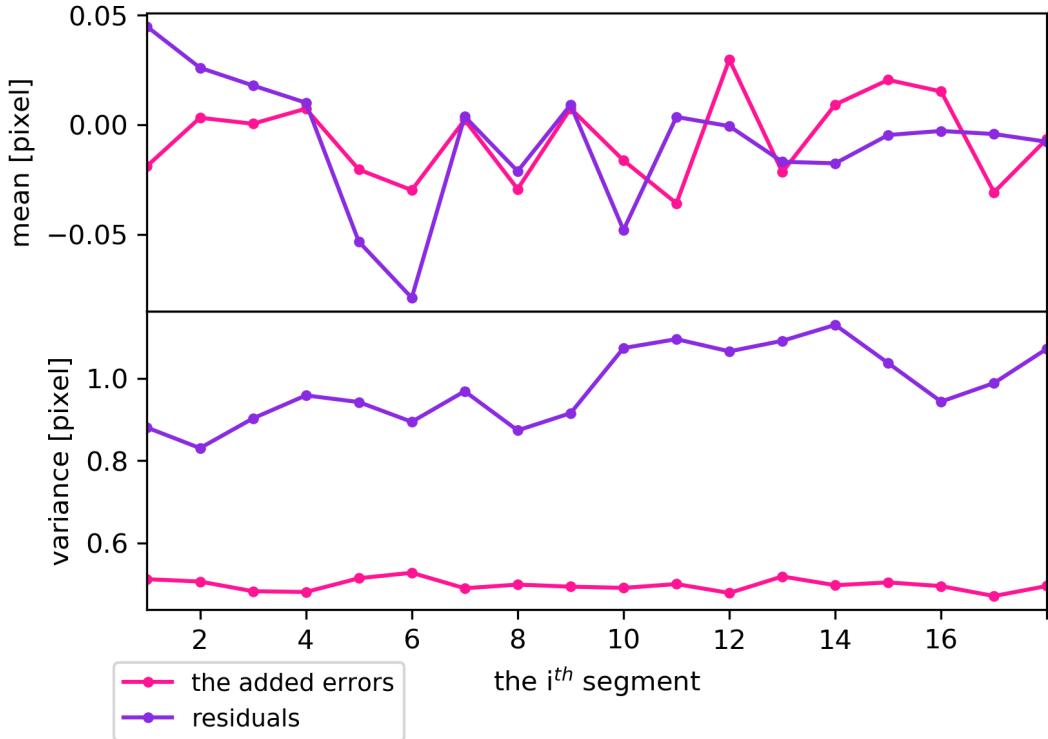


Figure 3.12: The relation between the added random Gaussian noise and the adjusted residuals.

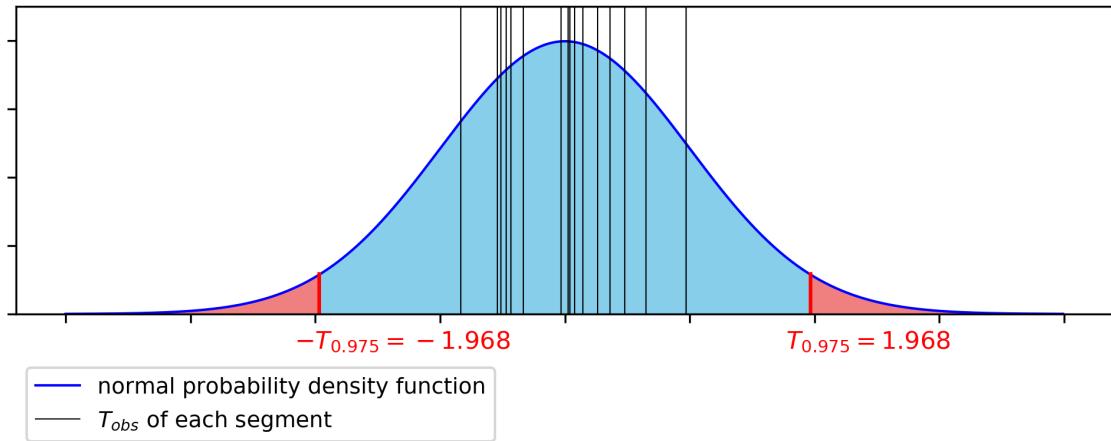


Figure 3.13: The red area under the probability density function is the rejection region, which is 5%. Since none of the T_{obs} falls in the rejection region, the null hypothesis could not be rejected.

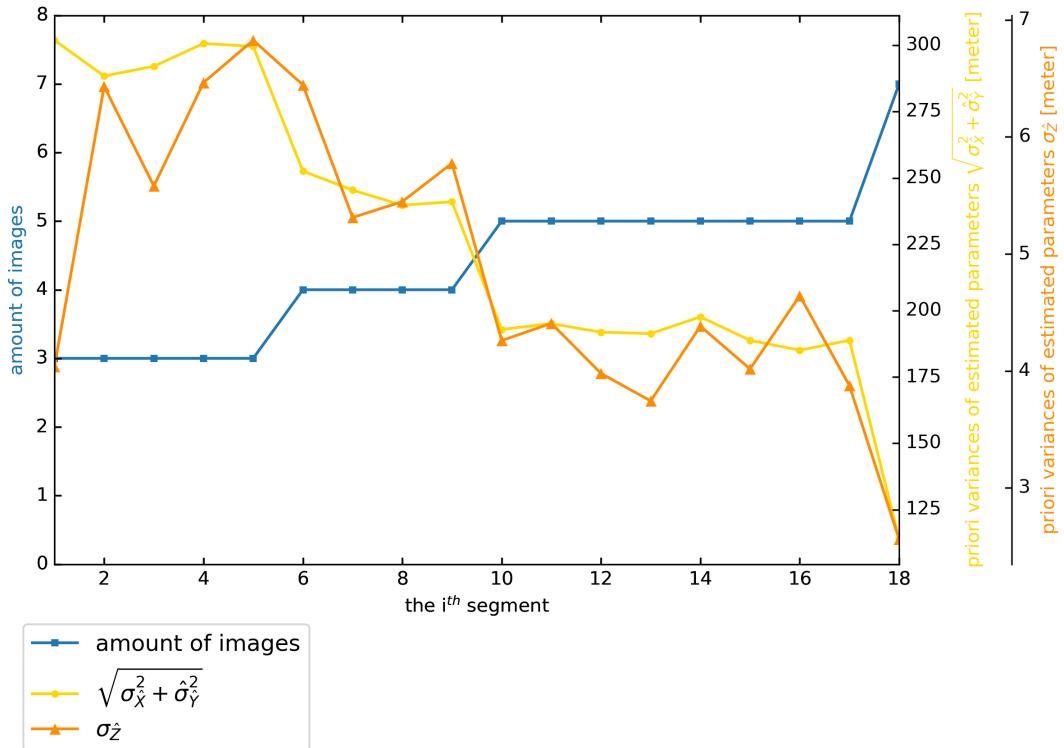


Figure 3.14: The relation between the variances of the estimated parameters and the amount of images.

From Figure 3.14 it can be seen that the estimated parameters generally have smaller prior variance values (in horizontal direction) $\sqrt{\sigma_X^2 + \hat{\sigma}_Y^2}$ and (in vertical direction) σ_Z with the increase of covering images, whereas larger deviations from the trend are contained in vertical direction. **By increasing the configuration strength, i.e. increasing the amount of covering images with different orientations, the precision of the estimated parameters can be improved.**

Besides, the prior variance of the estimated parameters is smaller in vertical direction σ_Z than in horizontal direction $\sqrt{\sigma_X^2 + \hat{\sigma}_Y^2}$. This tells that **the reconstructed nodes have higher precision in vertical direction.** [reason? sec 3.1]

3.4 True Data

3.4.1 True Data Result

A continuous lane marking of 255.9 meters length is reconstructed. Figure 3.15 shows the reconstructed line segments and the DSM profile in UTM coordinate system (in Zone 32N). The distances from the reconstructed line segment to the DSM profile are computed and plotted into histogram in Figure 3.16. They are collected along the reconstructed line segments with 0.2 meter spacing (considering the DSM grid of 0.2 meter), resulting in sample size of 1256. The sample mean is -0.180 [meter] and the sample standard deviation is 0.174 [meter].

Assuming DSM height profile being significantly lower than the reconstructed line segments for more than 17??? centimeters, a lower-tailed Z-test is adopted. Null hypothesis (H_0) and (one-tailed) alternative hypothesis (H_A) are stated as:

$$H_0 : \mu \geq -0.170$$

$$H_A : \mu < -0.170$$

A significance level $\alpha = 0.05$ is selected, i.e. the area in body is 0.950 out of 100%. The corresponding z-score is:

$$Z_{0.950} = 1.64$$

leads to the decision rule: if Z_{obs} is less than -1.64 , reject the null hypothesis.

With the sample mean $\bar{x} = -0.181$, the proposed population mean $\mu_0 = -0.170$, the sample standard deviation $\sigma = 0.174$, and sample size $n = 1256$, the test statistic for a One Sample Z Test has a calculated value:

$$Z_{obs} = \frac{\bar{x} - \mu_0}{\sigma/\sqrt{n}} = \frac{-0.181 - (-0.170)}{0.174/\sqrt{1256}} \approx -2.24$$

As the test statistic $Z_{obs} \approx -2.24$ is less than $-Z_{0.95} = -1.64$, i.e. in the rejection region, the null hypothesis is rejected. In other words, **with 95% confidence we can claim that the DSM profile is in average, statistically and significantly lower than the reconstructed line segments for at least 17 centimeters in this region.**

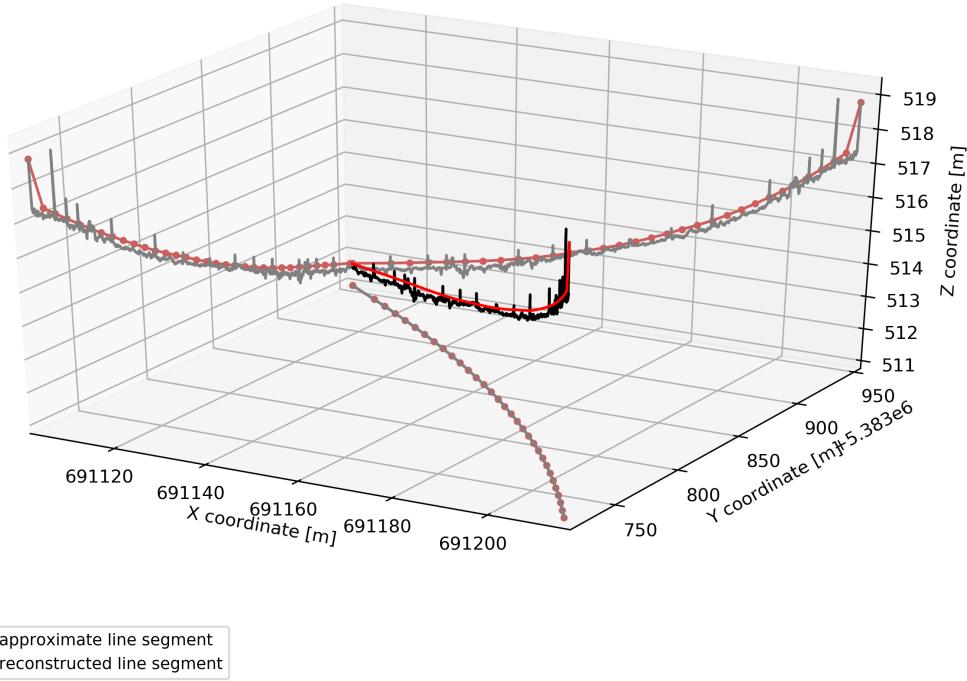


Figure 3.15: The reconstructed line segments and the unrefined DSM profile in UTM coordinate system (in Zone 32N).

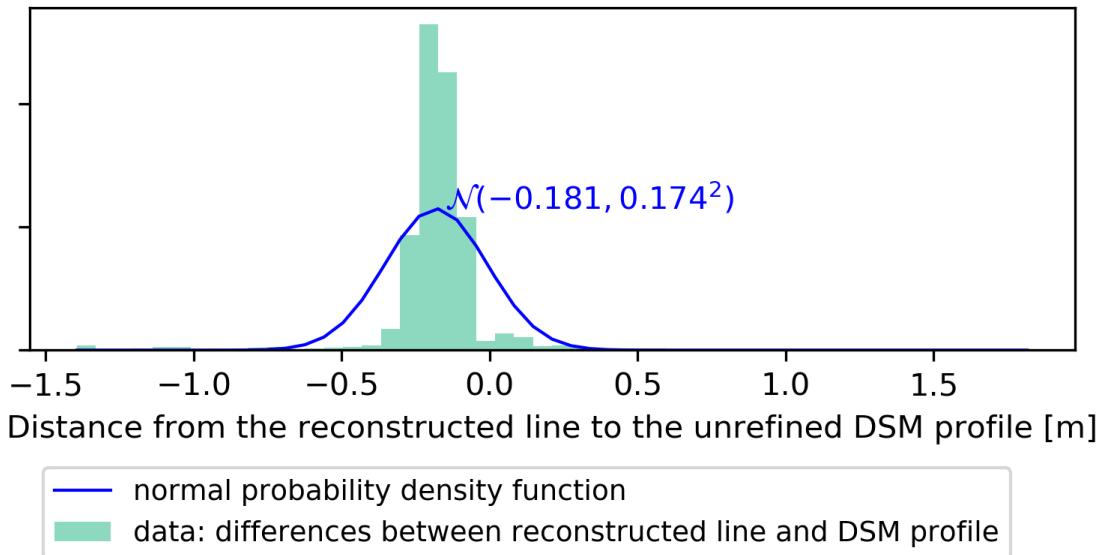


Figure 3.16: Histogram of the distances from the reconstructed line to the unrefined DSM profile.

3 Experimental Results and Evaluations

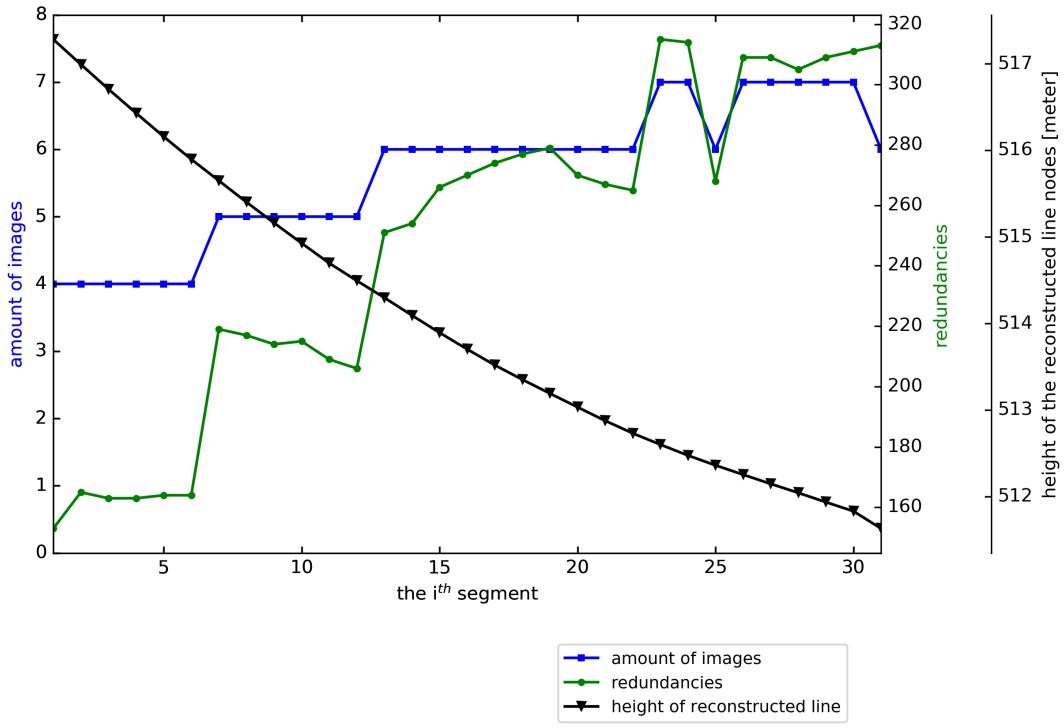


Figure 3.17: XXX...The relationship between image amount, the resulting reconstructed line segments, and the redundancies and posterior standard deviation in LS adjustment.

Figure 3.17 gives the information on the amount of covering images, the redundancies and the height value of the reconstructed nodes, of each segment. Note that in each segment, LS adjustment is processed independently.

Compare to the simulation data, where all line segments lie on a straight line and have same orientations, the true data (a continuous lane marking) is a curve line, whose line segments have different orientations from each other. Even some segments may have same image coverage configuration, their LS model may have different configuration due to their different orientations in 3D space. Figure 3.18 shows that the estimated parameters generally have smaller priori variance values with the increase of covering images. However there are some other factors influencing the configuration strength with same covering image amount.(vermutlich, the configuration strength in image space counts. show figures)

By increasing the configuration strength, the priori precision of the estimated parameters can be improved. **Not only the increase of the amount of covering images but also the line segment orientations in image space count for the configuration strength.**

3.4 True Data

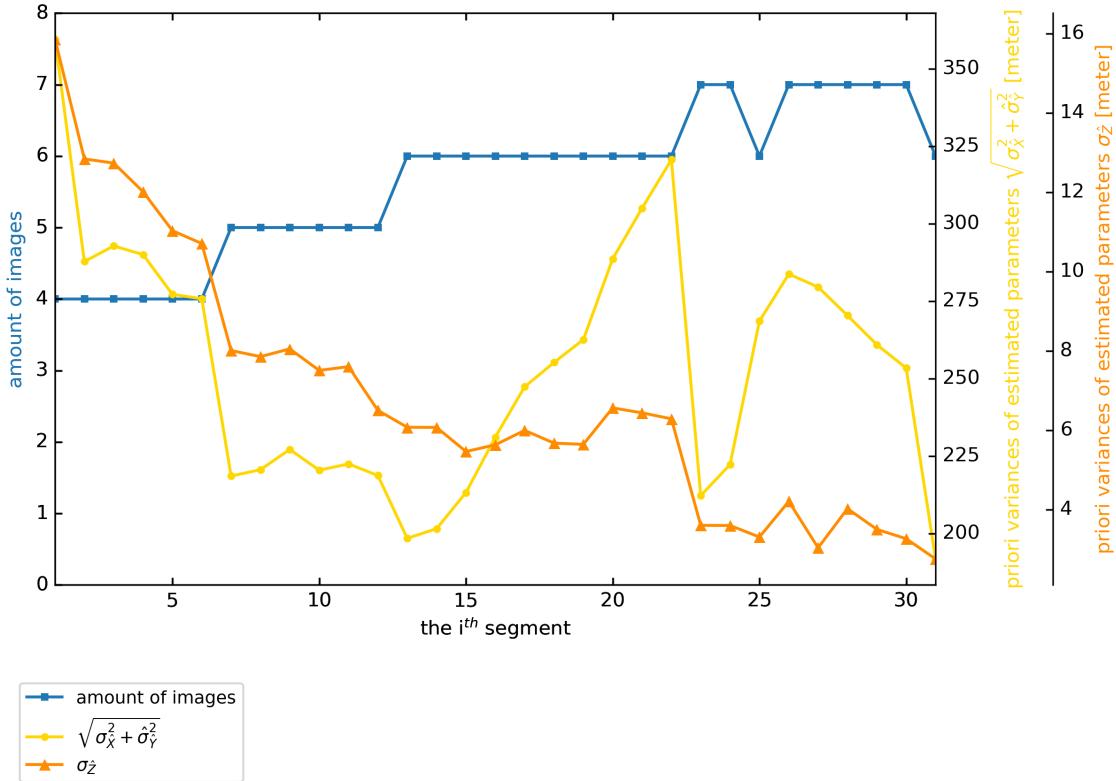


Figure 3.18: The variances of the estimated object coordinates, in horizontal and vertical directions.

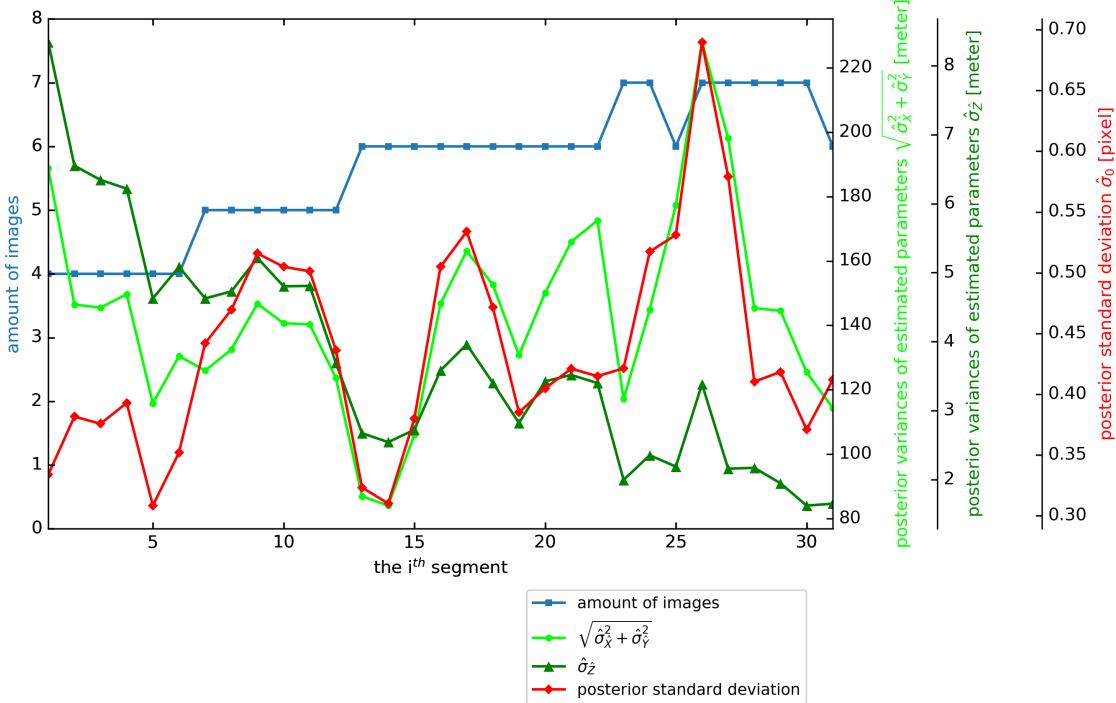


Figure 3.19: The estimated variances of the estimated object coordinates, in horizontal and vertical directions.

4 Conclusion and Future Work

Hier bitte einen kurzen Durchgang durch die Arbeit.

configuration plays an important roll in 3D reconstruction.

advantage our model can be used on lane markings reconstruction it not relying on textures or appearance-base matching

drawback unknown point-to-point relationship in image space, endpoints correspondence rely on quality of orientations

Future Work

Acknowledgements

The author gratefully acknowledges Dr.-Ing. Franz Kurz ...

Bibliography

- [ARB+15] M. Aeberhard, S. Rauch, M. Bahram, G. Tanzmeister, J. Thomas, Y. Pilat, F. Homm, W. Huber, N. Kaempchen. “Experience, Results and Lessons Learned from Automated Driving on Germany’s Highways.” In: *IEEE Intelligent Transportation Systems Magazine* 7.1 (Jan. 2015), pp. 42–57. ISSN: 1939-1390. DOI: [10.1109/MITS.2014.2360306](https://doi.org/10.1109/MITS.2014.2360306) (cit. on p. 11).
- [BFG05] H. Bay, V. Ferrari, L. V. Gool. “Wide-baseline stereo matching with line segments.” In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* 1 (June 2005), pp. 329–336. DOI: [10.1109/CVPR.2005.375](https://doi.org/10.1109/CVPR.2005.375) (cit. on p. 14).
- [Fra97] C. Fraser. “Digital camera self-calibration.” In: *ISPRS Journal of Photogrammetry and Remote Sensing* 52.4 (Aug. 1997), pp. 149–159. DOI: [10.1016/S0924-2716\(97\)00005-1](https://doi.org/10.1016/S0924-2716(97)00005-1) (cit. on p. 21).
- [HB08] H. Hirschmüller, T. Bucher. “Stereo Processing by Semiglobal Matching and Mutual Information.” In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30.2 (Feb. 2008), pp. 328–341. DOI: [10.1109/TPAMI.2007.1166](https://doi.org/10.1109/TPAMI.2007.1166) (cit. on p. 13).
- [HWB13] M. Hofer, A. Wendel, H. Bischof. “Line-based 3D Reconstruction of Wiry Objects.” In: *Proceedings of the 18th Computer Vision Winter Workshop* (Feb. 2013), pp. 78–85 (cit. on pp. 13–15).
- [Int14] S. International. *Taxonomy and Definitions for Terms Related to On-Road Motor Vehicle Automated Driving Systems*. On-Road Automated Driving (Orad) Committee, 2014 (cit. on pp. 11, 12).
- [JKTS10] A. Jain, C. Kurz, T. Thormählen, H. Seidel. “Exploiting global connectivity constraints for reconstruction of 3D line segments from images.” In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (June 2010), pp. 1586–1593. DOI: [10.1109/CVPR.2010.5539781](https://doi.org/10.1109/CVPR.2010.5539781) (cit. on p. 14).
- [Low99] D. Lowe. “Object recognition from local scale-invariant features.” In: *Proceedings of the Seventh IEEE International Conference on Computer Vision* 2 (Sept. 1999), pp. 1150–1157. DOI: [10.1109/ICCV.1999.790410](https://doi.org/10.1109/ICCV.1999.790410) (cit. on p. 13).
- [SZ97] C. Schmid, A. Zisserman. “Automatic Line Matching across Views.” In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* 1 (June 1997), pp. 666–671. DOI: [10.1109/CVPR.1997.609397](https://doi.org/10.1109/CVPR.1997.609397) (cit. on pp. 13, 14).

- [TA12] C. Topal, C. Akinlar. “Edge Drawing: A combined real-time edge and segment detector.” In: *Journal of Visual Communication and Image Representation* 23.6 (Aug. 2012), pp. 862–872. DOI: [10.1016/j.jvcir.2012.05.004](https://doi.org/10.1016/j.jvcir.2012.05.004) (cit. on p. 20).
- [TK95] C. Taylor, D. Kriegman. “Structure and motion from line segments in multiple images.” In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 17.11 (Nov. 1995), pp. 1021–1032. DOI: [10.1109/34.473228](https://doi.org/10.1109/34.473228) (cit. on p. 15).
- [WWH09] Z. Wang, F. Wu, Z. Hu. “MSLD: A robust descriptor for line matching.” In: *Pattern Recognition* 42.5 (May 2009), pp. 941–953. DOI: [10.1016/j.patcog.2008.08.035](https://doi.org/10.1016/j.patcog.2008.08.035) (cit. on p. 14).

All links were last followed on January 31, 2018.

Declaration

I hereby declare that the work presented in this thesis is entirely my own and that I did not use any other sources and references than the listed ones. I have marked all direct or indirect statements from other sources contained therein as quotations. Neither this work nor significant parts of it were part of another examination procedure. I have not published this work in whole or in part before. The electronic copy is consistent with all submitted copies.

place, date, signature