SINGLE VIEW COPLANAR PHOTOGRAMMETRY AND UNCERTAINTY ANALYSIS FOR TRAFFIC ACCIDENT RECONSTRUCTION

A Thesis

Submitted to the Faculty of Graduate Studies and Research

In Partial Fulfillment of the Requirements for the

Degree of Master of Applied Science in

Industrial Systems Engineering at the

University of Regina

by

Mehrdad Bakhtiari

Regina, Saskatchewan

June 2014

UNIVERSITY OF REGINA

FACULTY OF GRADUATE STUDIES AND RESEARCH

SUPERVISORY AND EXAMINING COMMITTEE

Mehrdad Bakhtiari, candidate for the degree of Master of Applied Science in Industrial Systems Engineering**,** has presented a thesis titled, ***Single View Coplanar photogrammetry and Uncertainty Analysis for Traffic Accident Reconstruction,*** in an oral examination held on June 9, 2014. The following committee members have found the thesis acceptable in form and content, and that the candidate demonstrated satisfactory knowledge of the subject material.

External Examiner:	Dr. Howard Hamilton, Department of Computer Science
Supervisor:	Dr. Raman Paranjape, Electronic Systems Engineering
Committee Member:	Dr. Amir Henni, Industrial Systems Engineering
Committee Member:	Dr. Adisorn Aroonwilas, Industrial Systems Engineering
Chair of Defense:	Dr. Jim Farney, Department of Political Science

Abstract

Photogrammetry is defined as the science of measuring distances and objects by using photographs. Rapid advances in computer technology and digital photography in recent years have resulted in growing applications of digital photogrammetry. Although various aspects of photogrammetry have been investigated; the question of uncertainties in measurement, which is important for forensic applications, such as traffic accident reconstruction, has not been widely explored. This study attempted to develop an applicable comprehensive mathematical model to perform coplanar photogrammetry and provide a measure of uncertainty for its measurements. Photogrammetry requires information about the interior and exterior orientation of the camera. Process maximum likelihood estimation (MLE) of the interior parameters of the camera or camera calibration and methods to provide uncertainty for those parameters are discussed. To find the maximum likelihood estimation of the exterior orientation of the camera, a new pose estimation algorithm has been developed that requires only information about the length of three or more lines in the scene while taking into account nonlinearities due to lens distortion. Formulation to find the maximum likelihood estimation of back projection of individual points to world frame considering uncertainties in interior and exterior orientation has been provided. Finally, all uncertain models have been compiled into an uncertain comprehensive model for photogrammetric measurement on a coplanar scene. A computer application equipped with a simulation tool has been developed for the Windows operating system to implement the model. A simulation has been performed to investigate the effect of various parameters on photogrammetry measurement and it was found that the error in the length of reference lines has the most significant effect.

Aerial photos of a mock accident scene were used for the experimental study. Experiment results showed that the proposed method can provide distance measurement with an average error of 1%, while the uncertainty for its measurements follows the three-sigma rule.

Acknowledgment

Firsthand and foremost, I am indebted to my supervisor Dr. Raman Paranjape for all the guidance, supports and encouragements he provided me during my years in University of Regina, without whom my study would never have been finished. I also owe a particular thank to my program chair Dr. Amr Henni whom I benefited from his wise advice and enjoyed his backings at different stages of my study. My gratitude again goes to Sgt. Ryan Case for providing me with invaluable insight into collision reconstruction science and the opportunity to perform data collection. Thanks go as well to my colleague in the University of Regina UAV group Mr. Seang Cau for his assistance in my experiments. I also need to thank Saskatchewan Government Insurance company for providing financial support for this project.

Table of Contents

Abstract ii

Acknowledgment iv

Table of Contents v

List of Tables x

List of Figures xi

Nomenclature xiii

1. Introduction 1

1.1. Definitions 1

1.2. Historical Overview 1

1.3. Photogrammetry for Traffic Accident Reconstruction 3

1.4. Problem Statement 6

1.5. Thesis Structure 7

2. Theoretical Foundation and Mathematical Models 9

2.1. Introduction 9

2.2. Camera Projection Model 9

2.2.1. Central Projection Camera Model and Perspective Transform 9

2.2.2. Digital Cameras and Pixel Coordinate Representation 12

2.2.3. Lens Distortion 13

2.2.4. Complete Projection Model 16

2.3. Camera Calibration 17
2.3.1. Introduction 17
2.3.2. Zhang's Calibration Method 19
2.3.2.1. Overview 19
2.3.2.2. Closed Form Solution 22
2.3.2.3. Refining intrinsic and extrinsic parameters 24
2.3.2.4. Finding Radial Distortion Coefficients 25
2.3.2.5. Complete Maximum Likelihood Estimation 25
2.3.3. Calibration Parameters 26
2.4. Planar Scene Reconstruction 26
2.4.1. Closed Form Initial Guess 26
2.4.2. Maximum Likelihood Estimation for Back-Projection 28
2.5. Pose Estimation 28
2.5.1. Overview 28
2.5.2. PnP Pose Estimation Algorithms 30
2.5.3. Studied PnL Pose Estimation Algorithms 31
2.5.4. Limitations of Conventional Pose Estimation Algorithms 32
2.5.5. Pose Estimation Algorithm 34
2.5.6. Proof of Preserving Line Segment Lengths with Proposed Algorithm 34
2.5.7. Maximum Likelihood Estimation for Pose 38

2.6. Least Square Methods and Maximum Likelihood Estimation 40
2.6.1. Overview 40
2.6.2. Definitions 40
2.6.3. Gauss-Newton Algorithm 41
2.6.4. Levenberg-Marquardt Algorithm 43
2.6.5. Uncertainty in Model Parameters 44
2.6.6. Uncertainty in Model Predictions 45
2.6.7. Calculation of Jacobian Matrix 45
2.6.8. Uncertainties in Camera Intrinsic and Lens Distortion Parameters 46
2.6.9. Uncertainty in Pose Estimation 47
2.6.10. Uncertainty in Scene Reconstruction 47
2.6.11. Uncertainty in Measured Lengths 47
3. Implementation 49
3.1. Introduction 49
3.2. Developed Computer Software 51
3.2.1. Overview 51
3.2.2. Calibration Tab 51
3.2.3. Pose Estimation Tab 52
3.2.4. Photogrammetry Tab 52
3.3. Numerical and Image Processing Libraries 53

3.3.1. Overview 53

3.3.2. Emgu CV 53

3.3.3. ALGLIB 54

3.4. Simulation Tool 56

3.5. Summary 58

4. Results and Discussion 59

4.1. Overview 59

4.2. Simulation 60

4.2.1. Scene Configuration 60

4.2.2. Simulation Results 61

4.2.3. Simulation Scenarios 61

4.2.4. Effect of Error in Reference Line Measured Length 62

4.2.5. Effect of Error in Pixel Selection of Reference Lines 64

4.2.6. Effect of Camera Height and Look Direction 66

4.2.7. Camera Focal Length and Principal Point 68

4.3. Experiment 70

4.3.1. Overview 70

4.3.2. Experimental Setup 70

4.3.3. Measurement with the Total Station 71

4.3.4. Camera Calibration 74

4.3.5. Pose Estimation and Photogrammetry Analysis 74

4.3.6. Comparison between Total Station and Photogrammetry 77

4.3.7. Summary 78

5. Conclusion and Further works 80

References 84

List of Tables

Table 2.1. 26

Table 2.2. List of camera characteristic parameters as a result of camera calibration process..... 26

Table 4.1. Actual values of parameters for simulation 62

Table 4.2. Estimated values for intrinsic and lens distortion parameters for SONY NEX-5R camera of the experiment 73

Table 4.3. Results of pose estimation for selected photos 76

Table 4.4. Measurement comparison between total station and photogrammetry 78

List of Figures

Figure 2.1. Central projection camera model..... 11

Figure 2.2. Geometric distortion, vectors and contours of displacement: a) pure radial distortion b) pure tangential distortion c) combination of radial and tangential distortion d) sample grid subjected to both radial and tangential distortion 16

Figure 2.3. Sample tile of nine images from a chessboard pattern for calibration 20

Figure 3.1. Workflow of developed coplanar photogrammetry application..... 50

Figure 3.2. Snapshot of developed computer program GUI..... 50

Figure 3.3. Graphical User Interface of simulation tool 57

Figure 4.1. Simulated scene configuration 60

Figure 4.2. Effect of error in measured length of L1 on photogrammetry error..... 63

Figure 4.3. Effect of error in measured length of L2 on photogrammetry error..... 64

Figure 4.4. Photogrammetry error due to wrong pixel selection of first endpoint of line L1in horizontal direction..... 65

Figure 4.5. Photogrammetry error due to wrong pixel selection of first endpoint of line L1in vertical direction..... 66

Figure 4.6. Scene arrangement for finding effect of camera look direction on photogrammetry error 67

Figure 4.7. Variation of photogrammetry error due to change in camera look direction . 68

Figure 4.8. Effect of camera height on photogrammetry error 68

Figure 4.9. Effect of error in horizontal focal length on photogrammetry error 69

Figure 4.10. Effect of error in vertical element of focal length on photogrammetry error70

Figure 4.11. Simulated accident scene from ground view, (a) vehicles (b) DraganFlyer X4-ES 72

Figure 4.12. Schematic of measured points with total station surveying equipment 72

Figure 4.13. Lens distortion contours for camera which has been used in experiment 73

Figure 4.14. Selected photos for photogrammetry analysis with overlaid reference lines 76

Nomenclature

$\boldsymbol{A}$	Camera intrinsic matrix	-
$\boldsymbol{C}$	3×4 camera matrix	-
f_x	Horizontal focal length	[length]
f_y	Vertical focal length	[length]
$\boldsymbol{h}$	Homography matrix	-
$\boldsymbol{J}$	Jacobian matrix	-
$k_1 \; and \; k_2$	Radial distortion coefficients	-
L	Line segment length	[length]
$\boldsymbol{M}$	Point coordinate in world frame	-
$\boldsymbol{m}$	Observed image coordinate	-
$\hat{\boldsymbol{m}}$	Estimated image coordinate	-
$p_1 \; and \; p_2$	Tangential distortion coefficients	-
r	Radial distance of projection point from principal point	[pixel]
$\boldsymbol{R}$	Rotation matrix	-
$\vec{\boldsymbol{R}}$	Rotation vector	-
$\boldsymbol{T}$	Homogenous pose in matrix form	
$\boldsymbol{t}$	Translation vector	
u	Image plane horizontal element of coordinate	[pixel]
u_0	Camera principal point horizontal element of coordinate	[pixel]

u_d	horizontal element of distorted image coordinate	[pixel]
v	Image plane vertical element of coordinate	[pixel]
$\boldsymbol{V}$	Vector representation of a line segment	-
v_0	Camera principal point vertical element of coordinate	[pixel]
v_d	vertical element of distorted image coordinate	[pixel]
α	Horizontal focal length	[pixel]
β	Vertical focal length	[pixel]
σ	Uncertainty	-
$\delta_{,Rad}$	Radial distortion	[pixel]
$\delta_{,Tan}$	Tangential distortion	[pixel]
ϵ_z	Height difference perturbation	[length]

1. Introduction

1.1. Definitions

Photogrammetry is defined as "*the science of making reliable measurements by the use of photographs and especially aerial photographs (as in surveying)*" (Merriam-Webster Dictionary). In the other words, photogrammetry can be defined as a process of extracting shape and location of an object from one or more photographs (Luhmann, Robson, Kyle, & Harley, 2006).

In photography all 3-D objects are mapped on a 2-D plane. As a result, a set of information is lost in this process. Reconstructing an object from its image requires a description of the optical process of image formation.

A description of image formation usually requires two types of information: interior orientation and exterior orientation. Interior information characterizes the geometry and other characteristics of the camera. Exterior orientation parameters define the location and look direction of camera with respect to the object.

Performing photogrammetry needs these two types of information. Details of techniques for acquiring interior and exterior orientation of camera for photogrammetry application are discussed in length in following chapters.

1.2. Historical Overview

The history of photogrammetry is as long as that of photography. Just a few years after the invention of photography in the 1830s and 1840s, a French military officer conducted a set of experiments in 1849 on the image of a façade of the Hotel des Invalidas

(Luhmann, Robson, Kyle, & Harley, 2006). Maydenbauer, who invented the word "Photogrammetry," used photographs to extract plans of the cathedral of Wetzlar in 1858. He later developed an archive of 16,000 images of important historical buildings for the state of Prussia and he also invented methods for reproduction of plans of building façades (Luhmann, Robson, Kyle, & Harley, 2006).

The developments in photogrammetry since around 1850 can be classified in four periods (Konecny, 1985, pp. 919-933):

- Plane table photogrammetry – 1850 to 1900.
- Analog photogrammetry – 1900 to 1960.

  Analytical photogrammetry – 1960 to present, and
- Digital photogrammetry – 1980s to present.

After the first period, with the introduction of stereoscopic measurement, applications of photogrammetry that started as an experimental measurement method for architects rapidly grew to a widely accepted technique to develop topographic maps at a scale of 1:10000 for countries, such as Britain or Belgium, in coming decades. This process gained significant speed due to rapid developments of in aviation that provided not only vertical images from the earth, but also made access to remote areas possible. During this period analog plotters were the main tool in photogrammetry. The analog plotter or stereoplotter is a special optical table that can be used to extract elevation information from two overlapping stereoscopic aerial images. It should be noted that non-topographic and close range photogrammetry was sporadic before the 1930s (Collier, 2002).

The invention of digital computers started a new era in photogrammetry. While analog computations puts rigid restriction on the calculation process in terms of the number of inputs and outputs and therefore has limited accuracy, numerical methods can provide virtually unlimited accuracy that is bounded only to the applied mathematical model. To be more specific, numerical methods can provide an opportunity to improve accuracy by using overdetermined information. For example, instead of using only two images for extracting information in a stereoscopic system, numerical methods can benefit from all available images from a single scene to enhance the accuracy. In photogrammetry, this means to solve the triangulation problem in an overdetermined manner, which is known as bundle adjustment (Luhmann, Robson, Kyle, & Harley, 2006).

The emergence of digital photography and the use of opto-electronic sensors in conjunction with advances in computer technology in the 1980s, have enabled the development of pure digital photogrammetric systems from image acquisition up to analysis. Moreover, automatic recognition and measurement of image features became possible using modern computers (Luhmann, Robson, Kyle, & Harley, 2006).

1.3. Photogrammetry for Traffic Accident Reconstruction

The history of using photogrammetry techniques for mapping accident sites dated back to 1933. At that time, Swiss police started to use Wild stereometric cameras for mapping accident sites (Luhmann, Robson, Kyle, & Harley, 2006).

Fenton et al (Fenton & Kerr, 1997) investigated the use of PhotoModeler software to extract required information for diagramming accident scene. They proposed a four step procedure that uses multiple images from different cameras to reconstruct scenes:

determining world frame coordinates of control points, estimating camera characteristics, rectifying the photograph and trace the rectified image. They stated that the advantage of this process is its ability to extract information from old photographs.

Cooner et al (Cooner, Scott, & Balke, 2000) reported the results of their study on comparing measurement with total station surveying equipment and photogrammetry in terms of time saving and being cost effective. They concluded that photogrammetry is a cost effective and time-saving tool for incident management applications. They reported that photogrammetry requires about 50% of the training time for total station. They also pointed out some drawbacks of photogrammetry in this application such as: difficulty in handling long scenes, extra time required in off-line processing (3) necessity of enhancing visualization of evidences such as skid marks in nights. They also stated that the majority of police officers whom were interviewed for their study would recommend the use of photogrammetry to other departments.

Xu et al (Xu & Lu, 2002) investigated the application of two photographs with different view angles used for the measurement of an accident scene. They utilized a special reference object for camera calibration on the scene. Image matching in their study was performed based on the correlation of two points in each photograph.

Fraser et al (Fraser C. , 2006) described the overall procedure for acquiring the interior and exterior orientation of the camera for the developed commercial software package *iWitness,* which is used for close-range photogrammetry on accident scenes. This software package employed multi-images photogrammetry. For estimating exterior orientation, The software package employs a Monte Carlo type strategy to find initial

guess. The software package has an automatic calibration feature via colour coded targets. Using this software requires the manual marking of correspondences on images manually by operator (Fraser, Hanley, & Cronk, 2005).

Du et al (Du, Jin, Zhang, Shen, & Hou, 2009) studied the application of measurement of traffic accident geometry features using photogrammetry. They used the PhotoModeler software package for photogrammetric measurement. PhotoModeler employs multiple view geometry for extracting the exterior orientation of the camera. In this study, they used special target objects to introduce point correspondences. They compared the results of applying photogrammetric measurement with conventional roller tape employed by the Shanghai Municipal bureau of Public Security in 142 traffic accident cases. They concluded that photogrammetry can be up to 67% faster than conventional roller tape and therefore cost effective. As a result of numerical simulation and experiments, they stated that by having a correct and adequate set of reference points, photogrammetry provided acceptable accuracy. They further expanded their study by using photogrammetry for the reconstruction of 3-D deformations to analyze accident's dynamic. They stated that collected information through photogrammetry can be used as initial data for damage analysis software, such as SMAC, CRASH and PC-CRASH.

A study by Randles et al (Randles, Jones, Welcher, Szabo, Elliott, & MacAdams, 2010) provided a comparison between the accuracy of photogrammetry versus hands on measurement for deformation analysis in traffic accident reconstruction. They employed PhotoModeler software for post impact photogrammetric geometry analysis. Utilizing the assistance of qualified specialists equipped with a crush jig, plumb bobs and measurement tape, they compared measurements of both techniques. They found that the

mean difference between baseline and hands-on measurements is 0.6 ± 1.4 cm while the mean difference between the baseline and photogrammetry measurement is about 0.1 ± 1.0 cm. They claimed that photogrammetry is slightly more accurate than hands-on measurement.

Neale et al (Neale, Hessel, & Terpstra, 2011), investigated the effect of lens distortion on photogrammetric measurement error. They evaluated how lens distortion can affect image and its associated errors in photogrammetry measurement for evidences in accident scene reconstruction. In their study they investigated lens distortion parameters for five different lens focal lengths (from 28 mm to 50 mm) and 69 different cameras. They also provided an analysis on the effect of lens distortion on photogrammetric measurement in accident scene reconstruction based on the critical speed formula to acquire vehicle velocity from the radius of the path of the leading front tire skid mark. In an example they showed the effect of distortion can generate about six miles per hour error in determination of vehicle speed which is originally was around 42 mph. They also discussed common methods for removing this effect from images.

1.4. Problem Statement

A Literature review shows that none of the previous studies addressed the following issues:

- Most of them used multiple view photogrammetry that requires extensive time for finding exterior orientation and also for each measurement.

- Little or almost nothing can be found in scientific literature on addressing the question of uncertainty in photogrammetry measurement for traffic accident reconstruction.
- Most of the previous studies or developed software packages require introducing a specific target object to the scene which is not always practical

Therefore, the following objectives, which can also be assumed as contributions, have been defined for the current study:

- Developing a comprehensive mathematical model that can address single view coplanar photogrammetry with uncertainty analysis while taking into account nonlinearities in projection equations.
- Developing a flexible method to estimate exterior orientation.
- Developing a computer application to implement the mathematical model, and
- Performing a simulation and experiment to investigate the accuracy of the proposed method, both in terms of measurement and uncertainty analysis.

1.5. Thesis Structure

This thesis is divided into five chapters. The introduction is provided in chapter one. Chapter two provides the details of the mathematical model to acquire camera interior and exterior orientation, photogrammetry measurement and uncertainty analysis. Chapter three presents the implementation of the proposed mathematical model and describes the developed computer application tool and its features. Chapter four provides the results of the simulation and experiment and discusses various aspects of error in photogrammetry

analysis and finally chapter five provides concluding remarks and outlines possible future research.

2. Theoretical Foundation and Mathematical Models

2.1. Introduction

In this chapter, mathematical foundation, details of the problem at hand and proposed solution methods will be discussed. The first section provides the formulation details, which has been used to model camera projection using camera intrinsic parameters. The second section describes the camera calibration process and additional implemented measures to extract the uncertainty values for the camera intrinsic matrix and distortion coefficients. The third section addresses the proposed pose estimation method that has been developed for current study and provides the proof for validity of the algorithm. Finally, in last section, all sub models have been combined into a comprehensive model that can provide photogrammetric measurement with uncertainties.

2.2. Camera Projection Model

2.2.1. Central Projection Camera Model and Perspective Transform

A projective camera model maps a point from 3-D space to 2-D space: $\mathbb{R}^3 \rightarrow \mathbb{R}^2$. To represent the image formation of the camera, this model should follow perspective transformation. Among different existing camera models, the central projection model is commonly used for computer vision. Figure 2.1 shows the schematic of this model. In the central projection model it is assumed that (Corke, 2011):

- All rays converge on a single point known as the camera origin.
- The image is projected on an image plane, which is placed at a distance equal to the focal length (f) and which is also perpendicular to the optical axis of the camera.

Using the triangle similarity for a point in a 3-D world, such as P at the coordinates $P(X, Y, Z)$, the projected representation of that point namely $p(x, y)$ can be calculated by:

$$x = f\frac{X}{Z}, y = f\frac{Y}{Z} \tag{2-1}$$

Equation (2-1) can be rewritten in matrix format as follows:

$$p = \begin{bmatrix} f_x & 0 & 0 \\ 0 & f_y & 0 \\ 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \tag{2-2}$$

It should be noted that during this transformation, the depth information is lost. Therefore, every point on the image plane can be associated to an infinite number of points on the line that crosses the camera origin and actual point P in the real world.

Equation (2-2) can be rewritten in homogenous form by using the homogenous coordinate of point $P_c = (X, Y, Z, 1)^T$ in linear form (Corke, 2011):

$$p = \begin{bmatrix} f_x & 0 & 0 & 0 \\ 0 & f_y & 0 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix} P_c \tag{2-3}$$

or

$$p = \boldsymbol{C}P \tag{2-4}$$

where $\boldsymbol{C}$ is a 3×4 matrix known as camera matrix. It should be noted that subscript c in P_c denotes to coordinates of the point P in camera coordinate system.

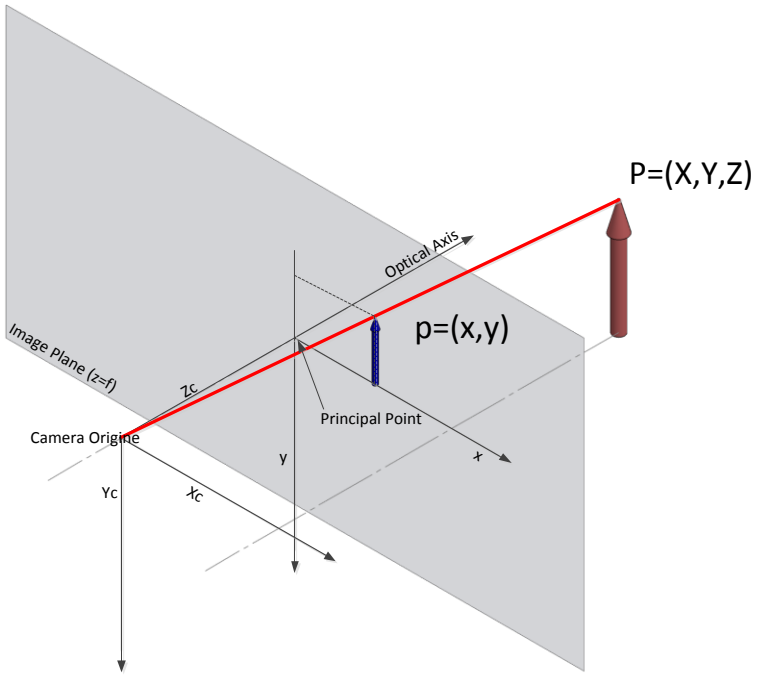

Figure 2.1. Central projection camera model

To further expand the model, first a "pose" term is be introduced. Pose or exterior orientation is defined as the "position and orientation of camera reference frame with respect to world reference frame" (Haralick, Lee, Ottenberg, & Nölle, Glossary of computer vision terms, 1991). In general, pose has six degrees of freedom, three for the rotation vector and three for the translation vector elements. Given all these pieces of information, the complete position and look direction of and camera can be defined. Pose is usually expressed in the homogenous 4 × 4 matrix form as follows:

$$\boldsymbol{T} = \begin{bmatrix} \boldsymbol{R}_{3\times3} & \boldsymbol{t}_{3\times1} \\ \boldsymbol{0}_{1\times3} & 1 \end{bmatrix} \tag{2-5}$$

where $\boldsymbol{R}$ is 3×3 rotation matrix and $\boldsymbol{t}$ is 3×1 translation vector.

In general, when the camera is located at an arbitrary pose T_c (in matrix form) with respect to the world coordinate system, to acquire the position of a point such as P_w in world coordinate system on image plane, it is needed first to determine the coordinates of that point in the camera coordinate system and then proceed with the projection. This can be formulated as follows:

$$P_c = \boldsymbol{T_c^{-1}} P_w \tag{2-6}$$

where T_c represents the homogenous matrix form of the camera pose in world frame, therefore:

$$\boldsymbol{T_c^{-1}} = \left(\begin{bmatrix} \boldsymbol{R_c} & \boldsymbol{t_c} \\ \boldsymbol{0}_{1\times3} & 1 \end{bmatrix}\right)^{-1} = \begin{bmatrix} \boldsymbol{R_c^T} & -\boldsymbol{R_c^T t_c} \\ \boldsymbol{0}_{1\times3} & 1 \end{bmatrix} \tag{2-7}$$

where $\boldsymbol{R_c}$ and $\boldsymbol{t_c}$ denote the rotation matrix and translation vector of the camera frame with respect to the world frame that is known as camera pose. Using this notation, equation (2-4) can be rewritten as follows:

$$p = \boldsymbol{C} P_c = \boldsymbol{C T_c^{-1}} P_w = \begin{bmatrix} f_x & 0 & 0 & 0 \\ 0 & f_y & 0 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{R_c^T} & -\boldsymbol{R_c^T t_c} \\ \boldsymbol{0}_{1\times2} & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{2-8}$$

Equation (2-8) provides the basis for the central projection camera model.

2.2.2. Digital Cameras and Pixel Coordinate Representation

In digital cameras the image plane consists of a $W \times H$ array of light sensor that collects information that is represented on a digital image in the form of pixels. The convention

for coordinate of pixels uses the top left corner of the image plane as the origin. The pixel coordinate of point $p(x, y)$ on the image plane can be determined by (Corke, 2011):

$$u = \frac{x}{\rho_w} + u_0,\ v = \frac{y}{\rho_h} + v_0 \tag{2-9}$$

where ρ_w and ρ_h are the width and height of the pixel in unit length and (u_0, v_0) is the coordinate of the principal point. Principal point is defined as the location of the intersection of the camera's optical axis and image plane. Applying above transformation to equation (2-8), substituting $\alpha = \frac{f_x}{\rho_w}$, $\beta = \frac{f_y}{\rho_h}$ and defining c as the skewness ratio for photosites; provides the general formulation for determining the pixel coordinate of P_w in the world coordinate system that is projected on the image plane:

$$p_{pixel} = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} \alpha & c & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}}_{intrinsic} \underbrace{\begin{bmatrix} \boldsymbol{R_c^T} & -\boldsymbol{R_c^T t_c} \\ \mathbf{0}_{1\times 2} & 1 \end{bmatrix}}_{extrinsic} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{2-10}$$

The first two matrices on the right hand side of the above equation are known as intrinsic parameters. Intrinsic parameters are independent from the scene and characterize the specification of the camera. The second matrix in intrinsic parameters might contain non-unity coefficients to represent light sensor (photosites) skewness and other characteristics that are assumed to be negligible in the current study. The third matrix shows the relative position and look direction of the camera (pose) relative to world frame and is known as extrinsic parameters.

2.2.3. Lens Distortion

Every camera lens shows some level of optical aberrations that ultimately causes image distortion. Aberrations are the results of the required shape of the lens, materials which

have been used in lens manufacturing, assembly error in multiple lenses configurations and other effects (Neale, Hessel, & Terpstra, 2011). Different types of optical aberrations exist and notable types of distortions as a result of them include (Corke, 2011):

- chromatic aberration (color fringing) that happens because of a different refractive index at different wavelengths.
- spherical aberration or astigmatism as a result of a variation of focus over the lens.
- geometric distortion that causes the points on the image plane to be seen at a different location from the expected location which has been calculated by equation (2-10).

Geometric distortion is known to have the largest effect on error in photogrammetric applications (Corke, 2011). There are two types of geometric distortion: radial and tangential distortion. Figure 2.2 demonstrates an example of the effect of these two types of distortions.

In radial distortion, ideal projected points of equation (2-10) are distorted symmetrically along the radial direction from the image principal point. This is caused by an imperfect lens shape. The widely accepted mathematical model for radial distortion expresses pixel displacement as a polynomial function of the radius from the principal point (Heikkila & Silvén., 1997):

$$\delta_{x,Rad} = (u - u_0)(k_1 r^2 + k_2 r^4 + k_3 r^6 + \cdots)$$
$$\delta_{y,Rad} = (v - v_0)(k_1 r^2 + k_2 r^4 + k_3 r^6 + \cdots) \quad (2\text{-}11)$$

where in the above equation:

- $\delta_{x,Rad}$ and $\delta_{y,Rad}$ are accordingly horizontal and vertical pixel displacements due to radial distortion.
- $r = \sqrt{(u-u_0)^2 + (v-v_0)^2}$ has been defined as the radial distance of projection point from camera principal point.
- and k_1,$k_2 \ldots k_n$ are coefficients of the radial distortion model.

In the current study, a fourth order polynomial has been introduced for accounting radial distortion and higher order terms and coefficients, such as $k_{3,}$ and so on have been assumed negligible.

Tangential or decentring distortion usually happens as a result of inaccurate lens alignment. (Zhang Z. , 2004). In the same lens distortion mathematical model, tangential distortion has been defined as follows:

$$\begin{aligned} \delta_{x,Tan} &= [p_1(r^2 + 2(u-u_0)^2) + 2p_2(u-u_0)(v-v_0)](1 + p_3 r^2 + \cdots) \\ \delta_{y,Tan} &= [2p_1(u-u_0)(v-v_0) + p_2(r^2 + 2(v-v_0)^2)](1 + p_3 r^2 + \cdots) \end{aligned} \quad (2\text{-}12)$$

where in above equation

- $\delta_{x,Tan}$ and $\delta_{y,Tan}$ are accordingly horizontal and vertical pixel displacements due to tangential distortion.
- p_1,$p_2 \ldots p_n$ are coefficients of the radial distortion model.

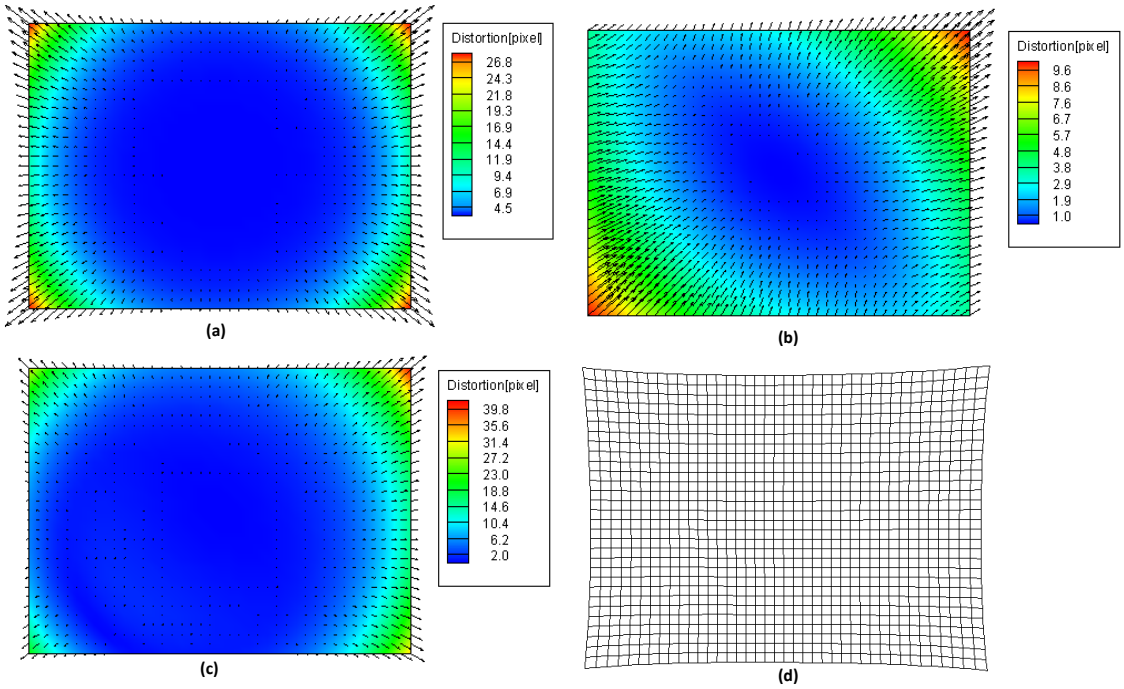

Figure 2.2. Geometric distortion, vectors and contours of displacement: a) pure radial distortion b) pure tangential distortion c) combination of radial and tangential distortion d) sample grid subjected to both radial and tangential distortion

Again, it should be noted that in the current study coefficients at order of p_3 and higher have been assumed insignificant.

2.2.4. Complete Projection Model

Combining equations (2-11) and (2-12) with equation (2-10) provides the complete projection model which has been used for the current study:

$$p_{pixel,distorted} = \begin{bmatrix} u_d \\ v_d \\ 1 \end{bmatrix} = \begin{bmatrix} u + \delta_{x,Rad} + \delta_{x,Tan} \\ v + \delta_{y,Rad} + \delta_{y,Tan} \\ 1 \end{bmatrix} \tag{2-13}$$

2.3. Camera Calibration

2.3.1. Introduction

Generally, equation (2-13) contains a set of unknown parameters. These parameters include intrinsic parameters, extrinsic parameters and lens distortion coefficients. Camera calibration is defined as a process of finding intrinsic-lens distortion parameters and extrinsic parameters with respect to the world coordinate system.

Although in camera calibration both intrinsic and extrinsic parameters are estimated, the focus is mostly on intrinsic and lens distortion parameters because extrinsic parameters are scene-dependent and vary from one photograph to anther. On the other hand, intrinsic and lens distortion parameters are related only to camera characteristics and can be used again as long as no change happens in the camera's settings (Qi, Li, & Zhenzhong, 2010).

Calibration techniques mostly use sets of points with known relative world coordinates and their known corresponding image coordinates to find calibration parameters (Corke, 2011). That set of points with known relative world coordinates forms what is called the calibration target. Based on the type of target, calibration techniques can be classified into three categories (Zhang Z. , 2004):

- 3-D reference object based calibration: calibration parameters are achieved by observing an object that its geometry in the 3-D world is known to have a high level of accuracy. In almost all cases, calibration target comprises of two or three pairwise orthogonal planes. These methods require an expensive calibration device and a time consuming setup while they can provide accurate parameters estimation (Faugeras, 1993).

- 2-D plane based calibration: techniques based on this type of calibration target use the observation of a planar pattern at different relative position-look directions to estimate calibration parameters. Since the pattern is easily reproducible, these techniques have an easy setup.
- 1-D line based calibration: the calibration target employed in this category consists of a set of collinear points. An example of this setup is a string of balls hanging from the ceiling.
- Self-calibration: In this category no specific calibration target is used and only image point correspondences from different images that have been taken by moving the camera in a static scene are required. However these techniques require estimation of a large number of parameters and are harder to solve as a mathematical problem.

In the current study a 2-D planar pattern based calibration technique has been used to estimate the camera's intrinsic and distortion model parameters. The detail of this method will be discussed in following section.

It should be noted that, since distortion models, such as what has been presented earlier in equation (2-11), apply a nonlinearity to camera geometric model, the problem of camera calibration boils down to a nonlinear regression analysis.

2.3.2. Zhang's Calibration Method

2.3.2.1. Overview

Zhang proposed a flexible calibration technique based on the observation of a planar pattern shown at a few different position-look directions that has been evolved into one of the most popular calibration techniques in computer vision applications (Corke, 2011).

His method has been applied in or inspired calibration tools in numerous machine vision libraries, such as Camera Calibration Toolbox for MATLAB (Bouguet, 2010), OpenCV Computer vision library (Bradski & Kaehler, 2008) and its .NET wrapper Emgu CV, have been used in the current study.

A common approach in implementing Zhang's calibration technique uses multiple images from a chessboard pattern for the calibration target, which are taken at different relative pose as has been shown in Figure 2.3. The chessboard pattern provides the advantage of finding image correspondence automatically. This means that by knowing the number of squares in horizontal and vertical directions, a corner detection algorithm can be used to detect (or match) corresponding points in different images without manual intervention.

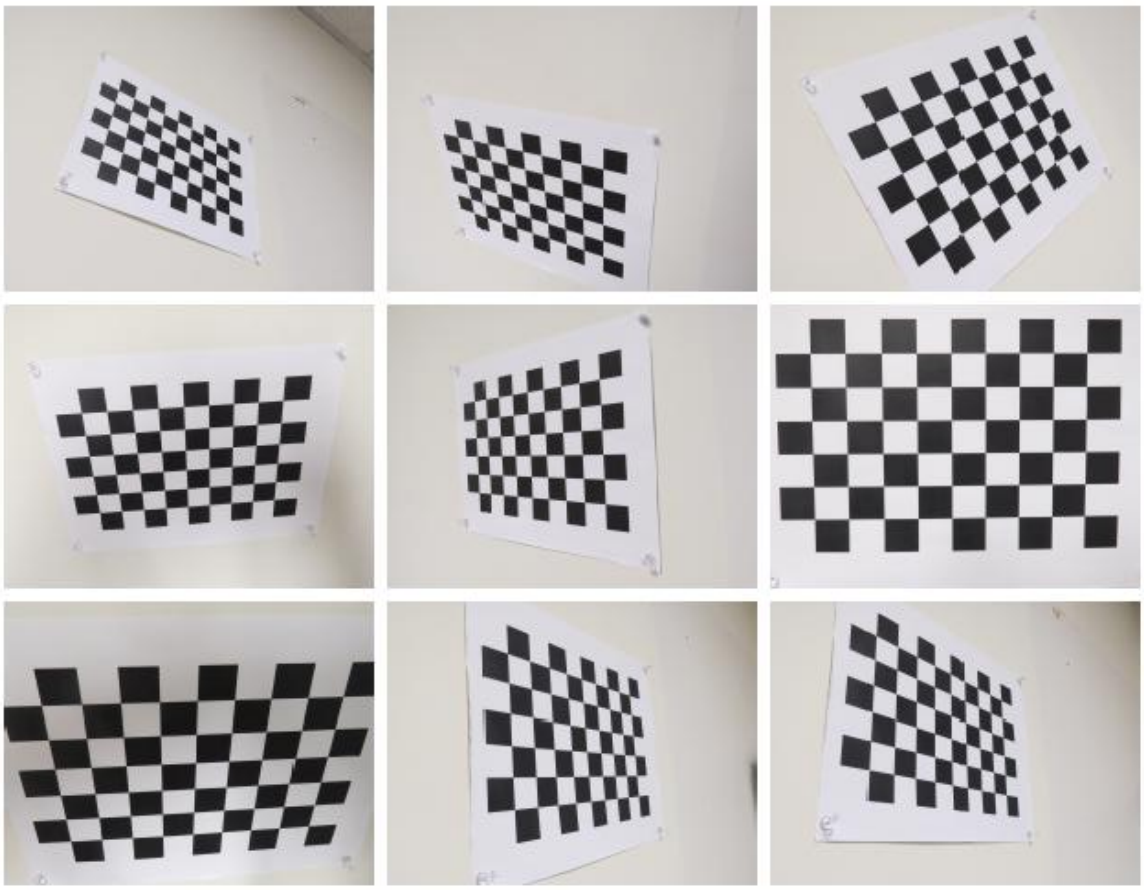

Figure 2.3. Sample tile of nine images from a chessboard pattern for calibration

Zhang's method extensively benefitted from the Levenberg-Marquardt minimization algorithm in several steps and since the same algorithm has been used for the proposed pose estimation method in the current study, details of that will be discussed later at the end of this chapter.

Zhang's method includes multiple steps in determining camera parameters:

- It starts with calculating the Maximum Likelihood Estimation for the homography matrix of three or more images using a nonlinear least square minimization with the Levenberg-Marquardt algorithm. To perform this step, the

coordinate of each point in the world coordinate and its correspondent pixel coordinate is required. Since the chessboard pattern has been defined with known dimensions, coordinates of each point in the real world are known. To find the pixel coordinate of each point as mentioned earlier as a feature detection algorithm, such as corner detection, can be used.

- In the next ste, the homography matrix between the calibration target plane and the image planes is used to find a closed-form solution of intrinsic parameters. After finding the intrinsic parameters, extrinsic parameters for each image can be calculated through algebraic operation.
- Then calculated initial guesses for intrinsic and extrinsic parameters from the previous step, with neglecting effect of distortion is used as an initial guess for a nonlinear least square minimization to refine and find the MLE of parameters.
- In next step a linear closed-form solution based on ideal pixel coordinates and distorted pixel coordinates is used for the estimation of the radial distortion coefficients. These values are used in the next step as an initial guess for distortion parameters.
- In last step a complete maximum likelihood estimation is performed using initial guesses from the previous steps to refine all parameters including lens distortion coefficients.

Details of this method will be described in subsections 2.3.2.2 to 2.3.2.5, which have been taken from Zhang's original paper (Zhang Z. , 1999) unless it is mentioned otherwise.

2.3.2.2. Closed Form Solution

According to Zhang (Zhang Z. , 1999), since the technique uses a 2-D planar pattern for calibration, without losing generality it can be assumed that the pattern plane is located at the $Z = 0$ world coordinate plane. Using this assumption from equation (2-10) for each point on image plane yields:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \boldsymbol{A}\,[\boldsymbol{r_1} \quad \boldsymbol{r_2} \quad \boldsymbol{r_3} \quad \boldsymbol{t}] \begin{bmatrix} X \\ Y \\ 0 \\ 1 \end{bmatrix} \equiv \boldsymbol{A}[\boldsymbol{r_1} \quad \boldsymbol{r_2} \quad \boldsymbol{t}] \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} \tag{2-14}$$

where

$$\boldsymbol{A} = \begin{bmatrix} \alpha & c & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

and $\boldsymbol{r_i}$ denotes to i^{th} column of rotation matrix in extrinsic parameters of equation (2-10) and $\boldsymbol{t}$ denotes to translation vector of extrinsic parameters:

$$[\boldsymbol{r_1} \quad \boldsymbol{r_2} \quad \boldsymbol{r_3} \quad \boldsymbol{t}] = [\boldsymbol{R_c^T} \quad -\boldsymbol{R_c^T t_c}]$$

Therefore, any corner point, such as $\widetilde{M}$ on calibration target and its image correspondence $\widetilde{m}$, are related by a 3×3 homography matrix $\boldsymbol{H} = [\boldsymbol{h_1} \quad \boldsymbol{h_2} \quad \boldsymbol{h_3}]$:

$$\widetilde{m} = \boldsymbol{H}\widetilde{M} \tag{2-15}$$

Let's assume that the homography between the image plane and the calibration target plane is known. This means that equation (2-15) can be rewritten as follows:

$$[\boldsymbol{h_1} \quad \boldsymbol{h_2} \quad \boldsymbol{h_3}] = \lambda \boldsymbol{A}[\boldsymbol{r_1} \quad \boldsymbol{r_2} \quad \boldsymbol{t}] \tag{2-16}$$

where λ is an arbitrary scalar and $\boldsymbol{h_i}$ is the i^{th} column of the homography matrix. Because $\boldsymbol{r_1}$ and $\boldsymbol{r_2}$ are orthonormal by definition, from equation (2-16) it yields:

$$\boldsymbol{h_1^T\,(A^{-1})^T\,A^{-1}h_2 = 0} \tag{2-17}$$

$$\boldsymbol{h_1^T(A^{-1})^TA^{-1}h_1 = h_2^TA^{-T}A^{-1}h_2} \tag{2-18}$$

Since the homography matrix has 8 degrees of freedom and extrinsic parameters include 6 unknowns (3 rotation and 3 translation parameters), each homography provides only two constraints to find intrinsic parameters. Therefore, if homographies for n images (with $n \geq 3$) are known, there will be 6 or more constraints that can be used for the estimation of intrinsic parameters through solving a linear system as follows.

By defining $\boldsymbol{B}$ as:

$$\boldsymbol{B = (A^{-1})^TA^{-1}} = \begin{bmatrix} B_{11} & B_{12} & B_{13} \\ B_{21} & B_{22} & B_{23} \\ B_{31} & B_{32} & B_{33} \end{bmatrix}$$

$$= \begin{bmatrix} \frac{1}{\alpha^2} & -\frac{c}{\alpha^2\beta} & \frac{cv_0 - u_0\beta}{\alpha^2\beta} \\ -\frac{c}{\alpha^2\beta} & \frac{c^2}{\alpha^2\beta^2} + \frac{1}{\beta^2} & -\frac{c(cv_0 - u_0\beta)}{\alpha^2\beta^2} - \frac{v_0}{\beta^2} \\ \frac{cv_0 - u_0\beta}{\alpha^2\beta} & -\frac{c(cv_0 - u_0\beta)}{\alpha^2\beta^2} - \frac{v_0}{\beta^2} & \frac{(cv_0 - u_0\beta)^2}{\alpha^2\beta^2} + \frac{v_0^2}{\beta^2} + 1 \end{bmatrix} \tag{2-19}$$

and considering the symmetric nature of $\boldsymbol{B}$ a 6-D vector, such as $\boldsymbol{b}$, can be defined as follows:

$$\boldsymbol{b} = [B_{11}, B_{12}, B_{22}, B_{13}, B_{23}, B_{33}]^T \tag{2-20}$$

Considering $\boldsymbol{h_i} = [h_{i1}, h_{i2}, h_{i3}]^T$ as the i^{th}column of homography matrix $\boldsymbol{H}$, it is known that:

$$\boldsymbol{h_i^T B h_j = v_{ij}^T b} \tag{2-21}$$

where $\boldsymbol{v_{ij}} = \left[h_{i1}h_{j1}, hi_1h_{j2} + h_{i2}h_{j1}, h_{i2}h_{j2}, h_{i3}h_{j1} + h_{i1}h_{j3}, h_{i3}h_{j2} + h_{i2}h_{j3}, h_{i3}h_{j3}\right]^T$

Using the above notation, constraints of equations (2-17) and (2-18) can be rewritten in the homogenous form as follows:

$$\begin{bmatrix} \boldsymbol{v_{12}^T} \\ (\boldsymbol{v_{11} - v_{22}})^{\boldsymbol{T}} \end{bmatrix} \boldsymbol{b} = \boldsymbol{0} \tag{2-22}$$

Having n images of the calibration target plane, using the above equation following the linear system can be formed by stacking them:

$$\boldsymbol{Vb = 0} \tag{2-23}$$

In the above equation $\boldsymbol{V}$ is a $2n \times 6$ matrix ($n \geq 3$). The solution for **b** is the eigenvector of $\boldsymbol{V^T V}$, which is related to its smallest eigenvalue. After estimating $\boldsymbol{b}$, camera intrinsic and extrinsic parameters can be obtained through further algebraic operations. Details of that can be found in Zhang's original work elsewhere (Zhang Z. , 1999).

2.3.2.3. Refining intrinsic and extrinsic parameters

The closed-form solution from the previous section can be used as an initial guess to refine intrinsic and extrinsic parameters. This can be achieved through maximum likelihood analysis by minimizing this function:

$$\sum_{i=1}^{n}\sum_{j=1}^{m}\left|\left|\boldsymbol{m}_{ij} - \hat{\boldsymbol{m}}(\boldsymbol{A}, \vec{\boldsymbol{R}}_{\boldsymbol{i}}, \boldsymbol{t_i}, \boldsymbol{M_j})\right|\right|^2 \tag{2-24}$$

In the above equation $\hat{\boldsymbol{m}}\left(\boldsymbol{A}, \vec{\boldsymbol{R}}_{\boldsymbol{i}}, \boldsymbol{t_i}, \boldsymbol{M_j}\right)$ is the pixel coordinate of point j in i^{th} image calculated by the projection model of equation (2-14). Finding suitable values for camera intrinsic parameters ($\boldsymbol{A}$), camera relative Rodrigues' rotation and translation vectors

($\vec{\boldsymbol{R}}_{\boldsymbol{i}}, \boldsymbol{t}_{\boldsymbol{i}}$) for each image that can minimize the result of equation (2-24) has been performed by using the Levenberg-Marquardt algorithm. It should be noted that without having a good initial guess as a result of the closed-from solution, achieving stable convergence on the minimization of equation (2-24) is very difficult.

2.3.2.4. Finding Radial Distortion Coefficients

Radial distortion of equation (2-11) can be rewritten in following matrix form:

$$\begin{bmatrix} (u-u_0)(x^2+y^2) & (u-u_0)(x^2+y^2)^2 \\ (v-v_0)(x^2+y^2) & (v-v_0)(x^2+y^2)^2 \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \end{bmatrix} = \begin{bmatrix} \delta_{x,Rad} \\ \delta_{y,Rad} \end{bmatrix} \quad (2\text{-}25)$$

In above equation, u and v are the ideal position of each pixel (without distortion), which is calculated through the projection model of equation (2-14). Having m corresponding points in n images, $2mn$ equations can be built by stacking up equation (2-25) for each pixel. Assuming $\boldsymbol{k} = [k_1, k_2]^T$, this can be expressed in a matrix form as follows:

$$\boldsymbol{Dk} = \boldsymbol{d} \quad (2\text{-}26)$$

This is a linear least-squares problem that has an exact solution:

$$\boldsymbol{k} = (\boldsymbol{D^T D})^{-1} \boldsymbol{D^T d} \quad (2\text{-}27)$$

2.3.2.5. Complete Maximum Likelihood Estimation

By having an initial guess for radial distortion coefficients, the maximum likelihood of all parameters can be achieved through minimization of the following function:

$$\sum_{i=1}^{n} \sum_{j=1}^{m} \left|\left| \boldsymbol{m}_{ij} - \hat{\boldsymbol{m}}(\boldsymbol{M_j}, \boldsymbol{A}, \vec{\boldsymbol{R}}_{\boldsymbol{i}}, \boldsymbol{t_i}, k_1, k_2, p_1, p_2) \right|\right|^2 \quad (2\text{-}28)$$

Table 2.1.

Table 2.2. List of camera characteristic parameters as a result of camera calibration process

Horizontal Focal length	p_1 Tangential Distortion Coefficient
Vertical Focal Length	p_2 Tangential Distortion Coefficient
Horizontal Principal Point Coordinate	k_1 Radial Distortion Coefficient
Principal Point Vertical Coordinate	k_2 Radial Distortion Coefficient

Again, the Levenberg-Marquardt algorithm has been used for minimization with the initial guess provided by the values that have been calculated in previous sections. It should be should be noted that the Levenberg-Marquardt algorithm can also provide an estimation of uncertainty for each parameter. Details of that process will be discussed later in this chapter.

2.3.3. Calibration Parameters

Using the calibration method, combined with the mathematical foundation of the Levenberg-Marquardt algorithm, can provide a set of parameters and their uncertainties for camera characteristics that can completely describe the projection model of the camera. A list of these parameters can be found in 1.1.1. It should be noted that, as a common practice, all parameters have units in a way that the final result of the projection model can be calculated in the unit pixel.

2.4. Planar Scene Reconstruction

2.4.1. Closed Form Initial Guess

Since the focus of the current study is distance measurement on a planar scene, by using similar notation as discussed earlier in Zhang's method, although having both intrinsic

and extrinsic parameters (or pose) of camera, coordinates of a 3-D point from pixel coordinates on the image plane can be determined. Although as it has been discussed earlier, camera calibration techniques, such as Zhang's method, need to estimate the camera pose for a general optimization of camera intrinsic and distortion parameters, it is safe to assume that, these parameters are not subject to considerable change as long as the camera settings do not change. Therefore in practice, for photogrammetry applications and 3-D reconstruction once the intrinsic and distortion parameters of a camera have been estimated, they can be used as independent constants in the camera projection model over different scenes.

As with Zhang's approach, without the loss of generality, it can be assumed that the planar scene is positioned at $Z = 0$. The solution starts by dropping the nonlinearity terms due to lens distortion to achieve a closed form solution as an initial guess for further maximum likelihood estimation.

Suppose the camera pose is known, using equation (2-14) and mathematical manipulation, the 3-D coordinates of a point in the world frame can be estimated from its image plane pixel coordinates:

$$\boldsymbol{M} = \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} = (\boldsymbol{A}[\boldsymbol{r_1} \quad \boldsymbol{r_2} \quad \boldsymbol{t}])^{-1} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \boldsymbol{C^{-1}m} \tag{2-29}$$

2.4.2. Maximum Likelihood Estimation for Back-Projection

Equation (2-29) does not take into account the nonlinearity due to lens distortion. The maximum likelihood of that point in the world frame can be refined by minimizing the following function with a closed form solution as an initial guess:

$$\left|\left|\boldsymbol{m} - \hat{\boldsymbol{m}}(\boldsymbol{C}, k_1, k_2, p_1, p_2, \boldsymbol{M})\right|\right|^2 \tag{2-30}$$

where in above equation the camera matrix $\boldsymbol{C}$ and distortion coefficients are assumed constant and the only variable is $\boldsymbol{M}$ or coordinates of the point in world frame. In the current study, the Levenberg-Marquardt algorithm has been used for the minimization of equation (2-30).

2.5. Pose Estimation

2.5.1. Overview

As already has been mentioned, 3-D reconstruction requires both intrinsic and extrinsic parameters of the camera. While intrinsic parameters of the camera can safely be assumed to be independent of the scene, the relative pose should be calculated for each image individually. The process of finding this pose, known as pose estimation, is defined as "*the determination of camera position and orientation from known correspondences of 3-D reference points and their images*" (Quan & Zhongdan, 1999). In mathematical terms, this means finding the right hand side of equation (2-7) in a way that all reference points in the real world are projected at their respective correspondences in the image plane.

This problem of single-image pose estimation is as old as photogrammetry itself and numerous pose estimation methods or algorithms can be found in scientific literature.

Theoretically, given the coordinates of three reference points in a world frame and their correspondences on an image plane; a solution (though possibly ambiguous) can be found. Analytical solutions for this problem go back to Grunert in 1841 (Haralick, Lee, Ottenberg, & & Nölle, Review and analysis of solutions of the three point perspective pose estimation problem., 1994).

Any pose estimation method requires some knowledge about the scene, which is called the reference here. In terms of reference type, two approaches are common in formulating the pose estimation problem (Ying & Zha, 2005):

- The first approach, known as Perspective-n-Point (PnP), determines the camera pose from a given set of 3-D.
- The second approach, known as Perspective-n-Line (PnL), employs a set of 3-D lines to solve the pose estimation problem.

Common methods for pose estimation are using point-correspondences (Shapiro & Stockman, 2001).

On the other hand, with respect to the solution method, pose estimation methods can be classified under two categories (Phong, Horaud, Yassine, & Tao, 1995):

- Closed-form solutions: whenever a limited number of point correspondences are present, an analytical solution is practicable. Examples of such solutions include methods based on three points (Fischler & Bolles., 1981), four coplanar points (Hung, Yeh, & Harwood, 1985), and three non-parallel lines (Dhome, Lab. d"Electron., Richetin, Lapreste, & Rives, 1989).

- Numerical and iterative solutions: when the number of points is large, an analytical solution requires solving a system of nonlinear equations with a number of equations greater than the number of unknowns, which is not practical and efficient.

Closed-form solutions are mostly suitable for real-time applications due to their speed, but they cannot easily take advantage from all available data in the case of an overdetermined problem. Therefore in current study, an iterative approach has been developed for pose estimation. In next sections a brief overview of some of PnP and PnL pose estimation algorithms is provided.

2.5.2. PnP Pose Estimation Algorithms

Given a set of known points in the world frame and their correspondences on the image plane, the pose can be extracted. Nister (Nistér, 2004) proposed a solution for the five points pose estimation problem by computing the coefficients of a tenths degree polynomial in closed form and then finding its roots. He compared the performance of the 5-point algorithm with 6-, 7- and 8-point algorithms and showed that under certain circumstances this algorithm worked while others were failing.

Quan et al (Quan & Zhongdan, 1999) introduced an algorithm for pose estimation based on correspondences of 3-D reference points and their image points. The algorithm can work with 5 or more points for extracting the pose and can provide solutions for coplanar configurations.

Schweighofer et al (Schweighofer & Axel, 2006) presented their algorithm for pose estimation based on four coplanar but not collinear points. They investigated the

ambiguity problem due to the existence of two local minima in pose estimation and addressed this problem by acquiring a unique and robust solution.

2.5.3. Studied PnL Pose Estimation Algorithms

Chen (Chen, 1990) introduced an analytical solution for pose estimation from line-to-plane correspondences. He showed that given unit vectors for three non-parallel lines on a plane and the unit vector of that plane solution for the pose can be devised analytically.

Heuel et al (Heuel & Förstner., 2001) developed an algorithm for matching 2-D line segments in multiple views and reconstruction of 3-D line segments from these matches. They employed homogenous coordinates for 2-D and 3-D geometric entities and developed a formulation for the propagation of uncertainty under a linear projection model.

Liu et al (Liu, Zhu, Jin, & Xia, 2012) developed an algorithm for determining the pose using a one point feature and two coplanar line features and their correspondences on the image plane. They employed geometric analysis to acquire a closed form solution.

Mirzaei et al (Mirzaei & Roumeliotis, 2011) proposed an approach by directly solving the corresponding least square problem algebraically from a system of polynomial equations for line correspondences. They stated that their algorithm guaranteed to find the globally optimal estimate and does not require initialization. They demonstrated that their algorithm is superior in terms of computational performance in comparison to previous approaches.

2.5.4. Limitations of Conventional Pose Estimation Algorithms

Having all six elements of pose that can provide the exact location of each point in the world frame is necessary for cases such as cartography, navigation in 3-D world and etc. However, this over-conclusive information might not be essential for certain planar photogrammetry applications. In accident scene reconstruction, measuring relative distances between objects on the scene is the objective and it can be mathematically proven that relative distances are indifferent to rotations along an axis perpendicular to the scene plane and translations on that plane.

On the other hand, as has been discussed earlier, the required information to make either a PnP or PnL pose estimation algorithm work, might be hard to collect. In the case of a PnP algorithm, it is required to collect X,Y and Z elements of coordinates of reference points, which in turn require accurate surveying equipment and techniques such as a combination of theodolite and distance measurement tools (measurement tape, etc…) or a total station. Setting up such a device on an accident scene might be difficult and is certainly time consuming. Using any PnL pose estimation algorithm, at least requires having information about the relative direction of lines (or unit vector) and/or coordinates of line segment endpoints. Besides the difficulty of collecting such information that might not be necessary to measure relative distance between objects on the scene, any 2-D measurement with a single device could lead to accumulative error. This problem arises from the inevitability of having one shared reference point such as the world frame origin among all references.

The other shortcomings in conventional pose estimation algorithms comes from neglecting or simplifying the nonlinearities due to lens distortion. As it will be shown in

the following sections, calculating the uncertainty in the final measurement of distances requires an unbroken chain of uncertainty propagation from camera intrinsic and lens distortion parameters to the final measurements. However, as mentioned earlier, in several modern camera calibration techniques, such as Zhang's method, the lens distortion has been incorporated in the final optimization process, the same thing cannot be stated about independent pose estimation algorithms with given camera intrinsic and lens distortion parameters. The common approaches work with the assumption of a separate step to produce an undistorted image and then using that image for pose estimation. In the scientific literature very little can be found on the incorporation of the lens distortion effect in the pure pose estimation problem definition. One example of such work is that of Josephson (Josephson & Byrod, 2009), which introduced a method to estimate the position, rotation, focal length and radial distortion from a minimal set of correspondences to a 3-D model. However, this method requires a considerable number of points in the world frame and their correspondences in the image plane.

To summarize the limitation:

- Popular pose estimation algorithms provide all six degrees of freedom of the pose matrix, which are not necessarily required for distance measurement on a planar scene.
- They require 3-D coordinates or unit vectors of reference points or lines on the scene. However, it is hard to collect that type of information.
- They usually do not take into account various nonlinearities in the model, which are needed for the estimation of uncertainty propagation.

2.5.5. Pose Estimation Algorithm

To overcome the limitations that have been discussed in the previous section and to make the approach more practical, a new pose estimation algorithm has been developed in this study that works with only three or more pairwise non-parallel line segments with known lengths. These line segments can be located across the scene with or without intersection and no given knowledge about their direction vectors or end points is required to estimate the pose.

This approach provides an unprecedented level of flexibility in providing reference for pose estimation since line segments with known lengths can be easily marked without any requirement of sophisticated equipment. Moreover, instances of such segments can be readily extracted from existing features in an accident scene such as road surface markings, sidewalk curbs, vehicle length, etc.

Because no similar approach has been found in literature, first it is necessary to prove that the resulting estimated pose can preserve line lengths in scene reconstruction of equation (2-29) and accordingly equation (2-30). To achieve the proof first, it is assumed that the undistortion process naturalize the non-linearity from the projection model.

2.5.6. Proof of Preserving Line Segment Lengths with Proposed Algorithm

Theorem: If an estimated pose back-projects three coplanar pairwise non-parallel lines in a way that it preserves their lengths on back projection, it will preserve the length of any other line on the same plane.

Proof: Suppose three coplanar line segments, such as L_1, L_2 and $L_{3,}$ accordingly with unknown coordinates in the world frame for start points at $P_{1,1}$, $P_{2,1}$ and $P_{3,1}$ and end

points at $P_{1,2}$, $P_{2,2}$ and $P_{3,2}$. Assuming they have been projected on the image plane with the known camera intrinsic matrix $\boldsymbol{A}$ and unknown actual pose $[\boldsymbol{r_1}\ \boldsymbol{r_2}\ \boldsymbol{t}]$. Assume a pose, such as $[\boldsymbol{r'_1}\ \boldsymbol{r'_2}\ \boldsymbol{t'}]$ is estimated, that back-projects those lines through equation (2-29) in a way that preserves the length of all three lines.

Starting with equation (2-14) for the first line and using actual pose results:

$$\begin{bmatrix} u_{1,1} \\ v_{1,1} \\ 1 \end{bmatrix} = \boldsymbol{A}[\boldsymbol{r_1} \quad \boldsymbol{r_2} \quad \boldsymbol{t}] \begin{bmatrix} X_{1,1} \\ Y_{1,1} \\ 1 \end{bmatrix} \tag{2-31}$$

$$\begin{bmatrix} u_{1,2} \\ v_{1,2} \\ 1 \end{bmatrix} = \boldsymbol{A}[\boldsymbol{r_1} \quad \boldsymbol{r_2} \quad \boldsymbol{t}] \begin{bmatrix} X_{1,2} \\ Y_{1,2} \\ 1 \end{bmatrix} \tag{2-32}$$

Since back-projection uses the same pixel coordinates in the image plane, similarly for the estimated pose it yields:

$$\begin{bmatrix} u_{1,1} \\ v_{1,1} \\ 1 \end{bmatrix} = \boldsymbol{A}[\boldsymbol{r'_1} \quad \boldsymbol{r'_2} \quad \boldsymbol{t'}] \begin{bmatrix} X'_{1,1} \\ Y'_{1,1} \\ 1 \end{bmatrix} \tag{2-33}$$

$$\begin{bmatrix} u_{1,2} \\ v_{1,2} \\ 1 \end{bmatrix} = \boldsymbol{A}[\boldsymbol{r'_1} \quad \boldsymbol{r'_2} \quad \boldsymbol{t'}] \begin{bmatrix} X'_{1,2} \\ Y'_{1,2} \\ 1 \end{bmatrix} \tag{2-34}$$

Because horizontal and vertical focal lengths cannot be zero, $\det[\boldsymbol{A}] = \alpha\beta \neq 0$, equations (2-31) to (2-34) can be combined by multiplying them to $\boldsymbol{A^{-1}}$ as follows:

$$[\boldsymbol{r_1} \quad \boldsymbol{r_2} \quad \boldsymbol{t}] \begin{bmatrix} X_{1,1} \\ Y_{1,1} \\ 1 \end{bmatrix} = [\boldsymbol{r'_1} \quad \boldsymbol{r'_2} \quad \boldsymbol{t'}] \begin{bmatrix} X'_{1,1} \\ Y'_{1,1} \\ 1 \end{bmatrix} \tag{2-35}$$

$$[\boldsymbol{r_1} \quad \boldsymbol{r_2} \quad \boldsymbol{t}] \begin{bmatrix} X_{1,2} \\ Y_{1,2} \\ 1 \end{bmatrix} = [\boldsymbol{r'_1} \quad \boldsymbol{r'_2} \quad \boldsymbol{t'}] \begin{bmatrix} X'_{1,2} \\ Y'_{1,2} \\ 1 \end{bmatrix} \tag{2-36}$$

Introducing vector representations V_1 and V_1' for line segment L_1 and the back projection from estimated pose L_1' yields:

$$\boldsymbol{V}_1 = \begin{bmatrix} X_{1,2} \\ Y_{1,2} \end{bmatrix} - \begin{bmatrix} X_{1,1} \\ Y_{1,1} \end{bmatrix} = \begin{bmatrix} V_{1,x} \\ V_{1,y} \end{bmatrix} \tag{2-37}$$

$$\boldsymbol{V_1'} = \begin{bmatrix} X'_{1,2} \\ Y'_{1,2} \end{bmatrix} - \begin{bmatrix} X'_{1,2} \\ Y'_{1,2} \end{bmatrix} = \begin{bmatrix} V'_{1,x} \\ V'_{1,y} \end{bmatrix} \tag{2-38}$$

Reshaped actual and estimated pose in their matrix format:

$$[\boldsymbol{r_1} \quad \boldsymbol{r_2} \quad \boldsymbol{t}] = \begin{bmatrix} r_{11} & r_{12} & t_1 \\ r_{21} & r_{22} & t_2 \\ r_{31} & r_{32} & t_3 \end{bmatrix} \tag{2-39}$$

$$[\boldsymbol{r_1'} \quad \boldsymbol{r_2'} \quad \boldsymbol{t'}] = \begin{bmatrix} r'_{11} & r'_{12} & t'_1 \\ r'_{21} & r'_{22} & t'_2 \\ r'_{31} & r'_{32} & t'_3 \end{bmatrix} \tag{2-40}$$

Substituting the above notations into equations (2-35) and (2-36) with some mathematical manipulations yield:

$$\overbrace{\begin{bmatrix} r_{11} & r_{12} \\ r_{21} & r_{21} \end{bmatrix}}^{\boldsymbol{R}_{2D}} \boldsymbol{V_1} = \overbrace{\begin{bmatrix} r'_{11} & r'_{12} \\ r'_{21} & r'_{21} \end{bmatrix}}^{\boldsymbol{R}'_{2D}} \boldsymbol{V_1'} \tag{2-41}$$

Similarly for lines L_2 and L_3:

$$\begin{bmatrix} r_{11} & r_{12} \\ r_{21} & r_{21} \end{bmatrix} \boldsymbol{V_2} = \begin{bmatrix} r'_{11} & r'_{12} \\ r'_{21} & r'_{21} \end{bmatrix} \boldsymbol{V_2'} \tag{2-42}$$

$$\begin{bmatrix} r_{11} & r_{12} \\ r_{21} & r_{21} \end{bmatrix} \boldsymbol{V_3} = \begin{bmatrix} r'_{11} & r'_{12} \\ r'_{21} & r'_{21} \end{bmatrix} \boldsymbol{V_3'} \tag{2-43}$$

It should be noted that $\boldsymbol{R}_{2D}$ and $\boldsymbol{R}'_{2D}$ represent a 2-D rotation matrix with an axis perpendicular to the plane containing all three lines.

Considering the fact that the lines and accordingly their vector representations are assumed to be pairwise non-parallel, vector $\boldsymbol{V_3}$ that is associated with line L_3 can be formulated as the linear combination of $\boldsymbol{V_1}$ and $\boldsymbol{V_2}$:

$$\boldsymbol{V_3} = a \cdot \boldsymbol{V_1} + b \cdot \boldsymbol{V_2} \tag{2-44}$$

Combining equations (2-41) to (2-43) with some mathematical manipulation reveals that $\boldsymbol{V_3'}$ can be reconstructed by same scalars as a and b from $\boldsymbol{V_1'}$ and $\boldsymbol{V_2'}$:

$$\boldsymbol{V_3'} = a \cdot \boldsymbol{V_1'} + b \cdot \boldsymbol{V_2'} \tag{2-45}$$

It is assumed that the estimated pose preserves the lengths of the third line in back-projection, therefore:

$$|\boldsymbol{V_3'}| = |\boldsymbol{V_3}| \tag{2-46}$$

or

$$\begin{aligned} &\left(aV_{1,x} + bV_{2,x}\right)^2 + \left(aV_{1,y} + bV_{2,y}\right)^2 \\ &\qquad = \left(aV'_{1,x} + bV'_{2,x}\right)^2 + \left(aV'_{1,y} + bV'_{2,y}\right)^2 \end{aligned} \tag{2-47}$$

In view of $|\boldsymbol{V_1'}| = |\boldsymbol{V_1}|$ and $|\boldsymbol{V_2'}| = |\boldsymbol{V_2}|$, it can be deduced:

$$\left(V_{1,x}\right)^2 + \left(V_{1,y}\right)^2 = \left(V'_{1,x}\right)^2 + \left(V'_{1,y}\right)^2 \tag{2-48}$$

$$\left(V_{2,x}\right)^2 + \left(V_{2,y}\right)^2 = \left(V'_{2,x}\right)^2 + \left(V'_{2,y}\right)^2 \tag{2-49}$$

Substituting (2-48) and (2-49) into (2-47) yields:

$$V_{1,x}V_{2,x} + V_{1,y}V_{2,y} = V'_{1,x}V'_{2,x} + V'_{1,y}V'_{2,y} \tag{2-50}$$

Similarly, any arbitrary line and its vector representation, such as $\boldsymbol{V}$, can be devised from two of the non-parallel reference vectors, for example from $\boldsymbol{V_1}$ and $\boldsymbol{V_2}$:

$$\boldsymbol{V} = a' \cdot \boldsymbol{V_1} + b' \cdot \boldsymbol{V_2} \tag{2-51}$$

Similar to equation (2-45) for its back-projection with estimated pose it can be stated that:

$$\boldsymbol{V'} = a' \cdot \boldsymbol{V_1'} + b' \cdot \boldsymbol{V_2'} \tag{2-52}$$

Hence the length of $\boldsymbol{V}$ is equal to**:**

$$\begin{aligned} |\boldsymbol{V}| &= \left(a' V_{1,x} + b' V_{2,x}\right)^2 + \left(a' V_{1,y} + b'^{V_{2,y}}\right)^2 \\ &= a'\left(V_{1,x}^2 + V_{1,y}^2\right) + b'\left(V_{2,x}^2 + V_{2,y}^2\right) \\ &\quad + 2a'b'\left(V_{1,x}V_{2,x} + V_{1,y}V_{2,y}\right) \end{aligned} \tag{2-53}$$

And the length of its back projection $\boldsymbol{V'}$ with estimated pose is equal to**:**

$$\begin{aligned} |\boldsymbol{V'}| &= a'\left(\left(V_{1,x}'\right)^2 + \left(V_{1,y}'\right)^2\right) + b'\left(\left(V_{2,x}'\right)^2 + \left(V_{2,y}'\right)^2\right) \\ &\quad + 2a'b'\left(V_{1,x}'V_{2,x}' + V_{1,y}'V_{2,y}'\right) \end{aligned} \tag{2-54}$$

Having in mind (2-48) to (2-50) and comparing (2-53) and (2-54) yields:

$$|\boldsymbol{V}| = |\boldsymbol{V'}| \tag{2-55}$$

Or back-projection of any line on the same plane with this estimated pose has the same length as the original line.

2.5.7. Maximum Likelihood Estimation for Pose

In the previous section, it has been proved that if an estimated pose preserves the lengths of three coplanar non-parallel lines in back-projection, it will preserve lengths of any other line on the same plane while neglecting distortion.

Let's assume that n line segments such as $l_1, l_2 \dots l_n$ on image plane with known end pixel coordinates $\boldsymbol{m}_{i,1} = \left[u_{i,1}\ v_{i,1}\right]^T$ and $\boldsymbol{m}_{i,2=}\left[u_{i,2}\ v_{i,2}\right]^T$ are given. Suppose that these

lines on the image plane are associated with n coplanar reference line segments with known lengths, such as $L_1, L_2 \dots L_n$, in the world frame. Having a pose, such as $[\boldsymbol{r_1} \quad \boldsymbol{r_2} \quad \boldsymbol{t}]$ and given camera intrinsic and lens distortion parameters, it is possible to back-project the end points of image plane lines to the world frame from the minimization of equation (2-30):

$$\boldsymbol{M}_{i,1} = \begin{bmatrix} X_{i,1} \\ Y_{i,1} \end{bmatrix} = \underset{\boldsymbol{M}}{\operatorname{argmin}} \left|\left| \boldsymbol{m}_{i,1} - \widehat{\boldsymbol{m}}(\boldsymbol{C}, k_1, k_2, p_1, p_2, \boldsymbol{M}) \right|\right|^{\mathbf{2}} \tag{2-56}$$

$$\boldsymbol{M}_{i,2} = \begin{bmatrix} X_{i,2} \\ Y_{i,2} \end{bmatrix} = \underset{\boldsymbol{M}}{\operatorname{argmin}} \left|\left| \boldsymbol{m}_{i,\mathbf{2}} - \widehat{\boldsymbol{m}}(\boldsymbol{C}, k_1, k_2, p_1, p_2, \boldsymbol{M}) \right|\right|^{\mathbf{2}} \tag{2-57}$$

Subsequently, the length of the back-projected line can be easily calculated from:

$$\widehat{\boldsymbol{L}}_i(\boldsymbol{C}, k_1, k_2, p_1, p_2, \boldsymbol{M}) = \sqrt{\left(X_{i,2} - X_{i,1}\right)^{\mathbf{2}} + \left(Y_{i,2} - Y_{i,1}\right)^{\mathbf{2}}} \tag{2-58}$$

Now, the pose can be found through the minimization of following error function:

$$\sum_{i=1}^{n} \left(\frac{(L_i - \widehat{L}_i(l_i, \boldsymbol{A}, [\boldsymbol{r_1} \quad \boldsymbol{r_2} \quad \boldsymbol{t}], k_1, k_2, p_1, p_2)}{L_i} \right)^2 \tag{2-59}$$

Since no origin has been introduced so far and line lengths are invariant to translations on the same plane, to improve the minimization convergence of equation (2-59) and avoiding infinite drifting in translation elements of pose, without losing generality the starting point of the first line can be chosen as the origin. In this case, the error function becomes:

$$\left(X_{1,1}\right)^2 + \left(Y_{1,1}\right)^2 + \sum_{i=1}^{n} \left(\frac{(L_i - \widehat{L}_i(l_i, \boldsymbol{A}, [\boldsymbol{r_1} \quad \boldsymbol{r_2} \quad \boldsymbol{t}], k_1, k_2, p_1, p_2)}{L_i} \right)^2 \tag{2-60}$$

It should be noted that the error for each line has been normalized by its original length. This is necessary, otherwise long lines on the scene will become dominant in the minimization process and a considerable level of accuracy for smaller lines and subsequently for the whole pose estimation process will be lost.

2.6. Least Square Methods and Maximum Likelihood Estimation

2.6.1. Overview

At several points in the previous sections, to find the maximum likelihood estimation of parameters, minimization of a least square error function has been mentioned. The maximum likelihood estimation deals with estimating the parameters of a statistical model and one of common approaches in finding the maximum likelihood estimation is least squares.

In the following subsections, the numerical method for the minimization of a least square error will be briefly reviewed and the formulation for calculating uncertainty in model parameters and model predictions will be discussed. Later, the general formulation will be applied to the problems in hand and a comprehensive model for the prediction of uncertainty in distance measurement for coplanar photogrammetry will be devised.

2.6.2. Definitions

Assume having n observations $(\boldsymbol{X_i}, y_i), i = 1,2, \dots, n$ from a phenomenon that follows a nonlinear mathematical model with a known functional relationship f (Seber & Wild, 2003):

$$y_i = f(\boldsymbol{X_i}, \boldsymbol{\theta}^*) + \epsilon_i \tag{2-61}$$

where the likelihood of ϵ_i is zero ($E[\epsilon_i] = 0$), $\boldsymbol{X_i}$ is a $m \times 1$ vector in the real domain and the true value for model parameters $\boldsymbol{\theta}^*$($\left[\theta_1^* \, \theta_2^* \, ... \, \theta_p^*\right]^T$) is known to belong to $\boldsymbol{\Theta}$, a subset of $\mathbb{R}^p$. The least square estimate of model parameters $\boldsymbol{\theta}^*$ minimizes the error of the sum of squares:

$$S(\boldsymbol{\theta}) = \sum_{i=1}^{n} [y_i - f(\boldsymbol{X}_i, \boldsymbol{\theta})]^2 \equiv \sum_{i=1}^{n} \left(r_i(\boldsymbol{\theta})\right)^2 \tag{2-62}$$

It should be noted that the nonlinear least-square, unlike the linear least-square, might possess more than one relative minima other than absolute minimum $\widehat{\boldsymbol{\theta}}$. If the joint distribution of ϵ_i is known, then the maximum likelihood estimate of $\boldsymbol{\theta}$ is obtained by minimizing the likelihood function. If ϵ_i is normally distributed, by having a few other regulatory assumptions, it can be proved that $\widehat{\boldsymbol{\theta}}$ is also the maximum likelihood estimator (Seber & Wild, 2003).

In the present study, it is assumed that there is no systematic error. Moreover, it is assumed that all errors are normally distributed with a mean equal to zero. Therefore, the maximum likelihood estimation is equivalent to the least square minimization.

2.6.3. Gauss-Newton Algorithm

The usual approach to obtain the least square method is the Gauss-Newton algorithm. The algorithm is an iterative algorithm that starts with an initial guess and proceeds forward by calculating a correction term until achieving a satisfactory convergence.

It is known that at the minimum, partial derivatives of S by elements of model parameters ($\theta_1, \theta_2, \ldots, \theta_p$) is zero, or (Wolberg, 2006):

$$\frac{\partial S}{\partial \theta_k} = 0 \quad k = 1 \; to \; p \tag{2-63}$$

Starting with an initial guess for the iteration step s, such as $\boldsymbol{\theta}^s = \left[\theta_1^s \; \theta_2^s \; ... \; \theta_p^s\right]^T$; for true value $\boldsymbol{\theta}^*$, incorporating (2-63) into (2-62) and some mathematical manipulations yield:

$$\sum_{i=1}^{n} f(\boldsymbol{X_i}, \boldsymbol{\theta}^*) \frac{\partial f(\boldsymbol{X_i}, \boldsymbol{\theta}^*)}{\partial \theta_k^*} = \sum_{i=1}^{n} y_i \frac{\partial f(\boldsymbol{X_i}, \boldsymbol{\theta}^*)}{\partial \theta_k^*} \qquad k = 1 \; to \; p \tag{2-64}$$

Using Taylor series $f(\boldsymbol{X}_i, \boldsymbol{\theta}^*)$ (which will be called $f_i(\boldsymbol{\theta}^*)$ from now on) can be replaced with its first order approximation in the small neighbourhood of its true value $\boldsymbol{\theta}^*$:

$$f_i(\boldsymbol{\theta}^s) = f_i(\boldsymbol{\theta}^*) + \sum_{k=1}^{p} \frac{\partial f(\boldsymbol{X_i}, \boldsymbol{\theta}^*)}{\partial \theta_k} (\theta_k^s - \theta_k^*) \tag{2-65}$$

To calculate the correction term for the next iteration step, it is assumed that $\boldsymbol{\theta}^{s+1} = \boldsymbol{\theta}^*$. Replacing (2-65) in (2-64), further linearization and mathematical manipulations yield the correction term as follows:

$$\boldsymbol{\beta} = \boldsymbol{\theta}^s - \boldsymbol{\theta}^{s+1} = ((\boldsymbol{J}^s)^{\boldsymbol{T}} \boldsymbol{J}^s)^{-\mathbf{1}} (\boldsymbol{J}^s)^{\boldsymbol{T}} \boldsymbol{r}^s \tag{2-66}$$

where $\boldsymbol{J}$ is the $n \times p$ Jacobian matrix:

$$\boldsymbol{J}^s = \boldsymbol{J}(\boldsymbol{\theta}^s) = \begin{bmatrix} \frac{\partial r_1(\boldsymbol{\theta}^s)}{\partial \theta_1} & \cdots & \frac{\partial r_1(\boldsymbol{\theta}^s)}{\partial \theta_p} \\ \vdots & \ddots & \vdots \\ \frac{\partial r_n(\boldsymbol{\theta}^s)}{\partial \theta_1} & \cdots & \frac{\partial r_n(\boldsymbol{\theta}^s)}{\partial \theta_p} \end{bmatrix} \tag{2-67}$$

an $\boldsymbol{r}$ is $n \times 1$ the residual vector:

$$\boldsymbol{r}^s = \boldsymbol{r}(\boldsymbol{\theta}^s) = \begin{bmatrix} r_1(\boldsymbol{\theta}^s) \\ \vdots \\ r_n(\boldsymbol{\theta}^s) \end{bmatrix} \tag{2-68}$$

In the calculation of the correction term of equation (2-66) if the process is started from an initial guess that is far from the true value, the correction term might become so large and lead to numerical divergence. To avoid this, a damping ratio, such as λ, can be introduced to limit the correction term. In this case, the final form of the algorithm for the correction of $\boldsymbol{\theta}$ in an iterative manner becomes:

$$\boldsymbol{\theta}^{s+1} = \boldsymbol{\theta}^{s} - \lambda[(\boldsymbol{J}^{s})^{\boldsymbol{T}}\boldsymbol{J}^{s}]^{-\mathbf{1}}(\boldsymbol{J}^{s})^{\boldsymbol{T}}\boldsymbol{r}^{s} \tag{2-69}$$

2.6.4. Levenberg-Marquardt Algorithm

At an intermediate step in minimization by the Gauss-Newton algorithm, the Jacobian matrix can become the rank deficient and some difficulties in proceeding might arise. The first approach to overcome this problem is to introduce second derivatives in the Taylor expansion of equation (2-65). Adding second derivatives is expensive in terms of computation time and complexity. The second approach, known as the trust region method or Levenberg-Marquardt algorithm, is to further dampening the Gauss-Newton algorithm, which has been first suggested by Levenberg and further improved by Marquardt (Björck, 1996). . This can be achieved by adding a matrix that consists of diagonal elements of $(\boldsymbol{J}^{s})^{\boldsymbol{T}}\boldsymbol{J}^{s}$ to obtain the correction term:

$$\boldsymbol{\theta}^{s+1} = \boldsymbol{\theta}^{s} - [(\boldsymbol{J}^{s})^{\boldsymbol{T}}\boldsymbol{J}^{s} + \lambda \boldsymbol{diag}((\boldsymbol{J}^{s})^{\boldsymbol{T}}\boldsymbol{J}^{s})]^{-\mathbf{1}}(\boldsymbol{J}^{s})^{\boldsymbol{T}}\boldsymbol{r}^{s} \tag{2-70}$$

where $\boldsymbol{diag}(\boldsymbol{M})$ operator creates a matrix with the same size as $\boldsymbol{M}$ with zero on its non-diagonal elements.

Many different approaches can be found in scientific literature about how to choose the damping factor λ, but that discussion is out of the scope of current study.

2.6.5. Uncertainty in Model Parameters

Suppose that the Levenberg-Marquardt algorithm minimized the sum of square errors to some satisfactory final value such as S_f at certain final value of $\widehat{\boldsymbol{\theta}}(\hat{\theta}_1, \hat{\theta}_2, \dots, \hat{\theta}_p)$. The final variance of error ϵ between model predictions $f_i(\widehat{\boldsymbol{\theta}})$ and observations y_i can be calculated from the following equation:

$$\sigma_\epsilon^2 = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - f_i\left(\widehat{\boldsymbol{\theta}}\right)\right)^2 \tag{2-71}$$

Defining $\boldsymbol{C} = \boldsymbol{J}^{\boldsymbol{T}}\boldsymbol{J}$ the covariance matrix for uncertainty in model parameters is (Seber & Wild, 2003):

$$\boldsymbol{\Sigma}_{\boldsymbol{\theta}} = \sigma_\epsilon^2 \boldsymbol{C}^{-1} = \begin{bmatrix} \sigma_1^2 & \sigma_{12} & \cdots & \sigma_{1p} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{p1} & \sigma_{p1} & \cdots & \sigma_p^2 \end{bmatrix} \tag{2-72}$$

Therefore, the standard deviation for each estimated parameter, such as $\hat{\theta}_k$, can be extracted from diagonal elements of the above matrix. In other words, the value of $\hat{\theta}_k$ in relationship with true value θ_k^* follows this equation:

$$(\hat{\theta}_k - \theta_k^*) \sim \mathcal{N}(0, \sigma_k^2) \tag{2-73}$$

where $\mathcal{N}(0, \sigma_k^2)$ represents a normal distribution with a mean and standard deviation accordingly equal to zero and σ_i.

An approximation to calculate the final variance of error, such as $\sigma_\epsilon^2 \cong S_f/(n-p)$, can be found in statistical textbooks, which has not been used in the current study due to its requirement for having a large number of observations ($n \gg p$).

2.6.6. Uncertainty in Model Predictions

After calculating uncertainties in the model's parameters, those uncertainties can be propagated to model predictions through the following equation (Wolberg, 2006):

$$\sigma_f^2 = \sum_{k=1}^{p} \left(\frac{\partial f}{\partial \theta_k}\sigma_k\right)^2 + \sum_{j=1}^{j=p}\sum_{k=j+1}^{k=p} 2\left(\frac{\partial f}{\partial \theta_j}\frac{\partial f}{\partial \theta_k}\sigma_{jk}\right) \tag{2-74}$$

Dealing with equation (2-74) requires special care due to the existence of a covariance between the uncertainties of different parameters, which will be discussed in the following sections.

2.6.7. Calculation of Jacobian Matrix

Acquiring a solution for a least square problem requires the calculation of the Jacobian matrix of equation (2-67). When the number of parameters is small or a simple mathematical function is the subject of minimization, having an analytical representation for $\partial f_i/\partial \theta_j$ is not difficult. But this approach might not be feasible all the time, and the alternative solution that has been used in the current study is the calculation of partial derivatives by central difference:

$$\frac{\partial f(\theta_1, \dots, \theta_k, \dots, \theta_p)}{\partial \theta_k} \cong \frac{f(\theta_1, \dots, \theta_k + h, \dots, \theta_p) - f(\theta_1 \dots, \theta_k - h, \dots, \theta_p)}{2h} \quad (2\text{-}75)$$

2.6.8. Uncertainties in Camera Intrinsic and Lens Distortion Parameters

As it has been discussed earlier in Zhang's calibration method, n images each containing m reference points have been used for the final least square estimation of equation (2-28). Model parameters include 4 camera intrinsic parameters, 4 lens distortion parameters and $6 \times n$ extrinsic parameters of each image.

Emgu CV library, which has been used in current study for camera calibration, can provide the user with analytical derivatives of the projection model of equation (2-13). Derivatives are provided independently for horizontal and vertical pixel coordinates. For j^{th} point, such as $\boldsymbol{m}_j(u_{i,j}, v_{i,j})$ in i^{th} image a 14×2 subset of the Jacobian matrix $\boldsymbol{J_{i,j}}$ can be devised from those derivatives. Using that subset, the total Jacobian matrix $\boldsymbol{J_{2mn\times(8+6n)}}$ can be formed. This Jacobian matrix has been used through equation (2-72) for calculating uncertainty in estimated camera intrinsic and lens distortion parameters and the covariance matrix.

It should be noted that later observations are independent from extrinsic parameters of the images that have been used for calibration. Therefore, only an 8×8 covariance matrix (between camera intrinsic and lens distortion parameters) has been kept for further calculation of correlative uncertainties in model predictions from equation (2-74).

2.6.9. Uncertainty in Pose Estimation

In previous sections, n coplanar lines with known lengths have been used for the pose estimation. As mentioned earlier, the length of a line is invariant to translations along the scene plane and rotations around an axis perpendicular to that plane. Therefore, the number of applicable extrinsic parameters for the single view pose estimation is reduced from six to three. Equation (2-60) can be interpreted as $n + 2$ observations. Therefore, the Jacobian matrix for calculating pose uncertainty is a $(n + 2) \times 3$ matrix.

2.6.10. Uncertainty in Scene Reconstruction

Having uncertainties for camera intrinsic, lens distortion and pose from the previous section, the uncertainty in the back-projection of equation (2-30) can be easily calculated from equation (2-74). It should be noted that the covariance between the pose uncertainty and other parameters are assumed to be zero because they are the results of independent observations.

There are 16 parameters and their uncertainties including uncertainty in selected pixel involved in the calculation of each point coordinate. Therefore, 32 derivatives are needed for the calculation of the total uncertainty. The result of this analysis provides uncertainty for coordinates of each point in the world frame as follows:

$$\widehat{M} \sim \mathcal{N}\left(\begin{bmatrix}\hat{X}\\ \hat{Y}\end{bmatrix}, \begin{bmatrix}\sigma_x^2\\ \sigma_y^2\end{bmatrix}\right) \tag{2-76}$$

2.6.11. Uncertainty in Measured Lengths

Propagation of uncertainty for a multivariable function, such as $z = f(x_1, x_2, \dots, x_m)$, is given by:

$$\sigma_z^2 = \left(\frac{\partial f}{\partial x_1}\right)^2 \sigma_{x_1}^2 + \left(\frac{\partial f}{\partial x_2}\right)^2 \sigma_{x_2}^2 + \cdots + \left(\frac{\partial f}{\partial x_m}\right)^2 \sigma_{x_m}^2 \tag{2-77}$$

The length of a line segment as a function of its two end points coordinates is given as:

$$l = \sqrt{(X_2 - X_1)^2 + (Y_2 - Y_1)^2} \tag{2-78}$$

The derivatives of lengths by variation of its end point coordinates can be calculated from the following equation:

$$\left(\frac{\partial l}{\partial X}\right)^2 |_{X=X_1 \, or \, X_2} = \frac{(X_2 - X_1)^2}{(X_2 - X_1)^2 + (Y_2 - Y_1)^2} \tag{2-79}$$

$$\left(\frac{\partial l}{\partial Y}\right)^2 |_{Y=Y_1 \, or \, Y_2} = \frac{(Y_2 - Y_1)^2}{(X_2 - X_1)^2 + (Y_2 - Y_1)^2} \tag{2-80}$$

Therefore, the total uncertainty in line length can be calculated from:

$$\sigma_l^2 = \left(\sigma_{X_1}^2 + \sigma_{X_2}^2\right)\left(\frac{\partial l}{\partial X}\right)^2 + \left(\sigma_{Y_1}^2 + \sigma_{Y_2}^2\right)\left(\frac{\partial l}{\partial Y}\right)^2 \tag{2-81}$$

Since the pose estimation algorithm is based on coplanarity assumption, equation (2-81) can be expanded to take into account small perturbations in height such as ϵ_Z. Using the same notation as equation (2-81), the contribution of this perturbation to the total uncertainty can be calculated as follows:

$$\sigma_l^2 = \left(\sigma_{X_1}^2 + \sigma_{X_2}^2\right)\left(\frac{\partial l}{\partial X}\right)^2 + \left(\sigma_{Y_1}^2 + \sigma_{Y_2}^2\right)\left(\frac{\partial l}{\partial Y}\right)^2 + \frac{\epsilon_z^4}{l^2} \tag{2-82}$$

From equation (2-82) it can be concluded that the effect of the deviation from coplanarity assumption varies with the second order of the height to length ratio.

3. Implementation

3.1. Introduction

The previous chapter provided a mathematical model for calculating the parameters and their uncertainty. In this chapter, implementations of those equations will be discussed.

The overall steps for photogrammetry on a coplanar scene, such as an accident scene, can be described as follows:

- Camera calibration: this step determines the camera intrinsic and lens distortion parameters. Once these parameters are calculated, they can be used over different images that have been taken with the same camera.
- Pose estimation: given the camera parameters, in this step three or more lines with known length and their corresponding image points are used to find the camera relative position and look direction. Uncertainties in pose parameters are calculated at this step.
- By using the camera intrinsic and lens distortion parameters in combination with estimated pose, this step calculates the world frame distance between any two points (pixels) on the image.

The workflow of the developed photogrammetry computer application can be seen in Figure 3.1.

In the following sections, first an overview of computer application will be given, and then details of the implementation of each step will be discussed.

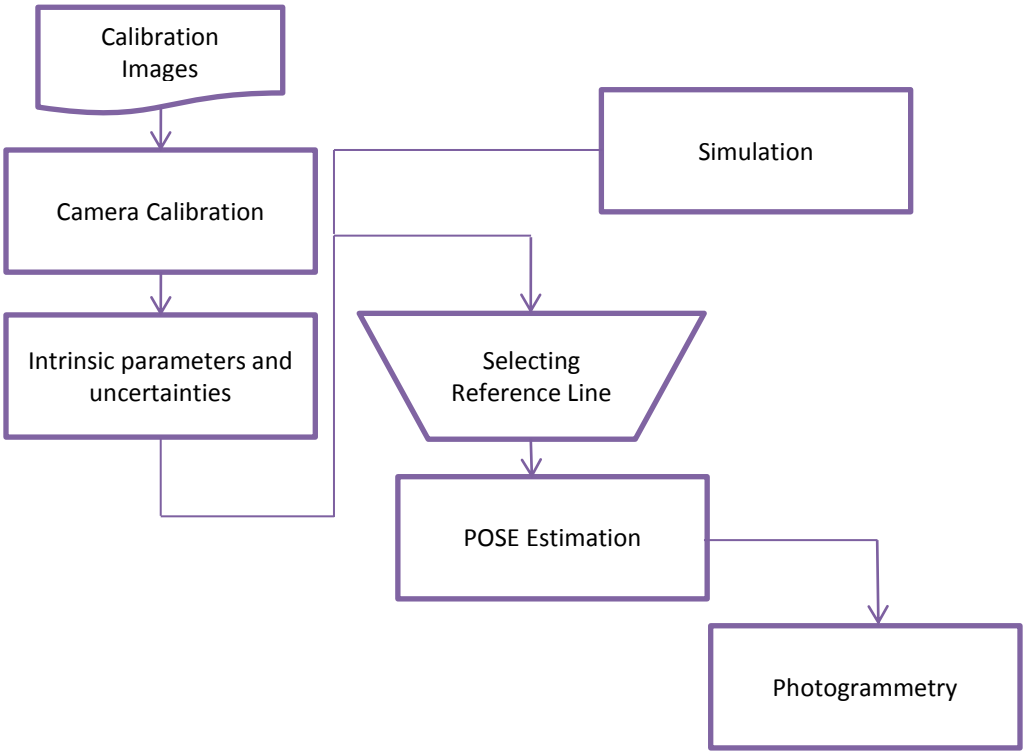

Figure 3.1. Workflow of developed coplanar photogrammetry application

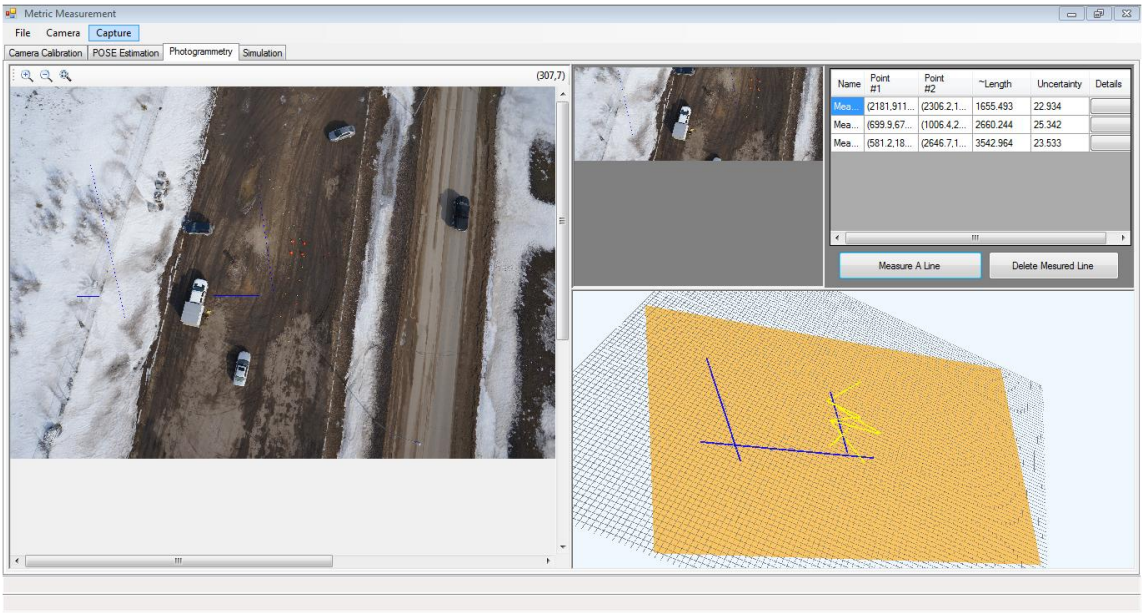

Figure 3.2. Snapshot of developed computer program GUI

3.2. Developed Computer Software

3.2.1. Overview

A computer application has been developed for the Windows 7-X64 operating system in the Visual Studio 2008 development environment as a Windows Form Application. Figure 3.2 shows a sample snapshot of a computer graphical user interface (GUI). To cover all required tasks in a single environment, the program GUI has three tabs for different steps of photogrammetry measurements. Moreover, a fourth tab has been added for research simulation purposes only. The simulation tab is not directly involved in single view coplanar photogrammetry. An overview of the functionality of each tab is discussed in the following sections.

3.2.2. Calibration Tab

The calibration tab allows the user to select a set of photos of a chessboard pattern for the calibration from a single folder, and then set the required parameters, such as number of squares in horizontal and vertical directions, and perform the calibration process. After calibration, estimated values and uncertainties for the camera intrinsic and lens distortion parameters can be seen on this tab.

Since an automated algorithm has been used for detecting the corners of a pattern, detected corners on each image are shown to the user as a second measure of verification. Options for saving and loading calibration results have also been implemented to ensure reusability of those parameters.

3.2.3. Pose Estimation Tab

In the pose estimation tab the user can select a single photo of a scene, mark lines with known lengths as reference elements for pose estimation based on the method that has been described in chapter 2 and perform pose estimation. Since this task involves precise pixel selection, a User Control has been developed that provides the user with a flexible ability to zoom and select at the sub-pixel level, navigate across an image and overlaying reference lines over the image.

After introducing three or more lines over the scene, the user can perform the pose estimation. If the pose estimation is found to be successful, the selected image is automatically transferred to the photogrammetry tab.

3.2.4. Photogrammetry Tab

In the photogrammetry tab by using same User Control for pixel selection, the user can measure the world frame distance between any two points (pixels) on the image. After selecting the second end point of the line segment of interest, measured distance, the total uncertainty in measured distance and components of uncertainty are shown to the user. Providing components of uncertainties enables the user to track down and verify the sources of uncertainty in measurements. This information might be used as a measure to decrease the uncertainty by reducing certain elements through, for example, repeating camera calibration or selecting different reference lines for pose estimation.

Moreover, a 3-D view of the scene, containing reference lines and measured lines, is shown on this tab to enhance visualization.

3.3. Numerical and Image Processing Libraries

3.3.1. Overview

Implementing the developed mathematical model into a computer application requires several image processing and numerical algorithms. Most of these algorithms are well-known and their implementations can be found in developed libraries that are available for general use.

To save development time, the current study, makes extensive use of two open-source libraries. The Emgu CV library has been used for camera calibration and image processing operations related to camera calibration and back projection. The ALGLIB library implementation of the Levenberg-Marquardt algorithm has been used for acquiring the least square estimation of parameters for the pose estimation and back-projection tasks. In the following sections, a brief review of these two libraries and the relevent tasks with their abilities will be provided.

3.3.2. Emgu CV

According to developers (Emgu CV, Main Page) "*Emgu CV is a cross platform .Net wrapper to the OpenCV image processing library. Allowing OpenCV functions to be called from .NET compatible languages such as C#, VB, VC++, IronPython etc. The wrapper can be compiled in Mono and run on Windows, Linux, Mac OS X, iPhone, iPad and Android devices.*" Emgu CV is completely associated with OpenCV and as stated by developers, acts only as a wrapper to that library, therefore both libraries are cited in coming paragraphs.

For the calibration task using a chessboard pattern, the following functionalities have been performed through Emgu CV:

- Conversion of color images to gray scale images for corner detection.
- Image resizing. Since there are some hardcoded parameters in the FindChessboardCorners method of OpenCV; it cannot handle large images for detecting a chessboard pattern (Guo, 2013). Therefore, it is necessary to scale the image down for chessboard pattern detection.
- Finding chessboard pattern corners in scaled down images and refining the corner sub-pixel locations in the original image.
- Camera Calibration. OpenCV includes a function for camera calibration with a built-in least-square optimizer using the Levenberg-Marquardt algorithm.
- Calculating derivatives to create the Jacobian Matrix, which is required for the estimation of uncertainties in camera intrinsic and lens distortion parameters.
- Projecting a 3-D point into the image plane having known camera intrinsic and lens distortion parameters. This function is required for back projection and subsequently for pose estimation.

3.3.3. ALGLIB

As it has been shown in the previous chapter, estimating the maximum likelihood for back-projection of equation (2-29) and pose estimation of equation (2-60) require a least square solver. Though OpenCV has its own the Levenberg-Marquardt least-square algorithm function implemented for camera calibration, that method is not accessible outside the calibration and it is highly customized for this specific task and related equations such as (2-24) and (2-28).

Therefore, the general purpose numerical and data analysis library "ALGLIB" has been used for back projection and pose estimation.

According to developers:

> ALGLIB is a cross-platform numerical analysis and data processing library. It supports several programming languages (C++, C#, Pascal, VBA) and several operating systems (Windows, Linux, Solaris). ALGLIB features include: Optimization and nonlinear solvers, Interpolation and linear/nonlinear least-squares fitting, Linear algebra, direct and iterative linear solvers, Fast Fourier Transform and many other algorithm (Bochkanov & Bystritsky, About ALGLIB).

The implemented Levenberg-Marquardt method of ALGLIB addresses problems similar to equation (2-62) and it can operate in three modes (Bochkanov & Bystritsky, Levenberg-Marquardt algorithm for multivariate optimization):

- **V** (function vector): only function vector is given and numerical differentiation and secant updates are used for the Jacobian calculation.
- **VJ** (vector and Jacobian): function vector and Jacobian are both provided by the user.
- **FGH** (function, gradient and Hessian): function vector, its gradient vector and the Hessian matrix are provided by the user.

For least square problems of back projection and pose estimation the Jacobian matrix is left to be calculated numerically by ALGLIB and "**V**" mode has been used. ALGLIB requires the user to define the difference step for calculating derivatives of equation (2-75).

ALGLIB provides the user with the capability to introduce initial guess, scale and bound for parameters that are under minimization. For back projection, the initial guess is provided by equation (2-29). For pose estimation, a scale for translation is devised by a simple cross-multiplication of scene size from the length of the longest reference line and 2π is selected as the scale for the rotation vector. No bound has been defined for either of the vectors.

3.4. Simulation Tool

To simulate and study error over an assumed scene, a simulation tool has been added to the developed program. Figure 3.3 shows the GUI for this simulation tool. The objective of this simulation is to investigate how absolute error in different parameters can affect the photogrammetric measurement accuracy. This can be done in four steps:

- Defining a scene composed of three reference lines and one line that is the subject of measurement. Reference lines are sides of a right triangle and the corner related to right angle is located at the origin. Legs of the triangle can be modified to change the scene. The line that is the subject of measurement can be defined by the coordinates of its endpoints.
- Defining camera position and look direction. Coordination of camera position including its height above this scene can be defined. Look direction can be defined with the coordinates of the cross point between the camera optical axis and the scene plane.
- Selecting the parameter of interest for error estimation, introducing the lower and upper bound of variation of that parameter and number of deviation across that range.

- Performing simulation.

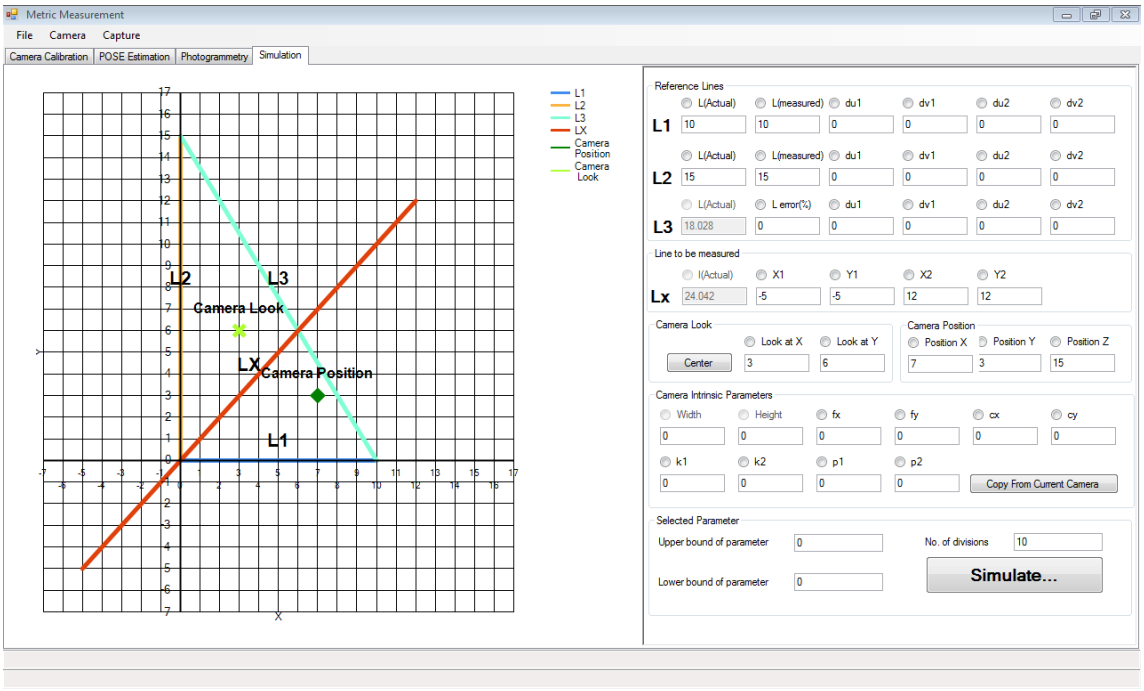

Figure 3.3. Graphical User Interface of simulation tool

Depending on the selected variable, two types of simulation can be performed. In the first type the value of a parameter can be changed across its range and the effect of that on accuracy can be studied. This is useful in instances when the nominal value of the parameter is important. For example, it is necessary to know how deviation from looking perpendicular to the image plane can affect the accuracy in general. In the second mode effect of deviation from the actual value of a parameter can be studied. For example, the simulation can be used for studying the case when the actual length of a reference line is 25 units of length, while the measured length for that line varies from 24 units to 26 units of length.

To perform the simulation, the tool uses the actual values of different parameters to project the scene on a virtual image plane. After this step, the value of the parameter of

interest changes across its bound and length of the line of interest is measured with this changed parameter and the error is measured.

This tool can provide an insight on how absolute error in value of parameters including reference lines can affect the absolute error in measurement. Moreover, since it uses the same procedure for photogrammetry measurement as the developed computer application, it is possible to investigate different hypotheses of error propagation.

3.5. Summary

In this chapter an overview of the methodology to implement the mathematical model and workflow for this process have been presented. The workflow has three steps consisting of camera calibration, pose estimation and photogrammetry. A computer application for the Windows 7 X64 operating system has been developed for carrying all those tasks of photogrammetry and overall structure of that including implemented libraries for image processing and numerical optimization have been discussed and presented. Moreover, a simulation tool has also been developed along with the photogrammetry workflow for the sole purpose of research and an overview of this simulation tool has also been presented in this chapter.

4. Results and Discussion

4.1. Overview

In previous chapters, the mathematical model and developed tool for applying the proposed method for coplanar photogrammetry have been presented. In this chapter, the results of that analysis, both for the developed simulation tool and a mock accident scene, will be presented.

The simulation provides a tool that can be used to investigate two types of effects. The first type is how the variation of a parameter can affect the accuracy of measurement. For example, how the variation of the height of the camera, or the length of reference lines can affect overall accuracy. The second type is studying how error in nominal value of a parameter can affect the accuracy.

For the experimental study, a simulated accident scene has been created and aerial photos of that accident scene have been used for photogrammetry measurement. As it will be shown in the following sections, the results of photogrammetry measurements have been compared with independent measurements that have been performed with a total station surveying equipment.

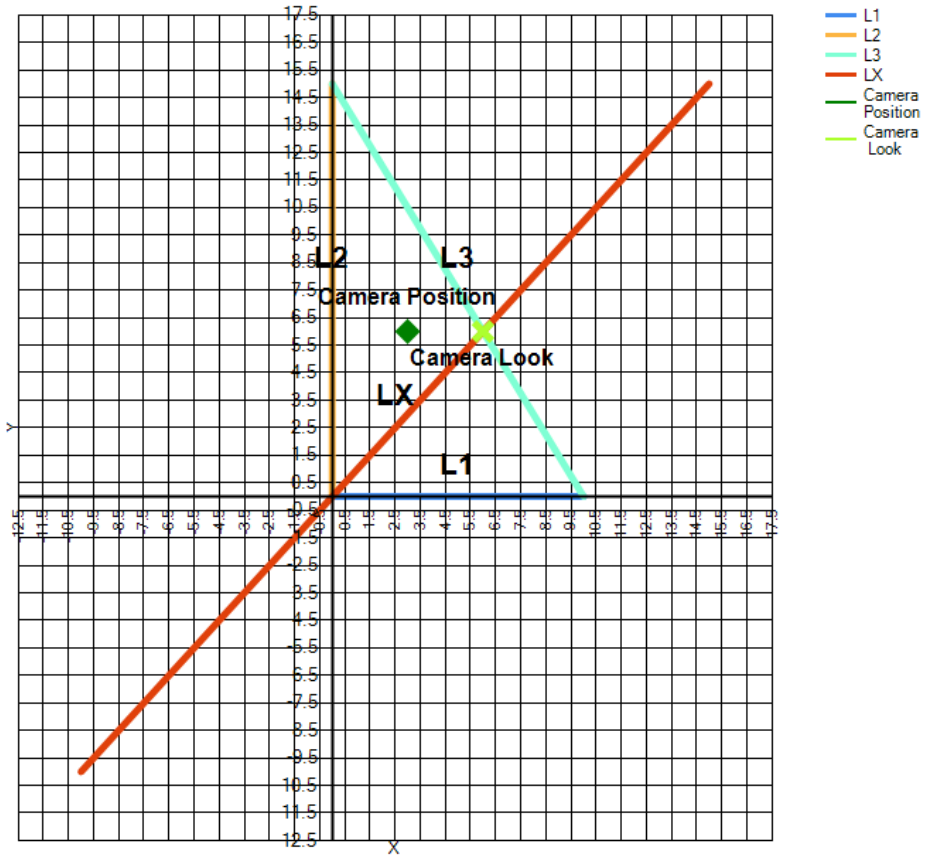

Figure 4.1. Simulated scene configuration

4.2. Simulation

4.2.1. Scene Configuration

Figure 4.1 shows an overview of the simulated scene configuration. The scene includes three reference line segments L1, L2 and L3 (respectively blue, orange and cyan lines), camera position (green diamond), camera looking target point (light green cross) and Lx, the line to be measured (red line).

Since the proposed pose finding algorithm works with three or more pairwise non-parallel line segments with known length, a right triangle has been used for introducing reference lines. The lengths of the legs can be modified to change their relative lengths

with respect to the whole scene. Two types of error can be induced for reference lines. The first type is selecting the wrong pixel location for end points. The second type is error in measuring the line segment length. Both errors are assumed to be common, since their related parameters are results of human observation.

4.2.2. Simulation Results

4.2.3. Simulation Scenarios

34 parameters are available for variation. Those include twelve parameters for pixel selection error in end points of three reference lines (horizontal and vertical), three for measured lengths of those lines, two for lengths of sides of right triangle, four for coordinates of end points of the line which is subject to measurement on scene plane, three for camera position, two for camera look point coordinates on scene plane, eight parameters for camera characteristics.

Since the variation of 34 parameters can provide an infinite number of scenarios for the simulation, ten typical scenarios were selected by selecting a few out of overlapping scenarios. For example, out of twelve parameters for introducing error in the pixel selection, two were chosen. These ten scenarios include two out of three for error in measured length of reference lines, two out of twelve for pixel selection error, two out of five for camera position and look direction and four out of eight for camera parameters were selected for the simulation.

Table 4.1 shows selected actual values for scene configuration and camera parameters. These parameters have been selected approximately equal to the parameters of the camera and the scene used for the experimental study.

Table 4.1. Actual values of parameters for simulation

Parameter	Value	Parameter	Value	Parameter	Value
L1 Length	10	Camera X	3	β	3370
L2 Length	15	Camera Y	6	u_0	2365
X1 (Lx)	-10	Camera Z	50	v_0	1500
Y1(Lx)	-10	Look X	6	k_1	-0.05
X2 (Lx)	15	Look Y	6	k_2	.13
Y2 (Lx)	15	α	3400	p_1	-0.009
Image Width	4912	Image Height	3264	p_2	-0.0046

4.2.4. Effect of Error in Reference Line Measured Length

As stated earlier, the length of reference lines are used for pose estimation. It is important to evaluate the outcome if the measured length of a reference line differs from its actual value, and how this error affects the photogrammetry measurement.

Figure 4.2 shows the effect of error in measuring L1 length over the measurement of Lx length. The actual value for this line is assumed to be 10 units of length and the measured value varied over a range of ±5% of actual value and this variation induced an error in range of (-4.5%,+5%) for photogrammetry measurement. This is the shortest reference line and the ratio of its length to line of interest is about 50% ($\frac{L1}{Lx} = \frac{10}{24.09} = 40\%$).

Figure 4.3 shows the same effect for error in measurement of the length of L2. As can be seen, induced error is larger than the induced error from inaccurate measurement of L1, which is expected because this line is 50% longer than L1. It should be noted that because the procedure of pose estimation uses a numerical approach and also pixels are

rounded to their nearest integer values, there is about 0.5% error in measurement even when there is no error in introduced reference length. Moreover, higher error values for the left side of the error curve of L1 can be attributed to additional constraints on that line in equation (2-60), since the beginning point of this line has been constrained to the origin.

From this analysis it can be deducted that any error in measuring the length of the reference lines can generate direct proportional error in photogrammetric measurement in the same order.

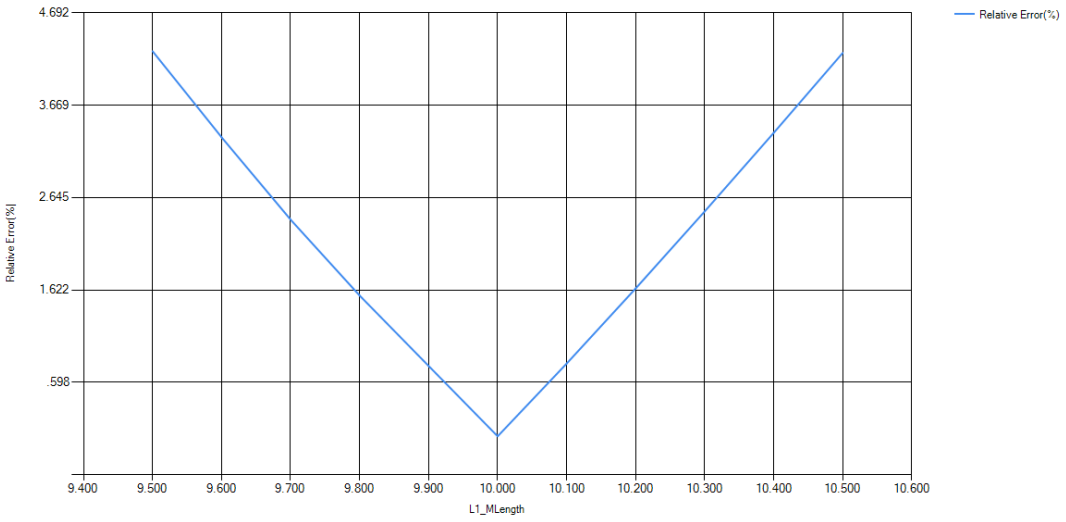

Figure 4.2. Effect of error in measured length of L1 on photogrammetry error

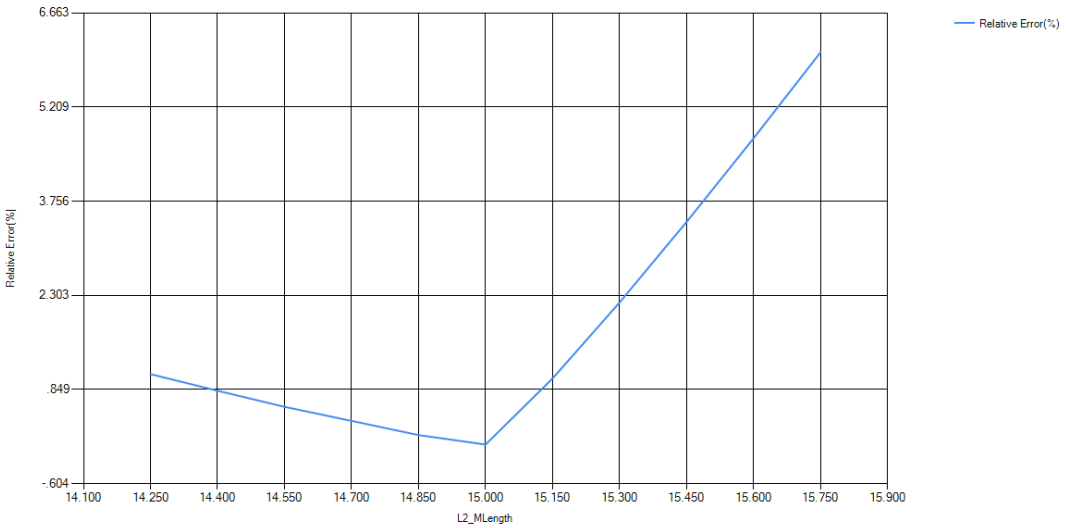

Figure 4.3. Effect of error in measured length of L2 on photogrammetry error

4.2.5. Effect of Error in Pixel Selection of Reference Lines

Another important source of error is pixel selection by the user. Since any image has a finite resolution, selecting the end points of the reference line to some extent depends on the judgment of the user. Based on the resolution and the type of marker that has been used on the actual scene, this process might induce few pixels error.

To analyze the contribution of this error to photogrammetry error, the first endpoint of the L1 reference line (located at the origin in Figure 4.1) has been subjected to pixel selection error in both horizontal and vertical directions.

Figure 4.4 shows the effect of pixel selection error for the first point of L1 in the horizontal direction on photogrammetry error. Since the camera look direction is perpendicular to the y axis of the scene, this error in pixel selection is along the length of L1. As can be seen about an error level of about 2% error resulted from a ±15 pixel selection error.

Figure 4.5 shows the effect of pixel selection error for the first point of L1 in the horizontal direction on photogrammetry error. As can be seen the induced error is found to be insignificant.

From this analysis it can be deducted that, pixel selection error along with the length of a reference line, can induce larger error in photogrammetry measurement than perpendicular direction.

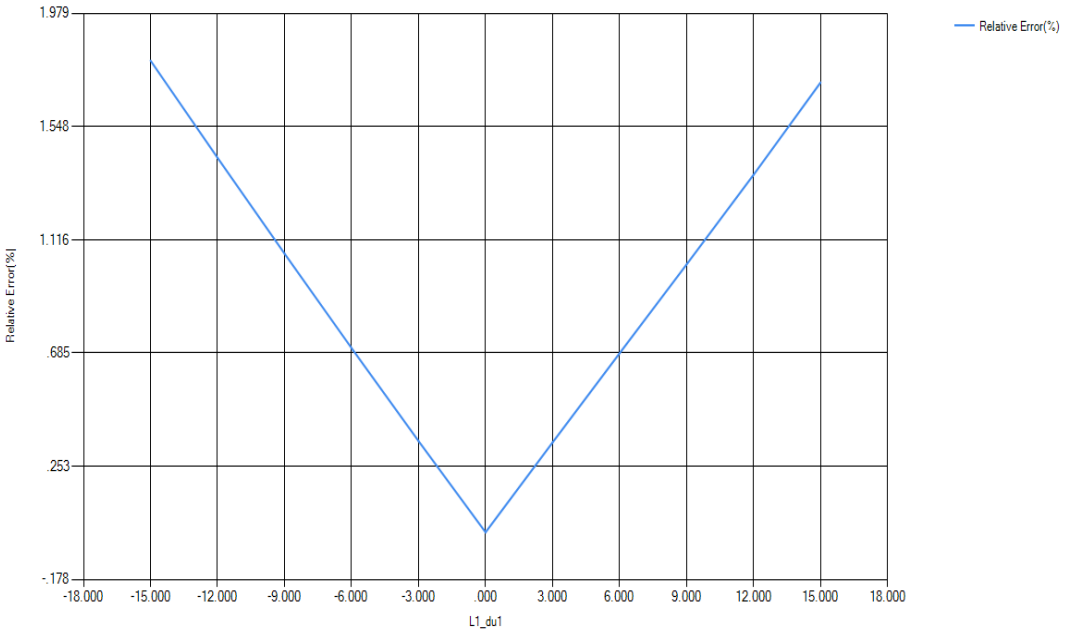

Figure 4.4. Photogrammetry error due to wrong pixel selection of first endpoint of line L1in horizontal direction

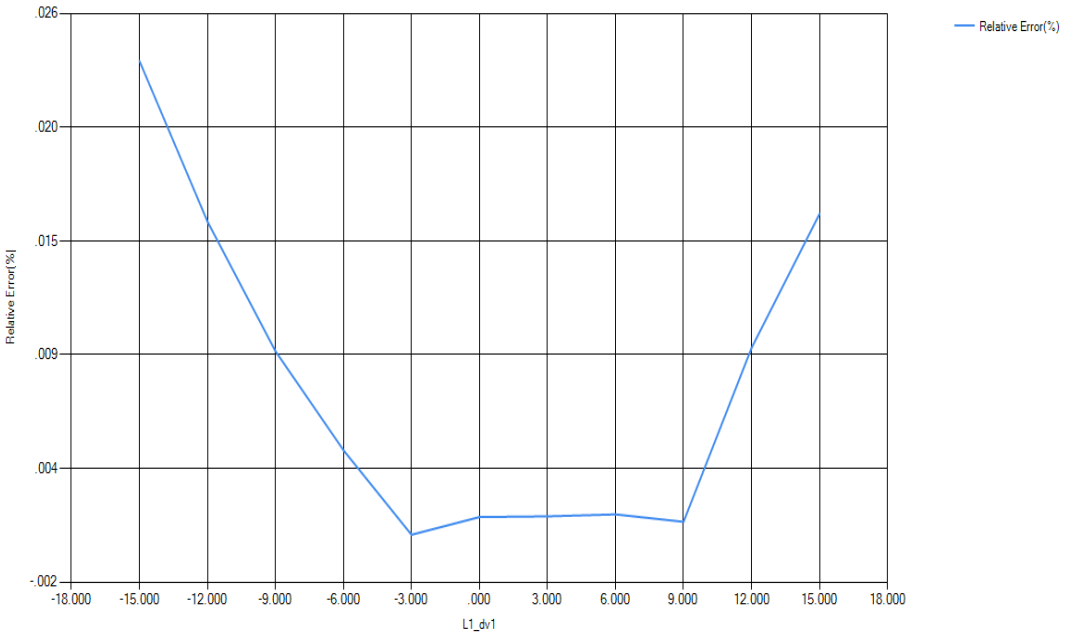

Figure 4.5. Photogrammetry error due to wrong pixel selection of first endpoint of line L1in vertical direction

4.2.6. Effect of Camera Height and Look Direction

Camera height and look direction can affect photogrammetry error in various ways. When the look direction diverges from being perpendicular to the scene plane, nonlinearity due to lens distortion varies considerably from one area to another. Moreover, because of this perspective, at points which are located far from the image center, resolution decreases dramatically. Camera height can also generate the same effect. When the camera gets further away from the scene each pixel represents a larger portion of the world frame.

To investigate the effect of look direction, while keeping the camera optical axis to pass at nearly the center of the scene, the camera position moved across the scene. In Figure 4.6 the green line denotes to the variation of camera position for simulating the effect of look direction. Since the camera look-at point was located at (6,6) and camera X

coordinate varied from -20 to 20, this variation is equivalent to varying the angle between camera optical axis and scene plane normal vector from -60º to +16º. As can be seen in Figure 4.7, photogrammetry error increases at higher angles as expected.

To investigate the effect of camera height above the scene, camera height varied from 30 units of length to 90 units of length. Figure 4.8 shows result of this simulation and it can be generally stated that increasing height can increase the photogrammetry error. It should be noted that due to quantization (rounding pixel values to nearest integer) there are some oscillations in error. This happened because, when the rounded value is closer to the actual value, it automatically results in less error.

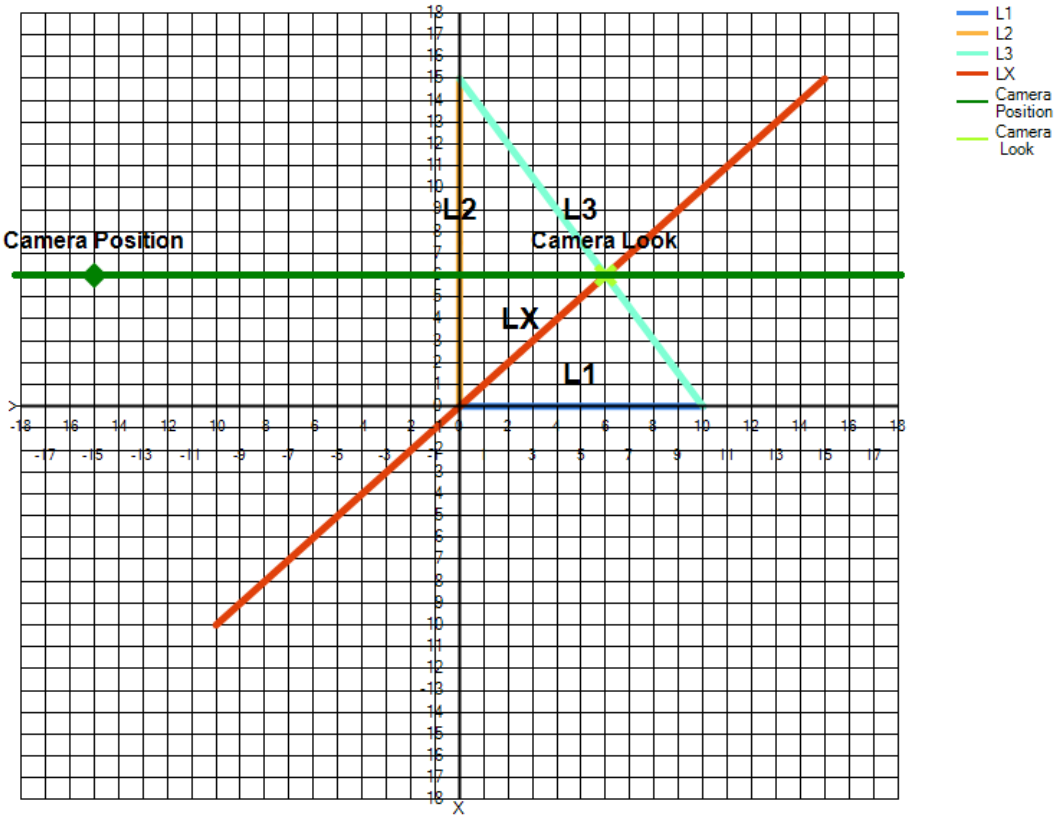

Figure 4.6. Scene arrangement for finding effect of camera look direction on photogrammetry error

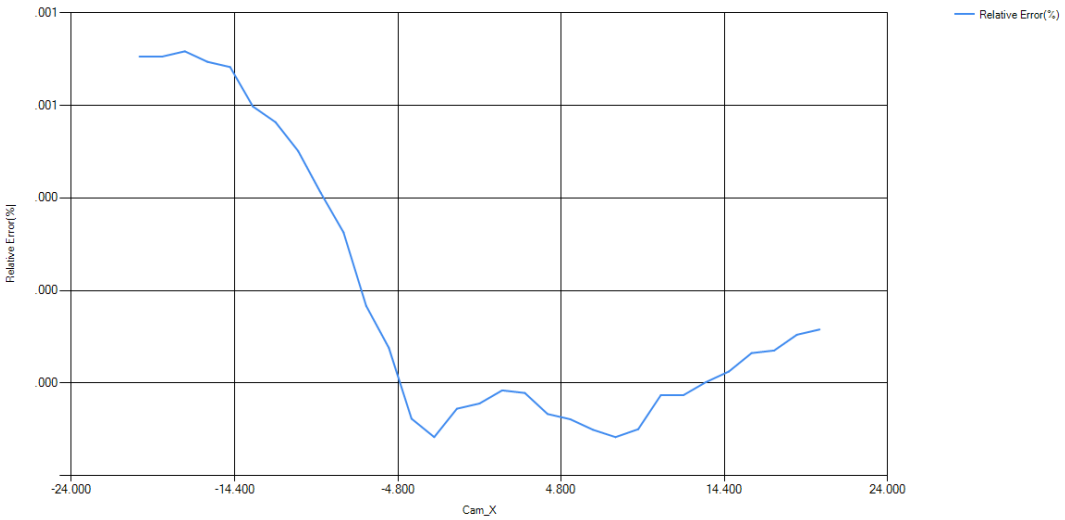

Figure 4.7. Variation of photogrammetry error due to change in camera look direction

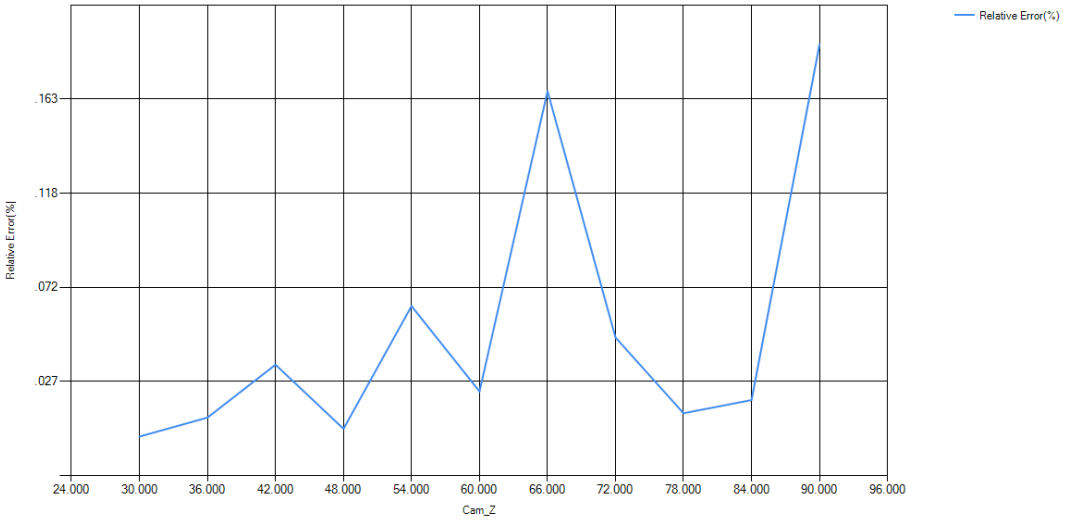

Figure 4.8. Effect of camera height on photogrammetry error

4.2.7. Camera Focal Length and Principal Point

Theoretically error in camera focal length can affect photogrammetry error since its value has been used directly in back-projection and finding the world frame coordinate. On the other hand, since in the current study the pose estimation algorithm tries to acquire a pose that preserves the length of reference lines, any error in focal length to some extent is compensated by adjusting the translation element of the pose. This means that, when

actual focal length is smaller than its nominal value, the pose finding algorithm comes up with a smaller estimation for the distance of the camera from scene plane. Figure 4.9 shows the direct relationship between error in horizontal focal length and photogrammetry error. The nominal value of focal length varied about 2% around its actual value. As can be seen due to compensation by pose estimation, the error is not significant.

On the other hand, location of principal point defines all nonlinearities due to lens distortion. Figure 4.10 shows variation of photogrammetry error due to the introduction of ±3% error to the vertical element of the principal point. As can be seen due to small distortion in this lens, error in the principal point has an insignificant effect on photogrammetry error.

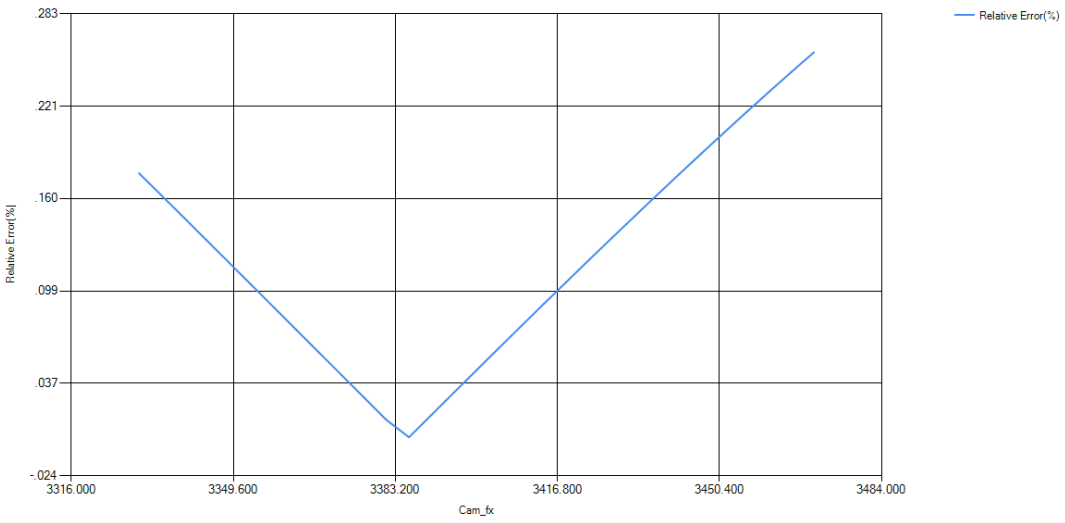

Figure 4.9. Effect of error in horizontal focal length on photogrammetry error

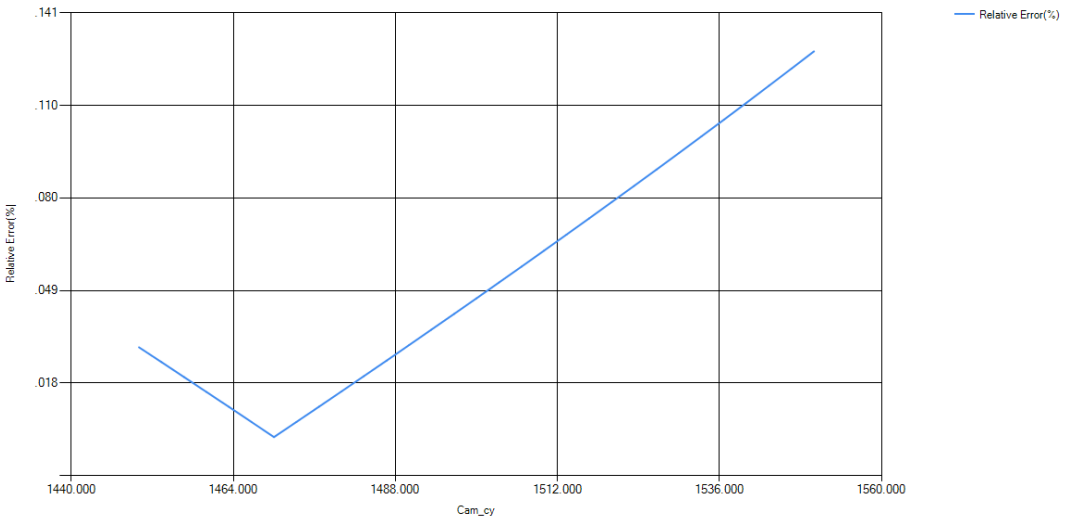

Figure 4.10. Effect of error in vertical element of focal length on photogrammetry error

4.3. Experiment

4.3.1. Overview

Because the focus of this study has been on traffic accident reconstruction, an experiment was conducted to measure the performance of the developed method and tool in this context.

A simulated accident scene was created at the University of Regina contractors' parking lot. Aerial photos of the simulated scene were taken using a quadrotor as the camera platform. Several distances over the scene were measured both using total station surveying equipment and the developed photogrammetry computer application.

4.3.2. Experimental Setup

The simulated accident scene represented a head-on collision between two cars. Skid marks of both front wheels of each vehicle were marked by small pylons (a different colour was used for each vehicle). Four corners of the impact area were also marked with

large orange traffic cones. The whole scene could be boxed in a ~14x26 meters square. The ground view of the simulated scene, including marked vehicles, can be seen in Figure 4.11. Moreover, to assist the photogrammetry analysis, six reference lines have been introduced into the scene.

The camera platform was a DraganFlyer X4-ES quadrotor. This quadrotor carries a SONY NEX-5R camera on a gyro stabilized mount. The camera has a resolution of 4912 x 3264 equivalent of ~16.1 megapixel (NEX-5R).

The quadrotor was flown over the scene around the impact area and several photos were taken for further photogrammetry measurement. Figure 4.11.b shows the relative position of the DraganFlyer above the scene from ground view.

To record distances between objects on the scene, a SOKKIA SRX Robotic total station has been used. Using a total station enabled recording the coordinates of marked points on the scene in three-dimensional space. Moreover, according to the manufacturer this surveying equipment can achieve accuracy of up to 1.5 mm in measurement (SRS Robotic Total Station).

4.3.3. Measurement with the Total Station

Forty four points have been measured with the total station, including:

- 1 fixed geographical reference point.
- 8 points of vehicles corners.
- 4 corners of impact area.
- 24 skid mark's points.
- 7 reference points for photogrammetry software.

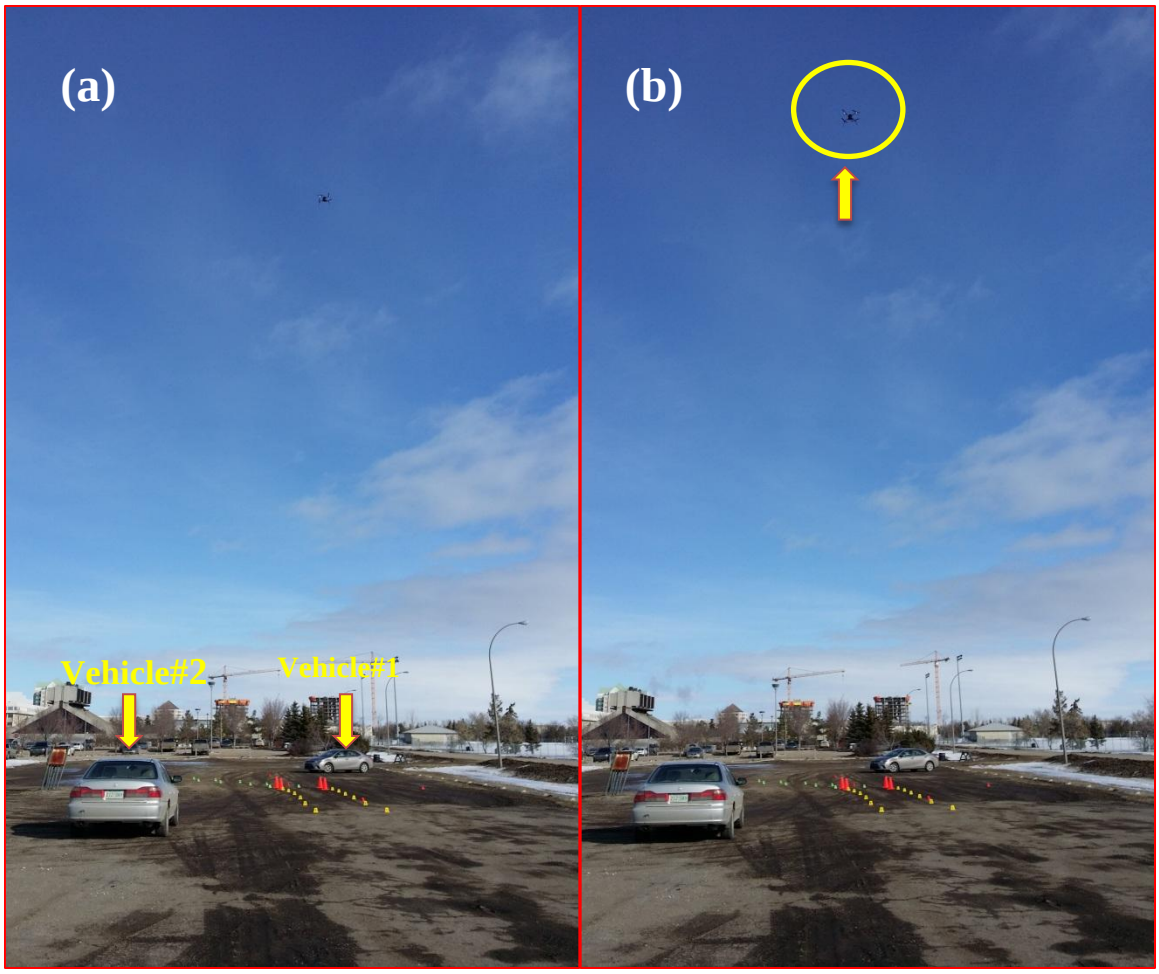

Figure 4.11. Simulated accident scene from ground view, (a) vehicles (b) DraganFlyer X4-ES

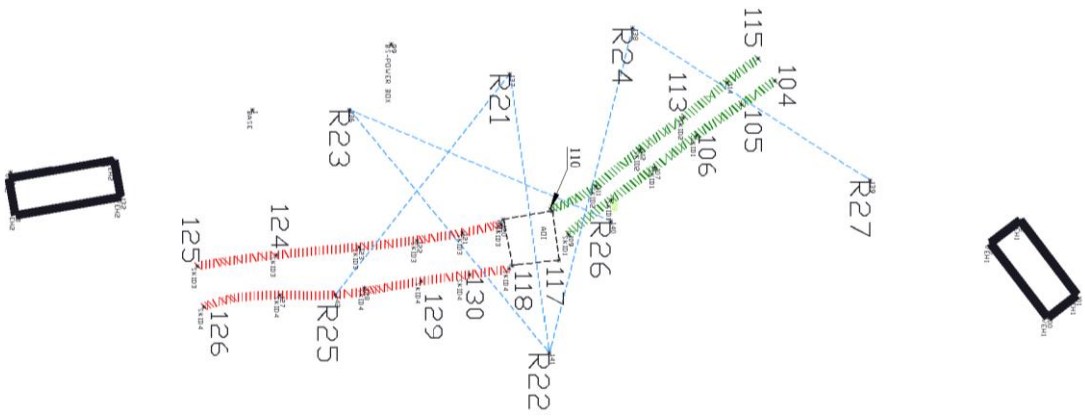

Figure 4.12. Schematic of measured points with total station surveying equipment

Table 4.2. Estimated values for intrinsic and lens distortion parameters for SONY NEX-5R camera of the experiment

Parameter	Value	σ
α	3343.66	11.63
β	3339.28	11.59
u_0	2397.69	7.493
v_0	1512.15	7.587
k_1	-00.07256	0.0023396
k_2	0. 1125	0.0041103
p_1	-00.004626	0.0004735
p_2	-00.0014386	0.0005156

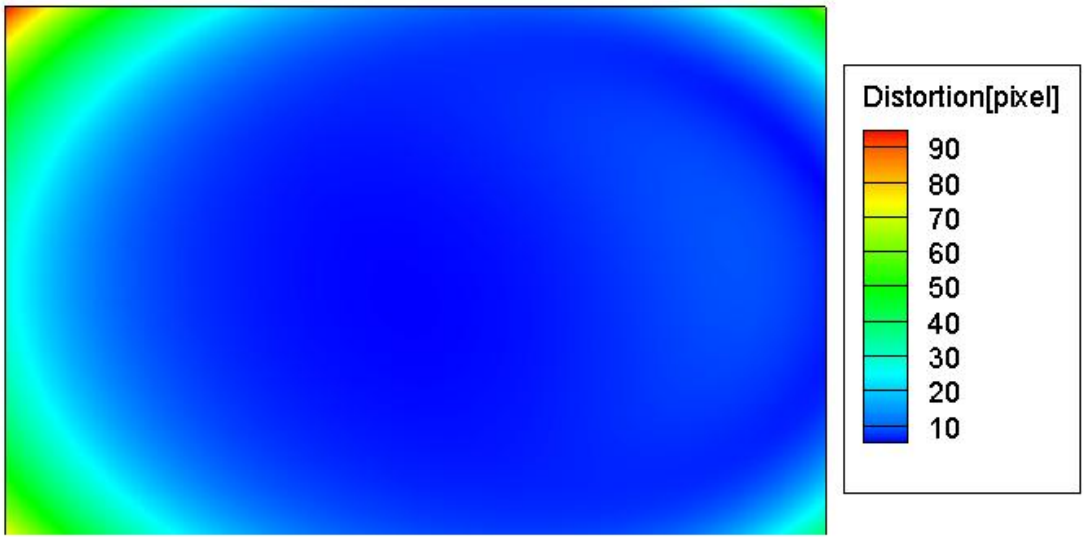

Figure 4.13. Lens distortion contours for camera which has been used in experiment

From collected data it appeared that the scene was not completely planar and there was about 30cm height difference between the lowest and highest points at the scene. The standard deviation for the height of measured points on the scene is found to be ~9 cm.

Figure 4.12 shows a schematic of the measured point with the total station. Red and green lines denote to tire skid marks of vehicle #1 and vehicle #2 respectively. Blue lines are reference lines that are added to the scene for pose finding in the photogrammetry process.

4.3.4. Camera Calibration

A set of 59 photos of an 8x5 square chessboard pattern has been used for estimating camera intrinsic and lens distortion parameters. Table 4.2 shows the estimated parameter for the camera that has been used for the experiment. A relatively large number of calibration photos helped to reduce uncertainties in camera intrinsic and lens distortion parameters.

The camera lens shows relatively low distortion values over a large portion of the image plane. Figure 4.13 shows the total distortion contours for this lens.

4.3.5. Pose Estimation and Photogrammetry Analysis

Two images that have been taken at two different angles have been selected for the photogrammetry measurement. The lengths of reference lines have been calculated from values provided by the total station measurement. Figure 4.14 shows selected photos for the photogrammetry analysis and overlaid reference line in light yellow. Six reference lines have been introduced for pose estimation.

Photo (a) has been taken with a relatively large angle between camera optical axis and scene plane normal vector while in photo (b) camera optical axis is almost perpendicular to the scene plane.

Table 4.3 shows the results of pose estimation. Average error in length of reference lines as a result of back projection with estimated pose for photo (a) and (b) is 0.92% and 0.31% respectively. Standard deviation of the same error for photos (a) and (b) is 0.56% and 0.17% accordingly. Higher errors in photo (a) can be attributed to its relative shallow

angle that resulted in larger error due to pixel quantization in areas far from the camera. For example, in photo (b) each pixel represent ~1.5 cm at the bottom and about ~3.2 cm at the top of the image in vertical direction, while in photo (a) at the bottom of the image each pixel is equivalent of 0.75 cm and at the top it represents ~40cm in the vertical direction. Moreover, the angle itself makes the accurate selection of the marker's pixels difficult.

However, the error values are found to be within an acceptable range for the purpose of this study.

Figure 4.14. Selected photos for photogrammetry analysis with overlaid reference lines

Table 4.3. Results of pose estimation for selected photos

Line#	Point 1	Point 2	Actual Length[cm]	Photo (a)		Photo (b)	
				Estimated Length[cm]	Error (%)	Estimated Length[cm]	Error (%)
1	R24	R27	1198	1210	1.00	1200	0.17
2	R22	R24	1425	1411	0.98	1421	0.30
3	R21	R22	1197	1205	0.67	1199	0.17
4	R21	R25	1197	1206	0.75	1202	0.42
5	R23	R26	1204	1181	1.91	1197	0.58
6	R22	R23	1336	1339	0.22	1338	0.20

4.3.6. Comparison between Total Station and Photogrammetry

To compare the results of the measurement with the total station, 10 random lines have been chosen across the scene. It should be noted that the measurements of the total station surveying equipment are also subject to error; therefore in this section instead of using the term "error" for comparing results of photogrammetry and surveying, the word "difference" has been used.

Table 4.4 shows the results of measurement with total station and photogrammetry for photo (a) and photo (b). The average of relative differences between total station measurement and photogrammetry measurement for photo (a) and photo (b) are 2.39% and 0.9% respectively. The standard deviation of relative differences for photo (a) and photo (b) is 3.79% and 0.9% accordingly.

As can be seen, due to its shallow angle and mostly because of quantization error (rounding pixel values), photo (a) shows larger differences. The uncertainty increases for the lines that are located further away from the camera. For example, in photo (a) the uncertainty in length for line 124-125 which is close to the camera is about 0.7% of its length while same value for line 104-115 which is further away from the camera is about 11.0% of the line length.

Further analysis indicates that, measured length with photogrammetry follows the three-sigma rule. In other words, if the measured value by total station is assumed as the true value of length, nearly all (99.7%) measurements with photogrammetry lie within $\pm 3\sigma$ of that true value. Though in this case, all measurements are actually within $\pm 2\sigma$ of the assumed true value.

Table 4.4. Measurement comparison between total station and photogrammetry

		Total Station		Photo (a)			Photo (b)		
Point 1	**Point 2**	$l[cm]$	$\Delta Z[cm]$	$l[cm]$	$\sigma[cm]$	$\Delta l[\%]$	$l[cm]$	$\sigma[cm]$	$\Delta l[\%]$
SKID1-104	SKID1-115	117	0.7	133	13	13.68	121	5	1.15
SKID1-104	SKID1-106	406	0.2	401	11.8	1.232	407	5	1.23
SKID2_115	SKID1-106	420	4.1	411	11	2.143	417	4	0
SKID2_110	SKID2_113	684	0.5	679	8	0.731	685	3.5	0.73
SKID3_125	SKID3_124	338	5.2	335	2.5	0.888	341	5	1.78
SKID4_130	SKID3_125	1157	13.3	1144	4	1.124	1160	4	1.13
AOI_117	AOI_118	199	0.3	203	6	2.01	201	2.9	0.47
SKID3_125	SKID2_115	2539	19.3	2523	9.05	0.63	2548	5.6	0.67
R21	SKID4_126	1634	7	1645	5	0.673	1652	4	1.35
SKID4_129	SKID3_113	1311	4.3	1322	7.2	0.839	1323	4	0.72

Another important observation from this comparison shows that small non-planarity on the scene does not affect the error in measurement dramatically. In other words, this error is proportional to the ratio of height difference to the measured line. For example, for line 125-115, which is the longest measured line on the scene, there is a height difference of 19.3cm or 0.76% of its length between its endpoints, while the error for measurement of that line in photo (b) is about 0.67%, which is less than %1 of line's length. Part of this robustness can be attributed to the pose estimation algorithm, which was effective at estimating a translation vector that minimizes this effect. This is in accordance with the mathematical model which was presented earlier in equation (2-82).

4.3.7. Summary

In this chapter, the results of the simulation and experiment have been presented and discussed. The developed simulation tool has been used to investigate the effect of errors in length of reference lines, pixel selection of reference lines, camera height, local direction, focal length and principal points on photogrammetry error. The simulation

results show that error in reference line length has the most important effect on photogrammetry error. It is also found that error in focal length can be compensated during the pose estimation process.

An experiment has been conducted to investigate the accuracy of the developed photogrammetry tool. A simulated traffic accident scene has been created and coordinates of a set of points on the scene have been measured by total station surveying equipment. Several photos have been taken for the photogrammetry analysis. Six lines have been used as references for the pose estimation. The error in estimated length of the reference lines using estimated pose was found to be less than 1%. Comparing the results of measurement with photogrammetry and surveying equipment showed the average difference has been maintained at about ~1% of the line's length. Also, small perturbations in the coplanarity assumption did not have a considerable effect on the photogrammetry measurement. Moreover, it was found that estimated uncertainty concurs with the three-sigma rule.

5. Conclusion and Further works

In this study, the problem of coplanar photogrammetry for traffic accident reconstruction has been investigated. Performing photogrammetry requires two types of information including information on camera characteristics and information on relative camera position and look direction with respect to the scene.

Information on camera characteristics includes elements of focal length and principal point, lens radial distortion coefficients and lens tangential distortion coefficients. Zhang's calibration technique has been presented as the calibration method for the current study and details of that technique including the maximum likelihood analysis for obtaining parameters were discussed.

Information on relative camera position and look direction with respect to the scene can be estimated by a process known as pose estimation. Pose estimation requires *a posteriori* knowledge about the scene. However, most of the pose estimation algorithms in scientific literature require collecting certain information, such as point coordinates or line direction vectors, which are not accessible easily enough for the application of traffic accident reconstruction. Therefore, a new pose estimation algorithm has been developed in the current study. This pose estimation algorithm requires three or more lines with known lengths. A theorem was proved that if an estimated pose preserves the length of three pairwise nonparallel coplanar lines on a scene in back projection, it can preserve the length of any other line on the same plane in back projection. Nonlinear maximum likelihood estimation was introduced to estimate camera pose.

Formulation for back projection or acquiring the world frame coordinates of objects for a planar scene having known pose and camera characteristics have been presented. A maximum likelihood estimation that uses an approximate closed form linear solution as the initial guess was presented for estimating back projection.

Considering the forensic nature of this specific application providing uncertainty for each measurement is necessary. Assuming normal distribution for all errors, formulation for least square minimization to estimate maximum likelihood has been presented. The Levenberg-Marquardt algorithm for obtaining the least square solution was presented and details of acquiring uncertainty in parameters and model prediction were also discussed. Application of this approach to the least square problems of camera calibration, pose estimation and back projection was provided. A comprehensive model to estimate propagation of uncertainties in parameters to length measurement in coplanar photogrammetry was devised.

A photogrammetry computer application for the Windows 7 operating system has been developed for implementing the mathematical model. The developed computer application is capable of handling all required steps for photogrammetry including camera calibration and pose estimation. A simulation tool has also been developed to investigate different hypotheses on error propagation.

The simulation was employed to study the effect of error in parameters such as focal length and principal point on photogrammetry error. The simulation found that error in the reference line length has the most significant effect on photogrammetry error. It was

also discussed that a small error in focal length can be compensated during the pose estimation.

A simulated accident scene was used to examine the proposed photogrammetry model. A set of reference points on the scene was recorded by total station surveying equipment. Aerial photos of the scene were collected using a quadrotor as the camera platform. Two photos with different view angles were employed for the photogrammetry analysis. Using six lines with known lengths on the scene, relative pose of the camera has been estimated for both photos with less than 1% average length error in the back projection process. Ten lines were measured using photogrammetry computer application and collected data by total station surveying equipment. Results comparison showed the average difference of about ~1% of line's length between surveying and photogrammetry measurements. Small perturbations in coplanarity assumption found to have a negligible effect on photogrammetry. Moreover, all photogrammetric measurements found to be in vicinity of 2σ of the total station measurements.

Original contributions include:

- Developing a new pose estimation for planar scenes and proving its validity.
- Developing an uncertain back projection method.
- Applying principles of nonlinear maximum likelihood estimation to acquire a comprehensive mathematical model for uncertain single view coplanar photogrammetry.

A limitation for the current study is that a only single actual flight experiment was done.

This study can be expanded by:

- Incorporating higher order terms of the lens distortion model.
- Collecting a large sample of data to further investigate the three-sigma rule.
- Automated mechanism for introducing reference lines.

References

Björck, A. (1996). *Numerical methods for least squares problems.* Siam.

Bochkanov, S., & Bystritsky, V. (n.d.). *About ALGLIB.* Retrieved 4 3, 2014, from http:// www. alglib .net /

Bochkanov, S., & Bystritsky, V. (n.d.). *Levenberg-Marquardt algorithm for multivariate optimization*. Retrieved 3 4, 2014, from http:// www. alglib. net/ optimization/ levenberg marquardt. php

Bouguet. (2010). *Camera calibration toolbox for MATLAB*. Retrieved 1 11, 2014, from http:// www.vision.caltech.edu/ bouguetj/

Bradski, G., & Kaehler, A. (2008). *Learning OpenCV: Computer vision with the OpenCV library.* O'Reilly Media, Inc.

Chen, H. H. (1990). Pose determination from line-to-plane correspondences: existence condition and closed-form solutions. *Proceedings, Third International Conference on Computer Vision, 1990.* IEEE.

Collier, P. (2002). The impact on topographic mapping of developments in land and air survey: 1900-1939. *Cartography and Geographic Information Science 29(3).*

Cooner, Scott, A., & Balke, K. N. (2000). *Use of photogrammetry for investigation of traffic incident scenes.* Austin: No. TX-99/4907-2.

Corke, P. (2011). *Robotics, Vision and Control: Fudamental Algorithms in MATLAB.* Vol. 73. Springer.

Dhome, M., Lab. d"Electron., U. B.-F., Richetin, M., Lapreste, J.-T., & Rives, G. (1989). Determination of the attitude of 3D objects from a single perspective view. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 11(12)*, 1265-1278.

Du, X., Jin, X., Zhang, X., Shen, J., & Hou, X. (2009). Geometry features measurement of traffic accident for reconstruction based on close-range photogrammetry. *Advances in Engineering Software 40(7)*, 497-505.

Emgu CV, Main Page. (n.d.). Retrieved 3 4, 2014, from http:// www. emgu. com/ wiki/ index. php/ Main_Page

Faugeras, O. (1993). *Three-dimensional computer vision: a geometric viewpoint.* MIT Press.

Fenton, S., & Kerr, R. (1997). Accident Scene Diagramming Using New Photogrammetric Technique. *SAE Technical Paper 970944*.

Fischler, M. A., & Bolles., R. C. (1981). Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM 24(6)*, 381-395.

Fraser, C. (2006). Accident Reconstruction via Digital Close-Range Photogrammetry. *American Society for Photogrammetry and Remote Sensing Annual Conference.* Reno.

Fraser, C. S., Hanley, H. B., & Cronk, S. (2005). Close-range photogrammetry for accident reconstruction. In A. Gruen, & H. E. Kahmen, *Optical 3D Measurements VII* (pp. 115-123).

Guo, T. (2013, 02 25). *OpenCV DevZone*. Retrieved 03 02, 2014, from http://code.opencv.org/issues/2831

Haralick, B. M., Lee, C. N., Ottenberg, K., & & Nölle, M. (1994). Review and analysis of solutions of the three point perspective pose estimation problem. *International Journal of Computer Vision 13(3)*, 331-356.

Haralick, B. M., Lee, C. N., Ottenberg, K., & Nölle, M. (1991). Glossary of computer vision terms. *Pattern recognition, 24(1)*, 69-93.

Heikkila, J., & Silvén., O. (1997). A four-step camera calibration procedure with implicit image correction." Computer Vision and Pattern Recognition. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition.*

Heuel, S., & Förstner., a. W. (2001). Matching, reconstructing and grouping 3d lines from multiple views using uncertain projective geometry. *Computer Vision and Pattern Recognition, 2001. Vol 2.* IEEE.

Hung, Y., Yeh, P.-S., & Harwood, D. (1985). Passive ranging to known planar point sets. *Robotics and Automation. Proceedings. 1985 IEEE International Conference on. Vol. 2.* IEEE.

Josephson, K., & Byrod, M. (2009). Pose estimation with radial distortion and unknown focal length. *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on.* IEEE.

Konecny, G. (1985). "The International Society for Photogrammetry and Remote Sensing-75 Years Old, or 75 Years Young.". *Photogrammetric Engineering and Remote Sensing 51(7).*

Liu, C., Zhu, F., Jin, Z., & Xia, R. (2012). Pose determination from one point and two coplanar line features. *Intelligent Computing Theories and Applications. Springer Berlin Heidelberg*, 577-584.

Luhmann, T., Robson, S., Kyle, S., & Harley, I. (2006). *Close range photogrammetry: Principles, methods and applications.* Whittles.

Merriam-Webster Dictionary. (n.d.). Retrieved 04 03, 2014, from http://www.merriam-webster.com /dictionary/photogrammetry

Mirzaei, F. M., & Roumeliotis, S. I. (2011). Globally optimal pose estimation from line correspondences. *"Robotics and Automation (ICRA).* IEEE.

Neale, W., Hessel, D., & Terpstra, T. (2011). Photogrammetric measurement error associated with lens distortion. Accident Reconstruction.

NEX-5R. (n.d.). Retrieved 3 4, 2014, from http://www.sony.com.au/product/nex-5r

Nistér, D. (2004). An efficient solution to the five-point relative pose problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence 26(6)*, 756-770.

Phong, T. Q., Horaud, R., Yassine, A., & Tao, P. D. (1995). Object pose from 2-D to 3-D point and line correspondences. *International Journal of Computer Vision 15(3),* 225-243.

Qi, W., Li, F., & Zhenzhong, L. (2010). *Review on camera calibration.* Control and Decision Conference (CCDC), 2010 Chinese. IEEE, 2010.

Quan, L., & Zhongdan, L. (1999). Linear n-point camera pose determination. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 21(8),* 774-780.

Randles, B., Jones, B., Welcher, J., Szabo, T., Elliott, D., & MacAdams, C. (2010). The accuracy of photogrammetry vs. hands-on measurement techniques used in accident reconstruction. *SAE 2010 World Congress & Exhibition.* 2010.

Schweighofer, G., & Axel, P. (2006). Robust pose estimation from a planar target. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 28(12),* 2024-2030.

Seber, G. A., & Wild, C. J. (2003). *Nonlinear regression.* A JOHN WILEY & SONS, INC., PUBLICATION.

Shapiro, L., & Stockman, G. C. (2001). *Computer Vision.* ed: Prentice Hall.

SRS Robotic Total Station. (n.d.). Retrieved 4 3, 2014, from http://www.sokkia.com.sg/index7a98.html?top=products &L1=ROBOTIC_TOTAL_STATION&L2=SRX

Wolberg, J. (2006). *Data analysis using the method of least squares.* Berlin Heidelberg: Springer Verlag.

Xu, H. G., & Lu, G. Q. (2002). Application of photogrammetry to measurement of traffic accident scene. *Traffic and Transportation Studies.* ASCE.

Ying, X., & Zha, H. (2005). Camera pose determination from a single view of parallel lines. *Image Processing, 2005. ICIP 2005. IEEE International Conference on. Vol. 3.* . IEEE.

Zhang, Z. (1999). Flexible camera calibration by viewing a plane from unknown orientations. *The Proceedings of the Seventh IEEE International Conference on. Vol. 1*. IEEE.

Zhang, Z. (2004). *"Camera Calibration", Chapter 2, in G. Medioni and S.B. Kang, eds., Emerging Topics in Computer Vision.* Prentice Hall Professional Technical Reference.