Comparison of Camera Calibration Methods

Using Virtual Images and Calibration

Environment

by

Meng-Lin Lee

A thesis submitted to

The Faculty of Graduate Studies and Research

in partial fulfillment for the degree of

Master of Applied Science in Electrical Engineering

Department of Systems and Computer Engineering

Carleton University

Ottawa, Ontario

January, 2004

Copyright 2004,

Meng-Lin Lee

National Library
of Canada

Bibliothèque nationale
du Canada

Acquisitions and
Bibliographic Services

Acquisitions et
services bibliographiques

395 Wellington Street
Ottawa ON K1A 0N4
Canada

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

ISBN: 0-612-89857-1

Our file Notre référence

ISBN: 0-612-89857-1

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this dissertation.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de ce manuscrit.

While these forms may be included in the document page count, their removal does not represent any loss of content from the dissertation.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

Canada

Abstract

For the past few decades, researchers have performed many comparison studies for different camera calibration methods. However, previous comparison studies mainly focus on back projection accuracy of a camera model and very few on accuracy of parameter estimation for the camera models. In this thesis, the comparison study is performed to observe the interaction between camera parameters and the calibration environment. The investigation focuses on Direct Linear Transformation, a popular linear method, and Tsai, a well known nonlinear camera calibration method. The effect of digitization error, resolution noise, radial lens distortion, and scene point coordinates measurement errors on Direct Linear Transformation (DLT) and Tsai methods were analyzed and compared. The calibration environment, true translation, and rotation (extrinsic) parameters, and true camera (intrinsic) parameters were completely controlled using computer simulation of image formation. Performance of the two methods is evaluated in terms of accuracy of parameter estimates for the respective camera models under various noise, distortion, and error conditions. The results of the analysis show that some camera parameters are more sensitive to noise than others. The comparison study also shows that in some cases Tsai's method is more sensitive to noise in the image plane than the DLT method. The experimental results also suggest that Tsai's method can estimate radial lens distortion reasonably well. The analysis indicates that both the DLT and Tsai's methods can still be improved. Suggestions for improvement are provided in the conclusions.

Acknowledgements

I would like to thank my supervisor, Professor V. C. Aitken, for his support and suggestions on my research and thesis. His financial assistance on the implementation of this thesis is greatly appreciated.

Table of Contents

Chapter 1 Introduction	1
1.1 Overview	1
1.2 Thesis Statement	6
1.3 Thesis Overview	8
Chapter 2 Camera Calibration	9
2.1 Image Formation	9
2.2 Types of image distortions.....	11
2.3 Types of Calibration Targets	14
2.4 Camera Parameters	15
2.5 Factors that Affect Camera Calibration Accuracy.....	18
2.6 Review of Accuracy Evaluation and Parameter Studies.....	21
2.7 The Proposed Camera Parameter Comparison Method.....	25
2.8 Camera Models	26
2.8.1 DLT and Modified DLT Camera Model	28
2.8.2 Tsai's Camera Model.....	34
2.9 Camera Calibration Methods	37
2.9.1 Linear DLT Camera Calibration Algorithm	38
2.9.2 Linear Modified DLT Camera Calibration Algorithm	40
2.9.3 Tsai's Non-coplanar Camera Calibration Algorithm.....	49
2.10 Comparison of Linear DLT, Linear Modified DLT and Tsai Camera calibration Methods.....	63
Chapter 3 Experiments and Discussion	66

3.1 Gaussian Noise Generation.....	67
3.2 Monte Carlo Simulation.....	68
3.3 Virtual Camera and Calibration Jig	70
3.4 DLT and Tsai Camera Calibration in a Perfect Calibration Environment.....	72
3.5 Effect of 3D Scene Point Measurement Errors.....	74
3.5.1 Human Measurement Error.....	75
3.5.2 Effect of Calibration Jig Errors.....	80
3.6 Effect of Radial Distortion on Camera Parameters.....	82
3.7 Effect of Resolution Noise on Camera Parameters.....	85
3.8 Effect of Digitization Error on Camera Parameters.....	87
3.9 Effect of 3D Scene Point Coordinate Measurement Errors, Radial Lens Distortion, Resolution Noise, and Digitization Error on Camera Parameters	89
3.10 Effect of Number of Calibration Points on Calibration Results	93
3.11 Discussion of Results.....	95
Chapter 4 Conclusions and Future Work.....	100
References.....	103

List of Figures

Figure 1.1: An illustration of geometric camera calibration	2
Figure 1.2: TART Camera Calibration	7
Figure 2.1: Image formation process	9
Figure 2.2: The images of the radial distortion. The left image is the object without lens distortion. The middle image illustrates the barrel distortion. The right image illustrates the pincushion distortion.	12
Figure 2.3: Types of image distortions [2].....	13
Figure 2.4: Types of calibration targets	15
Figure 2.5: An image of a circular control point	20
Figure 2.6: DLT method of mapping of 3D and 2D image points	28
Figure 2.7: The mapping of 3D scene points and 2D image points for Tsai's method	34
Figure 3.1: Zero-mean Gaussian distribution marked with 68%, 95%, and 99.7% Empirical Rule [30].....	67
Figure 3.2: Overview of one test case that applied Monte Carlo Simulation	69
Figure 3.3: Computer generated camera and calibration jig in world reference frame	70
Figure 3.4: Low cost wood calibration jig	72
Figure 3.5: Simulation overview for human measurement errors test cases	75
Figure 3.6: Experiment overview for jig errors	80
Figure 3.7: The back projected 3D scene points for LMDLT and Tsai's method.....	98
Figure 3.8: The average backprojection errors for each row in the X_w direction of the world frame	99

List of Tables

Table 2.1 A summary of intrinsic and extrinsic parameters used by DLT and Tsai	17
Table 2.2 Comparisons of DLT, LMDLT, and Tsai Camera Calibration Methods	65
Table 3.1 Calibration results for LMDLT and Tsai in Case #0	74
Table 3.2 Summary of experiment setup for test Cases #1 and #2	78
Table 3.3 Simulation results for Case #1	79
Table 3.4 Simulation results for Case #2	79
Table 3.5 Simulation results for LMDLT and Tsai methods when jig errors at 2mm noise level	81
Table 3.6 Simulation results for LMDLT and Tsai methods when jig errors at 5mm noise level	82
Table 3.7 Experiment setup for effect of radial distortion test cases: Cases #5 and Case #6	83
Table 3.8 LMDLT and Tsai methods test results comparison for Case #5	84
Table 3.9 LMDLT and Tsai methods test results comparison for Case #6	84
Table 3.10 Experiment setup for Case #7	86
Table 3.11 Experiment results for Case #7	87
Table 3.12 Experiment results for Case #8	88
Table 3.13 Summary of Experimental setup for test cases #9 and #10	91
Table 3.14 Experiment results for Case #9 Effect of digitization Error, resolution noises, radial distortion and human measurement error on camera parameters	92
Table 3.15 Experiment results for Case #10 Effect of digitization Error, resolution noises, radial distortion and human measurement error on camera parameters	92

Table 3.16 Experiment results for Tsai method: Cases #11, #12, and #13.....	94
Table 3.17 Experiment results for LMDLT method: Cases #14, #15, and #16.....	95
Table 3.18 Summary of parameter comparison between LMDLT and Tsai's method	96

Chapter 1 Introduction

1.1 Overview

Camera systems have been used by a variety of applications such as dense reconstruction, visual inspection, object localization, camera localization, autonomous robotics navigation systems, medical imaging, and industrial parts inspection, etc.[1].

Before a camera system can be used for these applications, the relationship between camera-captured images and real world objects has to be found. Camera calibration is one of the methods to find this relationship between a 3D scene and a 2D image. A general camera calibration consists of radiometric and geometrical stages [2].

Radiometric camera calibration is used to build an accurate CCD camera sensor model for a camera. In [3], a radiometric camera model was built to allow the removal of sensor noise by a radiometric CCD camera calibration. In the field of robotics navigation, we are more interested in the location of objects in the field of view of the camera and standard camera calibration methods usually concentrate on geometrical camera calibration. The discussions in this thesis focus on the Direct Linear Transformation method (DLT) and Tsai's Method [4][5].

The purpose of a geometric camera calibration is to estimate the camera position and orientation in the environment and parameters of the optical camera model, especially for image distortion. A typical geometric camera calibration setup is illustrated in Figure 1.1. To perform a geometric camera calibration, one needs a calibration target, a frame grabber, a computer running calibration software, and the camera to be calibrated. The calibration target needs to have easily detectable features

such as points placed at known positions. As shown in Figure 1.1, the camera is directed towards the calibration target to allow images to be captured. During a calibration process, the camera is fixed and three images are captured from the target board. The target board is placed at three different heights to give the non-coplanar set of 3D scene points needed by the calibration algorithms, and one image is captured at each height.

Figure 1.1: An illustration of geometric camera calibration

Camera calibration was initially used in photogrammetric aerial surveys. For these applications high-resolution photographic cameras were used. Because these high quality photographic cameras have little lens distortions, classical linear camera calibration methods such as DLT, or Faugeras [4][6] produce acceptable results. High quality photographic cameras are very expensive and may not allow real time processing.

For robotic imaging systems, real time images are often required by the robotic applications. Camera calibration processes used by robotic applications usually need to be reasonably fast and automatic. These requirements rule out photogrammetric cameras and make cost effective off-the-shelf CCD cameras a better choice for machine vision applications [2].

Although off-the-shelf CCD cameras have good features such as providing real time images and low cost, they have some drawbacks as well. As indicated in [8], image distortion is one of the major problems resulting from the low quality lenses used in these cameras. To obtain excellent camera calibration results one requires the knowledge of accurate 3D image point coordinates and their associated 2D image point coordinates. Lens distortion makes camera-captured image points deviate from their ideal positions on the image plane and make camera calibration results less reliable. In order to correct image distortion and improve the reliability of calibration results, lens distortions need to be modeled and this creates non-linear camera models. Solving these non-linear systems is not an easy task. It requires computation-intensive and time-consuming non-linear parameter optimization processes and usually results in computationally complex calibration methods. Other than the performance problem, these non-linear methods sometimes are also numerically unstable. This is because their search for a global optimal solution may fail and accidentally find local extremum. To overcome these problems, many proposed camera calibration techniques [5][8] focus on simplifying and improving the optimization process in order to increase calibration performance and reliability. There are many examples of non-linear camera calibration methods such as Faig, Karara, and Brown as discussed in Tsai's paper [5]. One of the most well-known non-linear

calibration techniques is Tsai's two-step method [5] because it produces good calibration results for off-the-shelf CCD cameras.

Off-the-shelf CCD cameras are widely used in robotics related applications and there are many linear or non-linear camera calibration methods available to be used to calibrate these cameras. How does a calibration operator choose the most appropriate camera calibration method among them? What are the differences between these proposed calibration methods? To answer these questions, a thorough performance comparison study for camera calibration methods such as DLT, modified DLT, and Tsai's method is required. Results of such a comparison will not only enable a camera calibration operator to choose an appropriate method for specific applications, but will also allow us to see which method is more sensitive to certain calibration environment variables to help researchers understand camera models better and make improvements. What evaluation schemes can be used for a comparison study? Many proposed accuracy evaluation schemes [1][2][5][7][8][9][10][11][12] are used by researchers and can be divided into two main categories:

1. Methods that compare the output accuracy of an estimated camera model such as the predicted back projection accuracy;

2. Methods that compare parameters of estimated camera models.

For example within category 1, Tsai [5] suggested three evaluation techniques:

1. Using 3D coordinates obtained from triangulation of two calibrated cameras;
2. Using radius of ambiguity; and
3. Using 3D measurements.

In [8], two approaches were mentioned: 1. physically measuring the position and orientation of the camera, 2. comparing the back projected 3D points with the corresponding true scene points. Of course, the second approach is more practical than the first because physically measuring the camera position and orientation is difficult or impossible to achieve in most cases.

While some researchers focus on evaluating the accuracy of camera calibration methods, others focus on studying the parameters of camera models. Both studies provide information regarding the accuracy of a camera calibration. Tsai [5] pointed out that “true parameters of a camera model” are difficult to obtain; hence, it is difficult to provide a reference for comparison. In spite of this difficulty, the study of camera parameters has still been carried out by some researchers [9][10][11] based on the following assumption: if we can precisely make a calibration target and detect the image points accurately, we can use this set of 3D scene points and 2D image points to obtain a set of calibration parameters that can serve as a reference.

Another method to study parameters is to use a software camera simulator to generate virtual calibration targets and images. This approach can resolve the difficulty of finding the true parameters. The most recent work of this kind was done by A. Wykurz [7]. In A. Wykurz’s thesis, a camera simulator called MatCam was introduced. Using MatCam a virtual camera and calibration target can be generated according to desired specifications. This overcomes the difficulty of finding “true scene points coordinates” and “true camera orientation and translation matrices” and allows a calibration comparison study to be carried out under a computer generated and controlled environment. Therefore, this virtual camera calibration environment was employed in

this thesis to investigate and compare DLT and Tsai camera calibration methods. Within MatCam, the user has complete control over placement of the 3D scene points, the position and orientation (extrinsic parameters) of the camera within the environment, and also the internal (intrinsic) camera parameters such as the focal length, scale factor, and distortion parameters.

1.2 Thesis Statement

Previous accuracy evaluation studies do not generally have access to true camera parameters. Therefore, one can only conduct an actual calibration and assume that parameters obtained are accurate enough to represent true parameters. Studies of this kind can be easily compromised by noise and errors such as 2D image resolution noise or 3D scene point coordinates measurement errors. Using a computer generated camera and calibration jig, one can control the camera calibration environment and can have true parameters to serve as a baseline for parameter estimation studies. Furthermore, the distribution of random errors can be controlled to be present at the 2D image plane only, at the 3D scene only or at both places. This enables us to observe the impact that each individual calibration environment variable, or the impact that a group of variables can have on camera parameter estimation. The study can be used to improve camera calibration results for autonomous robotics such as the one illustrated in Figure 1.2.

Figure 1.2: TART Camera Calibration

In this thesis MatCam is used to generate the calibration target and virtual images. DLT, modified DLT and Tsai camera calibration methods were implemented within this computer controlled environment. Several experiments were conducted to study and compare how DLT and Tsai camera parameters interact with the following factors:

1. Effect of Gaussian noise in 2D image points;
2. Effect of 3D scene points coordinates measurement error;
3. Effect of radial lens distortion; and
4. How the number of calibration points affects camera calibration results.

The calibration results were analyzed and compared based on camera model's intrinsic and extrinsic parameters. More details regarding theory, implementation, and experiments are discussed in the following chapters.

1.3 Thesis Overview

The rest of this thesis is organized as follows: Chapter 2 provides a review of camera models and calibration algorithms. Camera calibration implementation and assessment are discussed in the context of related literature. Chapter 3 explains the experiments performed and analyzes simulation results. Finally, the conclusions and future work are discussed in Chapter 4.

Chapter 2 Camera Calibration

2.1 Image Formation

The process of capturing an image using a CCD camera is described by the following steps as illustrated in Figure 2.1:

1. Project a real world object onto the CCD array of the camera;
2. Convert captured CCD images into an analog signal and output to a frame grabber;
and
3. Resample the analog video signal into a discrete signal, in units of pixels and store these inside computer memory.

Figure 2.1: Image formation process

In step1, image distortions occur when mapping a 3D scene object into the camera captured 2D image. There are many kinds of image distortions and more details about them are provided in Section 2.2. If an analog camera is used, there is a “line jitter” effect in step 3 when signal re-sampling occurs. This jitter is caused by conversion from an analog to a discrete signal [2]. A common CCD off-the-shelf camera has to convert its discrete sensor signal into a smooth analog video signal for output. When this output signal reaches a computer frame grabber, it is re-sampled into a discrete signal in the frame grabber. Usually, the clocks of the CCD camera and the frame grabber are running at different speeds and this clock-timing mismatch causes different signal sampling rates in the frame grabber and the CCD camera. Therefore, the captured pixel distance in the frame grabber memory may not be the same as the distance between CCD camera sensor cells. So a horizontal scale factor is needed in the camera model to fix this problem. In the vertical direction the sampling is controlled by the distance between adjacent CCD sensor rows; hence, in the vertical direction there is no line-jitter effect as in the horizontal. Tsai [5] models this line jitter effect using an “uncertainty image scale factor s_x ”. The DLT method [4] does not consider the line jitter effect.

During the image formation processes, there are also other factors such as distortions, noises, and errors contaminating the captured images stored in computer memory. More details of these sources of errors are introduced in Sections 2.2 and 2.5.

2.2 Types of image distortions

A camera system can have many kinds of image imperfections. Image imperfections are called aberrations [13]. To be more specific, an aberration means that the light rays from a single real world point do not intercept at a point on the camera's CCD image plane. According to [13], image aberrations in monochromatic light are classified into spherical aberration, coma, astigmatism, curvature of field, and distortion. The first four aberrations: spherical aberration, coma, astigmatism, and curvature of field, affect the quality of an image but not its geometry. The last aberration, distortion, affects the geometry of an image; this distortion is defined to be the displacement of an image away from its "theoretical" position determined by pure perspective projection.

As mentioned previously, this thesis focuses on geometrical camera calibration in the machine vision field. Therefore, of the five image aberrations only "distortion" is important and the other four are less significant for camera calibration because they have only minor effects on calibration results. One of the key things for geometrical camera calibration is to correctly determine lens distortions. Other than lens distortions, there is another kind of distortion that results from perspective projection. Distortion caused by perspective projection can distort a square shaped object into a rectangle. The most important image aberrations related to geometrical camera calibrations are summarized below.

1. Perspective distortion: Mapping a 3D scene object into a 2D image is called perspective projection. During this projection process the shape of the object in real world is distorted. As mentioned above, a square shape may be distorted into a rectangular shape in the image.

2. Lens distortions:

- a. Radial distortion: Radial distortion is a function of the radius of the lens and can be categorized into two types: barrel and pincushion distortions. The names are chosen according to the way object shapes are distorted in the image. Barrel distortion is very common in wide-angle lenses. As shown in Figure 2.2, barrel distortion causes straight lines to bend outward from the center of the image. Pincushion distortion makes straight lines bend inward towards the image centre.

Figure 2.2: The images of the radial distortion.¹ The left image is the object without lens distortion. The middle image illustrates the barrel distortion. The right image illustrates the pincushion distortion.

- b. De-centering distortion: De-centering distortion for a compound lens camera is caused by the non-collinearity of lens centers and other manufacturing defects [14]. According to Brown [16], a minor de-centering distortion can be approximated by adding a thin prism to the lens system. The de-centering distortion described by the thin prism model is

¹ These figures are obtained from the following web links:
<http://www.mellesgriot.com/glossary/wordlist/glossarydetails.asp?wID=102>, and
<http://www.mellesgriot.com/glossary/wordlist/glossarydetails.asp?wID=188>.

asymmetric to the principal point of auto-collimation² and is also called tangential distortion.

However, Brown’s research showed that when de-centering distortion is significant, the thin prism model is not sufficient. This is because when the degree of de-centering distortion is significant, both tangential and asymmetric radial distortions are present; hence, this makes the thin prism model insufficient. To fully describe de-centering distortion, the thin prism model must be augmented by additional terms [16][17].

Figure 2.3: Types of image distortions [2]

² According to page 12 in [14], definition for *principal point of auto-collimation* is: The point where a ray in the object space is perpendicular to the plane of the photograph intersects that plane. For this thesis, “principal point of auto-collimation” is the “principal point of a CCD camera”.

Figure 2.3 is a tree diagram of different types of image distortions. Of all the types of nonlinear distortions mentioned above, radial distortion causes image points to shift away from their theoretical position and has the most noticeable effect on geometrical camera calibration. In order to obtain a better calibration result, the camera model should include radial distortion. Low quality camera lenses shrink images differently at the edges than at the center of the field of view [15]. Radial distortion usually happens in wide-angle lenses and is less common in tele-angle lens. The virtual camera studied in this thesis is targeted for autonomous robotics vehicles and its main purpose is to provide image information for navigation purposes. In this case, an off-the-shelf wide-angle CCD camera is more suitable than a tele-angle lens camera. A wide-angle CCD camera provides a wider field-of-view and is more suitable to provide information for vehicle route planning or collision avoidance. Tsai proposed his camera calibration method [5] to deal with off-the shelf cameras that suffer from radial distortion. According to Tsai's paper [5], it is sufficient to consider only two terms of radial distortion in machine vision related applications. Therefore, the computer-generated camera in this thesis approximated a real physical camera up to second order distortion.

2.3 Types of Calibration Targets

In order to perform a camera calibration, an associated set of known 3D scene and 2D image coordinates is required. The coordinates of these 3D scene points are often measured directly using coordinate measuring devices but coordinates of their images have to be computed from camera-captured 2D images. To easily find coordinates in the image, scene targets need to have detectable features. With these detectable features in the image, coordinate calculation algorithms can easily extract essential information from

these camera-captured images to locate feature points and calculate their 2D image coordinates.

In practice, the most common calibration targets are planar surfaces with lines, circles, or squares mounted as shown in Figure 2.4 [2][7][18]. From these calibration targets, feature extraction techniques can be used to estimate the position of some image points. For example, for square calibration targets, a feature extraction technique can be used to locate image coordinates of the corner points; for circular targets, the centre of gravity of each circle is used as feature points for locating image coordinates. These feature points in the images can be detected either manually or automatically.

Figure 2.4: Types of calibration targets

2.4 Camera Parameters

In order to relate 2D information in an image to the 3D scene, a mathematical model of the camera and image formation process is used. The camera parameters are part of this mathematical model and the purpose of camera calibration is to estimate these camera parameters. Camera parameters are classified into two groups:

1. Intrinsic camera parameters
2. Extrinsic camera parameters.

The following definition for intrinsic and extrinsic camera parameters can be found from [5][18].

Intrinsic camera parameters describe optical, digital properties, and internal geometrical characteristics of a camera. Intrinsic parameters also define the relationship between an image point coordinate described in a camera reference frame and the frame buffer reference frame. Depending on different camera models, the intrinsic parameters used may vary slightly. For example, the DLT camera model does not include an intrinsic parameter to deal with the line-jitter effect but Tsai does. The following list of intrinsic parameters was proposed by Tsai's calibration method:

1. The principal point;
2. Effective focal length;
3. Number of elements in the horizontal scan line of the CCD array;
4. Number of pixels in computer frame memory;
5. Distance between adjacent pixels in CCD array;
6. Horizontal scale factor; and
7. Lens distortion coefficients.

The effective focal length, horizontal scale factor, and lens distortion coefficients are estimated during a camera calibration process. The other four intrinsic parameters are usually provided by the camera manufacturer. The image centre, also called the principal point, is usually chosen to be at the centre of the CCD image array.

Extrinsic camera parameters define the orientation and position of the camera with respect to a real world reference frame. The position of a point in the world frame, $[P]_{world}$, is related to its position in the camera frame, $[P]_{camera}$, by

$$[P]_{camera} = Q[P]_{world}, \quad (2.1)$$

$$\text{where } Q = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & T_x \\ r_{21} & r_{22} & r_{23} & T_y \\ r_{31} & r_{32} & r_{33} & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.2)$$

is the homogeneous transformation matrix ,

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \quad (2.3)$$

is the rotation matrix, and $T = [T_x \ T_y \ T_z]^T$ is the translation vector.

Although the rotation matrix I has 9 elements they actually only depend on three rotation angles around the X, Y, and Z axes. There are several different definitions for these three angles. This thesis employs the definition used by V. C. Aitken [19][27] and calls these three angles roll, pitch, and yaw angles respectively. Roll-pitch-yaw angles together with the translation vector T describe the camera's position and orientation in 3D real world space. Table 2.1 summarizes the intrinsic and extrinsic parameters used by the Tsai and DLT methods.

Table 2.1 A summary of intrinsic and extrinsic parameters used by DLT and Tsai

Parameter types	DLT	Tsai
Intrinsic Parameters	Principal point, effective focal length, Distance between adjacent pixels in CCD array	Principal point, Effective focal length, Number of elements in the in the horizontal scan line of the CCD array, Number of pixels in computer frame memory, Distance between adjacent pixels in CCD array, Horizontal scale factor, and Lens distortion coefficients.
Extrinsic Parameters	Roll, pitch, yaw, T_x, T_y, T_z	Roll, pitch, yaw, T_x, T_y, T_z

2.5 Factors that Affect Camera Calibration Accuracy

There are many factors that may change the accuracy of a camera calibration. It is important to discuss these factors to help understand possible sources of errors. With this knowledge in mind, a calibration operator can reduce unnecessary errors during the calibration process and obtain a better calibration result. These factors are summarized as follows:

1. Number of calibration points used;
2. Lens distortions;
3. Outliers;
4. Scene point coordinates measurement error;
5. Digitization error; and
6. Image Resolution noise.

Number of calibration points used: Tsai has clearly explained in his paper [5] that the calibration accuracy of his camera model may be affected by the number of scene points used. He also mathematically analyzed how the number of target points changes the calibration accuracy and he provided an equation for a calibration operator to determine whether the number of points used is sufficient for a good result. J. Kofman et al. [12] investigated Direct Linear Transformation stereo camera calibration and reported that accurate scene point measurements seem to result in better 3D reconstruction accuracy and can balance the non-uniform distribution of scene points.

Lens distortions: As mentioned previously, lens distortion results in image displacement. One of the main tasks of calibrating a camera is to find distortion parameters so that the

distorted image points can be corrected to their undistorted positions. However, lens distortion is nonlinear. This makes finding accurate distortion parameters very difficult.

Outliers: Outliers can also change the outcome of a camera calibration. The significance of outliers depends on the calibration method used to minimize the error of estimation; some methods are more sensitive to outliers than others. For example, when calibration methods only minimize the sum of squares of estimation errors, these techniques are more sensitive to gross errors or outliers in the data [10].

Scene point coordinates measurement error: Accurately known scene point coordinates are one of the very important factors for a successful camera calibration. These scene point coordinates can be measured accurately using coordinate measuring devices. However, coordinate measuring devices are very expensive, have limitations and cannot be applied for all calibration environments. Manufacturing an accurate calibration board is also very expensive and challenging. In most of the calibration environments, scene point coordinates measuring error is unavoidable. Because the calibration algorithm is based on the knowledge of both a known scene and known image point positions, scene points coordinates errors affect the calibration result.

Digitization errors: Digitization errors are the difference between the actual centre of an image point and the digitized centre [7]. Image digitization errors occur when a true image is digitized and divided into pixels. Digitization error causes the true position of an image point to be dislocated from its ideal position. The centre of the image may also shift because of the image digitization process. After digitization, the image is converted to discrete pixels. The centre of the image before digitization may not be located exactly at a discrete pixel location; hence, digitization error is introduced. Generally digitization

error is related to the resolution of the image. When the camera CCD array has a higher resolution, an image of the same size is represented by more pixels and the pixel-to-pixel distance becomes smaller. Therefore, the digitization error of the image center may be less.

Resolution noise: According to [13], image aberrations: spherical aberration, coma, astigmatism, and curvature of field, affect the quality or sharpness of an image and make selecting an image point more difficult. When the calibration target points are a group of circular points, to manually select the centre of a circle is difficult due to these four image aberrations and the resolution limit of off-the-shelf CCD cameras [7][9]. From Figure 2.5, we can clearly see the problem to determine the exact centre of a circular control point.

Figure 2.5: An image of a circular control point

Because the image is formed by a group of discrete pixels, when selecting the centre manually, an error is introduced. The study done by [9] showed that the error statistics

for manual selection and those from the method of moments are approximately Gaussian distributed.

2.6 Review of Accuracy Evaluation and Parameter Studies

A properly calibrated camera system can be very useful. It can be used to locate a scene point, to calculate the orientation and position of a robot if employed by the robot's navigation system, or to do measurement inspection for industrial products. Because a well-calibrated camera system is so powerful, the importance of an accurate camera calibration cannot be overemphasized. Many studies have been conducted either to evaluate the accuracy of a camera calibration [1][7][8] or to study the influence of noises on a camera calibration [7][9][10][11][21]. The goal of these studies is to gain a better understanding of a camera model and to find methods to improve the performance and accuracy of a camera calibration. Six accuracy assessment methods are outlined in [1]. They are divided into two categories:

1. Methods that use 3D coordinates. This category can be further divided into 4 subcategories.
 - 1.1 checking the difference between the real scene point and the stereo triangulation of back projected optical lines;
 - 1.2 checking the radius of ambiguity;
 - 1.3 checking the distance between the 3D scene point and its associated back projected optical ray (assuming the depth information is known for a single camera); and
 - 1.4 checking the normalized stereo calibration error [8].
2. Methods that use 2D image coordinates:

2.1 checking the distorted 2D image coordinates;

2.2 checking the non-distorted 2D image coordinates.

Common camera calibration methods require accurate 3D coordinate measurements of a calibration pattern. But the construction of a very accurate calibration pattern is difficult and expensive. Therefore, there are usually errors in the measured 3D scene point coordinates. How will the camera model behave if an imperfect calibration target is used? What is the sensitivity of the parameters to these coordinate measurement errors of 3D calibration scene points? These are interesting topics that researchers in the field would like to find out.

Many studies have been done by introducing noises during the calibration process to test the robustness of a calibration method. A few examples can be found in Lavest et al. [21], Kopparapu and Corke [9], Kumar and Hanson [10], and Kumar [11]. The surveys done by Kumar [11], and Kumar and Hanson [10] looked into the effect of outliers on the estimation of the camera's 3D location and orientation (also called exterior orientation or pose determination). In [11] a sensitivity analysis of pose determination to incorrect estimates of the image centre and focal length was performed. This study showed that an incorrect estimate of the focal length only notably affects the translational parameter that is in the same direction as the optical axis. The image centre estimation errors were found to have no great effect on the translational parameters. Lavest et al. [21] have analyzed the sensitivity of intrinsic parameters to the 3D scene coordinate measurement errors and also to the 2D image coordinate measurement errors. They use a new camera calibration approach that allows them to estimate both the camera calibration parameters and the 3D scene point coordinate at the same time. Their experimental

results, obtained from their new camera calibration technique, showed that when n points can be observed accurately out of a set of m different points of view, the accurate estimation of camera intrinsic parameters depends on the accurate measurement of the image point coordinate and does not depend on the accurate 3D scene coordinate measurements [21].

Of all the literature reviewed, the closest work regarding parameter studies was done by S. Kopparapu and Corke. Kopparapu and Corke [9] focus on discussing the effect of image coordinate noise on both intrinsic and extrinsic camera parameters. They have made a precise calibration target and then detect the image points to sub-pixel accuracy by employing the method of moments. Then this set of 3D scene points and 2D image points were used to obtain a set of calibration parameters that serve as a reference in their analysis. Gaussian random artificial noises were generated and then added to the 2D image points. These noisy 2D points with their corresponding accurate 3D world scene points were used to calculate the “noisy camera parameters”. A Monte Carlo simulation method was employed to investigate the effect of noise. Their mathematical analysis and experimental results indicated that the noise is proportional to the errors in image centre, horizontal and vertical scaling factors, and translation parameters. They also found that if the noises are Gaussian distributed, so are the errors in image centre, horizontal and vertical scaling factors, and translation parameters. The orientation parameters were found to be more complicated. They reported that the orientation parameters are more sensitive to noise in 2D image coordinate measurements. In addition, their simulation results revealed that pitch and roll parameters are less sensitive to noise than yaw [9].

The study conducted by S. Kopparapu and Corke is based on the assumption that 2D image coordinates and associated 3D scene point coordinates are both measured correctly and the least square solutions obtained from the set of corresponding 2D and 3D coordinates are correct as well and can represent the true parameters of their camera model. These “true parameters” are estimated using a least-squares method. Are there any errors in the true parameters? To verify this is very difficult because it can only be done by mechanically measuring the orientation of the camera used. One may argue that if a camera model can correctly back project 2D image points to their associated 3D scene positions then the camera parameters that describe this camera model are true. This argument may sound reasonable but according to K. Astrom [22], an estimated model that produces perfect outputs does not guarantee that it represents the true model. Tsai has also indicated that “true parameters of a camera model” are difficult to obtain; hence, it is difficult to provide a reference for comparison. Therefore, a parameter study based on a real camera calibration may be biased because the “true parameters” used for comparison are also estimated and may not be suitable to serve as a reference for the study.

However, the difficulty of knowing true camera parameters can be overcome by using a virtual computer generation environment. In A. Wykurz’s thesis [7], a software camera calibration environment was created. His work allows a calibration operator to specify the true intrinsic and extrinsic parameters of a camera. The software also captures image points from the computer generated 3D calibration target and interfaces with a camera calibration method to estimate intrinsic and extrinsic parameters. Therefore, a better approach for studying and comparing camera parameters for different calibration

algorithms such as DLT and Tsai is to employ Wykurz's MatCam. More details about this proposed approach are explained in the next section.

2.7 The Proposed Camera Parameter Comparison Method

As discussed in Section 2.5, there are six factors that contribute to camera calibration error. How will the camera model behave in a less perfect calibration environment with these six changing factors? What is the sensitivity of the parameters of the camera models to these different sources of error? Which camera calibration algorithm performs better and is less sensitive to errors? These are interesting topics to explore. The purpose of this thesis is to experiment and compare the effects of these error factors on the popular DLT and Tsai camera calibration techniques.

From the literature review, it has been identified that there are problems with the existing parameter study techniques and improvements are required. The studies conducted in [10][11] are mainly on parameter sensitivity analysis. The study by S. Kopparapu and Corke [9] used the estimated parameters from a preliminary camera calibration to serve as "true parameters" for comparison. Their study may be biased because it relies on the accuracy of their calibration equipment and their least-squares parameter optimization process. An accurate calibration target is known to be difficult to build and the least-squares method is sensitive to gross errors and outliers in the data [10]. Therefore, their "true parameters" obtained from an actual camera calibration may not be suitable to serve as a reference for a parameter study. Furthermore, the study performed by S. Kopparapu and Corke [9] concentrates only on the effect of 2D image coordinate errors.

The virtual calibration environment built by Wykurz [7] provides an excellent tool for a parameter comparison study. Wykurz has studied some of the effects of error sources on Tsai's camera parameters. However, his study does not employ Monte Carlo simulations and cannot represent the error statistics in some cases (for example, the effect of resolution noise or the effect of 3D scene coordinates errors.). Furthermore, his study is on Tsai's algorithm only and does not compare this with any other camera calibration algorithm.

The approach used in this thesis employs Wykurz's MatCam virtual camera calibration environment to ensure that true parameters are available. Monte Carlo simulation is conducted when required to estimate error statistics. The code, including DLT and Linear Modified Direct Linear Transformation (LMDLT), was implemented and interfaced with the MatCam software. Modifications were also made to MatCam and Tsai's non-coplanar camera calibration software to make them suitable for Monte Carlo simulation or for each different simulation case. As explained in a later section, the DLT method suffers from a "non-orthogonal problem" and cannot produce satisfactory calibration results alone. Therefore, the proposed parameter comparison study focuses on LMDLT and Tsai methods. Six main experiments are used to explore and compare the interaction of LMDLT and Tsai's parameters with different sources of errors.

2.8 Camera Models

A camera model is a mathematical formulation of a camera and image formation process. A camera model provides information of internal geometry, position, and orientation of a real camera [1]. As shown in Figure 2.1, an image formation process is modeled by this mathematical camera model. The goal of a camera calibration is to find

a camera model that most closely represents the real camera device so that scene points can be accurately mapped to camera-captured image points. Camera models can be categorized into explicit and implicit camera models. Implicit camera models do not require physical camera parameters, for example, the two-plane method proposed by G. Wei and S.D. Ma [20]. The so-called explicit camera models are the ones using physical camera parameters to provide the internal geometry, position, and orientation of the camera. This thesis discusses explicit camera models. The well-known examples of this kind are DLT and Tsai camera models.

The mapping from 3D scene points (in real world coordinates in meters) to camera-captured image points (in pixels) can be described in the following four steps [1][5]:

1. Transform the scene points from the world reference frame to the camera reference frame;
2. Use perspective projection to map 3D scene points into the image plane and find undistorted image points;
3. Model lens distortion and find distorted image points; and
4. Transform distorted image points from metric coordinates to pixel coordinates.

Details of these four steps are explained in the following sections based on the DLT and Tsai camera models respectively. A comparison of DLT and Tsai camera models is also provided.

2.8.1 DLT and Modified DLT Camera Model

F = focal length
 O_w = origin of world reference frame
 $O_c(X_o, Y_o, Z_o)$ = origin of camera reference frame, centre of projection
 O_i = image frame origin
 O_f = frame grabber frame origin
 $P_w(X_w, Y_w, Z_w)$ = scene points coordinates in world reference frame
 $P_c(X_c, Y_c, Z_c)$ = scene points coordinates in camera reference frame
 P_u = undistorted image coordinates
 P_d = distorted image coordinates

Figure 2.6: DLT method of mapping of 3D and 2D image points

Step1 Transfer Scene Points Reference from World frame to Camera Frame

The DLT camera model transfers scene point coordinates from the world reference frame to the camera reference frame using

$$P_c = [\overline{O_c P_w}]_c = I_c^w [\overline{O_c P_w}]_w, \quad (2.4)$$

where

$$I_c^w = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \text{ and}$$

$$[\overline{O_c P_w}]_w = \begin{bmatrix} X_w - X_o \\ Y_w - Y_o \\ Z_w - Z_o \end{bmatrix}_w.$$

Substitute I_c^w and $[\overline{O_c P_w}]_w$ into (2.4) to obtain

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix}_c = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{bmatrix} X_w - X_o \\ Y_w - Y_o \\ Z_w - Z_o \end{bmatrix}_w. \quad (2.5)$$

Step2 Project 3D Scene Points into Camera Image Plane

As shown in Figure 2.6, the DLT model carries out the second step using the collinearity condition. It applies a pinhole camera model and uses the following collinearity condition

$$[\overline{O_c P_u}]_c = \varepsilon [\overline{O_c P_w}]_c, \quad (2.6)$$

where vectors $[\overline{O_c P_u}]_c$ and $[\overline{O_c P_w}]_c$ are collinear and ε is a scaling factor. Using (2.5) and (2.6), the relationship between 3D and 2D points can be related by the transformation matrix I_c^w . Equation (2.6) can then be rewritten as

$$[\overline{O_c P_u}]_c = \varepsilon [\overline{O_c P_w}]_c = \varepsilon I_c^w [\overline{O_c P_w}]_w. \quad (2.7)$$

The camera frame's and image frame's axes are parallel to each other and have the same metric units. As shown in Figure 2.6, the image frame origin is located at the lower left

corner of the CCD plane and the camera frame origin is located at the projection centre along the optical axis of the camera. According to Figure 2.6, we know that the

undistorted image point is $[P_u]_i = [x_u \ y_u \ 0]_i^T$, and the intersection of the optical axis

with the image plane is $[P]_i = [x_{uo} \ y_{uo} \ 0]_i^T$. Hence, we know

that $[\overline{PP_u}]_i = [x_u - x_{uo} \ y_u - y_{uo} \ 0]_i^T$. But $[\overline{PP_u}]_i$ and $[\overline{PP_u}]_c$ represent the same vector

in the image frame and the camera frame respectively. Based on our assumption that the image and camera frames have the same units, the only difference is the location of their

origins. Therefore, we find $[\overline{PP_u}]_c = [x_u - x_{uo} \ y_u - y_{uo} \ 0]_c^T$ and

$$[\overline{O_c P_u}]_c = [\overline{O_c P}]_c + [\overline{PP_u}]_c = [0 \ 0 \ -f]_c^T + [x_u - x_{uo} \ y_u - y_{uo} \ 0]_c^T = [x_u - x_{uo} \ y_u - y_{uo} \ -f]_c^T$$

Now Equation (2.7) can be written into the following transformation equation.

$$\begin{bmatrix} x_u - x_{uo} \\ y_u - y_{uo} \\ -f \end{bmatrix} = \mathcal{E} \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{bmatrix} X_w - X_o \\ Y_w - Y_o \\ Z_w - Z_o \end{bmatrix} \quad (2.8)$$

Re-arranging terms in equation (2.8), we obtain

$$x_u - x_{uo} = \mathcal{E} [r_{11}(X_w - X_o) + r_{12}(Y_w - Y_o) + r_{13}(Z_w - Z_o)] \quad (2.9)$$

$$y_u - y_{uo} = \mathcal{E} [r_{21}(X_w - X_o) + r_{22}(Y_w - Y_o) + r_{23}(Z_w - Z_o)]$$

$$-f = \mathcal{E} [r_{31}(X_w - X_o) + r_{32}(Y_w - Y_o) + r_{33}(Z_w - Z_o)] \quad (2.10)$$

Solving (2.10) for $\mathcal{E}$ and using this in (2.9) gives

$$\begin{aligned} x_u - x_{uo} &= -f \frac{[r_{11}(X_w - X_o) + r_{12}(Y_w - Y_o) + r_{13}(Z_w - Z_o)]}{[r_{31}(X_w - X_o) + r_{32}(Y_w - Y_o) + r_{33}(Z_w - Z_o)]} \\ y_u - y_{uo} &= -f \frac{[r_{21}(X_w - X_o) + r_{22}(Y_w - Y_o) + r_{23}(Z_w - Z_o)]}{[r_{31}(X_w - X_o) + r_{32}(Y_w - Y_o) + r_{33}(Z_w - Z_o)]} \end{aligned} \quad (2.11)$$

Step3 Model lens distortion

The purpose of the third step is to model the lens distortions of the camera lenses. The distorted image points P_d are transformed from undistorted image points P_u according to the lens distortions defined by a specific camera model. The relationship between the distorted and undistorted image points are described by

$$P_d = P_u - \Delta r \quad (2.12)$$

where $P_u = (u, v)$, $P_d = (u_d, v_d)$, and Δr is the lens distortion expressed as $\Delta r = (\Delta u, \Delta v)$.

Now re-write equation (2.12) into

$$\begin{aligned} u_d &= u - \Delta u \\ v_d &= v - \Delta v \end{aligned} \quad (2.13)$$

The DLT and LMDLT methods implemented in this thesis do not consider lens distortions. Therefore, the distorted and non-distorted image points are located at the same position and the lens distortion Δu and Δv are both zero. Equation (2.13) becomes

$$\begin{aligned} u &= u_d \\ v &= v_d \end{aligned} \quad (2.14)$$

Step4 Coordinates transformation from Camera to Image Reference Frame

Undistorted image coordinates in equation (2.11) are described in the camera reference frame in units of meters. In this final step, these undistorted image coordinates have to be converted into coordinates described in the frame grabber reference frame in pixel units. The metric-to-pixel coordinate conversion is described as

$$\begin{aligned} [x_u]_c - [x_{uo}]_c &= \lambda_u ([u]_f - [u_o]_f) \\ [y_u]_c - [y_{uo}]_c &= \lambda_v ([v]_f - [v_o]_f), \end{aligned} \quad (2.15)$$

where λ_u and λ_v are the unit conversion factors and their units are meters/pixel.

Substitution of equation (2.15) into equation (2.11) gives

$$\begin{aligned} u - u_o &= -\frac{f [r_{11}(X_w - X_o) + r_{12}(Y_w - Y_o) + r_{13}(Z_w - Z_o)]}{\lambda_u [r_{31}(X_w - X_o) + r_{32}(Y_w - Y_o) + r_{33}(Z_w - Z_o)]} \\ v - v_o &= -\frac{f [r_{21}(X_w - X_o) + r_{22}(Y_w - Y_o) + r_{23}(Z_w - Z_o)]}{\lambda_v [r_{31}(X_w - X_o) + r_{32}(Y_w - Y_o) + r_{33}(Z_w - Z_o)]} \end{aligned} \quad (2.16)$$

Finally, rearranging equation (2.16), we get

$$\begin{aligned} u &= \frac{L_1 X_w + L_2 Y_w + L_3 Z_w + L_4}{L_9 X_w + L_{10} Y_w + L_{11} Z_w + 1} \\ v &= \frac{L_5 X_w + L_6 Y_w + L_7 Z_w + L_8}{L_9 X_w + L_{10} Y_w + L_{11} Z_w + 1} \end{aligned} \quad (2.17)$$

where (u, v) are the 2D image points in the frame buffer reference frame in pixels,

(X_w, Y_w, Z_w) are the scene points with respect to the world frame F_w in meters,

and L_1 through L_{11} are called the standard DLT parameters. From (2.16), we obtain

$$\begin{aligned} u &= \frac{u_o [r_{31}(X_w - X_o) + r_{32}(Y_w - Y_o) + r_{33}(Z_w - Z_o)] - \frac{f}{\lambda_u} [r_{11}(X_w - X_o) + r_{12}(Y_w - Y_o) + r_{13}(Z_w - Z_o)]}{r_{31}(X_w - X_o) + r_{32}(Y_w - Y_o) + r_{33}(Z_w - Z_o)} \\ &= \frac{(u_o r_{31} - \frac{f}{\lambda_u} r_{11})X_w + (u_o r_{32} - \frac{f}{\lambda_u} r_{12})Y_w + (u_o r_{33} - \frac{f}{\lambda_u} r_{13})Z_w - \left[(u_o r_{31} - \frac{f}{\lambda_u} r_{11})X_o + (u_o r_{32} - \frac{f}{\lambda_u} r_{12})Y_o + (u_o r_{33} - \frac{f}{\lambda_u} r_{13})Z_o \right]}{r_{31}X_w + r_{32}Y_w + r_{33}Z_w - (r_{31}X_o + r_{32}Y_o + r_{33}Z_o)} \\ &= \frac{(u_o r_{31} - \frac{f}{\lambda_u} r_{11})X_w + (u_o r_{32} - \frac{f}{\lambda_u} r_{12})Y_w + (u_o r_{33} - \frac{f}{\lambda_u} r_{13})Z_w - \left[(u_o r_{31} - \frac{f}{\lambda_u} r_{11})X_o + (u_o r_{32} - \frac{f}{\lambda_u} r_{12})Y_o + (u_o r_{33} - \frac{f}{\lambda_u} r_{13})Z_o \right]}{\frac{-(r_{31}X_o + r_{32}Y_o + r_{33}Z_o)}{r_{31}X_w + r_{32}Y_w + r_{33}Z_w} - (r_{31}X_o + r_{32}Y_o + r_{33}Z_o)} \end{aligned}$$

and

$$\begin{aligned} v &= \frac{(v_o r_{31} - \frac{f}{\lambda_v} r_{21})X_w + (v_o r_{32} - \frac{f}{\lambda_v} r_{22})Y_w + (v_o r_{33} - \frac{f}{\lambda_v} r_{23})Z_w - \left[(v_o r_{31} - \frac{f}{\lambda_v} r_{21})X_o + (v_o r_{32} - \frac{f}{\lambda_v} r_{22})Y_o + (v_o r_{33} - \frac{f}{\lambda_v} r_{23})Z_o \right]}{\frac{-(r_{31}X_o + r_{32}Y_o + r_{33}Z_o)}{r_{31}X_w + r_{32}Y_w + r_{33}Z_w} - (r_{31}X_o + r_{32}Y_o + r_{33}Z_o)} \end{aligned}$$

Finally, we define the DLT parameters as³

³ <http://www.kwon3d.com/theory/dlt>

$$\begin{aligned}
L_1 &= \frac{u_o r_{31} - t_u r_{11}}{d}, \\
L_2 &= \frac{u_o r_{32} - t_u r_{12}}{d}, \\
L_3 &= \frac{u_o r_{33} - t_u r_{13}}{d}, \\
L_4 &= \frac{(t_u r_{11} - u_o r_{31})X_o + (t_u r_{12} - u_o r_{32})Y_o + (t_u r_{13} - u_o r_{33})Z_o}{d}, \\
L_5 &= \frac{v_o r_{31} - t_v r_{21}}{d}, \\
L_6 &= \frac{v_o r_{32} - t_v r_{22}}{d}, \\
L_7 &= \frac{v_o r_{33} - t_v r_{23}}{d}, \\
L_8 &= \frac{(t_v r_{21} - v_o r_{31})X_o + (t_v r_{22} - v_o r_{32})Y_o + (t_v r_{23} - v_o r_{33})Z_o}{d}, \\
L_9 &= \frac{r_{31}}{d}, \\
L_{10} &= \frac{r_{32}}{d}, \text{ and} \\
L_{11} &= \frac{r_{33}}{d}
\end{aligned} \tag{2.18}$$

where $t_u = \frac{f}{\lambda_u}$ and $t_v = \frac{f}{\lambda_v}$, and $d = -(r_{31}X_o + r_{32}Y_o + r_{33}Z_o)$.

2.8.2 Tsai's Camera Model

O_w = origin of world reference frame
 $O_c(X_o, Y_o, Z_o)$ = origin of camera reference frame, centre of projection
 O_i = image frame origin
 $P_w(X_w, Y_w, Z_w)$ = scene points coordinates in world reference frame
 $P_c(X_c, Y_c, Z_c)$ = scene points coordinates in camera reference frame
 P_u = undistorted image coordinates
 P_d = distorted image coordinates

Figure 2.7: The mapping of 3D scene points and 2D image points for Tsai's method

Step1 Change Scene Points Reference Frame from World frame to Camera Frame

Tsai's method transfers the 3D scene point coordinates from the world reference frame to the camera reference frame using equation (2.19),

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = I_c^w \begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} + \begin{bmatrix} T_w^c \end{bmatrix}_c \quad (2.19)$$

where $P_c = [X_c \ Y_c \ Z_c]^T$ represents the scene points coordinates in the camera reference frame, $P_w = [X_w \ Y_w \ Z_w]^T$ represents the scene point coordinates in the world frame,

$$I_c^w = \begin{bmatrix} r_1 & r_2 & r_3 \\ r_4 & r_5 & r_6 \\ r_7 & r_8 & r_9 \end{bmatrix}$$

is the rotation matrix from the world reference frame to the camera reference frame, and

$[T_w^c]_c = [T_x \ T_y \ T_z]^T$ is the translation matrix from the camera reference frame to the world reference frame described in the camera reference frame.

Step2 Project 3D Scene Points into the camera image plane

According to Figure 2.7, the image plane is parallel to the plane defined by the x and y axis of the camera reference frame. The image plane is f meters away from the origin of the camera reference frame. The projection center for the perspective projection is located at the origin of the camera reference frame. Using perspective projection, a 3D scene point P_c in the camera reference frame is projected into the image plane at location P_u . This mapping is described by

$$\begin{aligned} x_u &= f \frac{X_c}{Z_c} \\ y_u &= f \frac{Y_c}{Z_c} \end{aligned} \tag{2.20}$$

where (x_u, y_u) are the undistorted 2D image points in the image reference frame

and (X_c, Y_c, Z_c) are the 3D scene point coordinates in the camera reference frame.

Step3 Model lens distortions

The purpose of the third step is to model lens distortions and find the distorted image points. The distorted image points are transformed from undistorted image points using

$$\begin{aligned}x_d &= x_u - \Delta u \\ y_d &= y_u - \Delta v\end{aligned}\tag{2.21}$$

where (x_d, y_d) are the distorted image points described in the camera reference frame in metric units and (x_u, y_u) are the undistorted image points. Tsai's method models the radial lens distortion only and defines the radial distortion $(\Delta u, \Delta v)$ to be [5]

$$\begin{aligned}\Delta u &= x_d(k_1 r^2 + k_2 r^4) \\ \Delta v &= y_d(k_1 r^2 + k_2 r^4),\end{aligned}\tag{2.22}$$

where $r = \sqrt{x_d^2 + y_d^2}$.

Step4 Coordinate transformation from Camera to Image Reference Frame

In the last step, the distorted image points are transformed from a metric coordinate system into a pixel coordinate system using

$$\begin{aligned}u &= s_x (\widetilde{d}_x)^{-1} x_d + u_o \\ v &= \widetilde{d}_y^{-1} y_d + v_o\end{aligned}\tag{2.23}$$

where (u, v) are 2D image points represented in the frame buffer reference frame in pixel units, (x_d, y_d) are the distorted image points in meters, (u_o, v_o) is the principal point in units of pixels, s_x is the uncertainty image scale factor, and $\widetilde{d}_x$ and $\widetilde{d}_y$ are the center to

center distances between adjacent CCD sensor elements in the x -direction and y -direction. The symbol $\widetilde{d}_x$ is defined as

$$\widetilde{d}_x = d_x \frac{N_{cx}}{N_{fx}} \quad (2.24)$$

where N_{cx} is the number of CCD sensor elements in the x direction and N_{fx} is the number of pixels sampled by the frame grabber in a horizontal line.

2.9 Camera Calibration Methods

According to their specific camera models defined in the previous sections, camera calibration algorithms are derived. Camera calibration methods derived from linear camera models are usually simpler than non-linear methods. This is because linear calibration methods can be solved easily using the least squares approach. Nonlinear methods require a good initial guess and use an iterative approach to obtain optimal solutions. If the initial guess is not accurate enough, the method may result in numerical stability problems. Examples of linear calibration methods are the DLT method [4] and Faugeras' method without lens distortion [6]. Non-linear methods are Tsai's [5] and Weng's [8] methods. Non-linear methods use an iterative process to find optimal solutions and hence are much slower than linear calibration methods.

In this Section, the classic DLT method [4], Linear Modified DLT method [23], and Tsai's method [5] are reviewed and compared. The rotation matrix I_c^w has nine elements, but only three degrees of freedom (roll, pitch, and yaw angles are used as parameters in this study). These nine unknowns are augmented with six constraints resulting from the "orthogonality constraint" that require orthogonal rows (and columns)

within I_c^w and that $\det(I_c^w) = +1$. The condition $\det(I_c^w) = +1$ is often enforced by estimating only two rows (or columns) of I_c^w , enforcing the orthogonality constraint, and then computing the remaining row from the cross product of the two estimated rows, and normalizing to preserve orthogonality of I_c^w . The cross product is arranged to ensure that $\det(I_c^w) = +1$ ⁴.

The classic DLT method does not include an orthogonal constraint for the rotation matrix I_c^w and the estimated parameters are not very accurate. The Linear Modified DLT method [23] was proposed to address this issue. More detailed discussions on this issue can be found in Sections 2.9.1 and 2.9.2.

2.9.1 Linear DLT Camera Calibration Algorithm

The basis of the classic DLT method is equation (2.17) defined by the DLT camera model. Rearranging terms in (2.17), we obtain

$$\begin{aligned} u &= L_1 X_w + L_2 Y_w + L_3 Z_w + L_4 - L_9 u X_w - L_{10} u Y_w - L_{11} u Z_w \\ v &= L_5 X_w + L_6 Y_w + L_7 Z_w + L_8 - L_9 v X_w - L_{10} v Y_w - L_{11} v Z_w. \end{aligned} \quad (2.25)$$

One can further change the equation above into matrix form as

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} X_w & Y_w & Z_w & 1 & 0 & 0 & 0 & 0 & -uX_w & -uY_w & -uZ_w \\ 0 & 0 & 0 & 0 & X_w & Y_w & Z_w & 1 & -vX_w & -vY_w & -vZ_w \end{bmatrix} \begin{bmatrix} L_1 \\ L_2 \\ \vdots \\ L_{10} \\ L_{11} \end{bmatrix}. \quad (2.26)$$

If there are n 3D scene points available, using equation (2.26), one can get the following system of equations.

⁴ An orthogonal matrix with determinant of negative one is a reflection matrix, not a rotation matrix.

$$\begin{bmatrix} u_1 \\ v_1 \\ \vdots \\ u_n \\ v_n \end{bmatrix} = \begin{bmatrix} X_{w_1} & Y_{w_1} & Z_{w_1} & 1 & 0 & 0 & 0 & 0 & -u_1 X_{w_1} & -u_1 Y_{w_1} & -u_1 Z_{w_1} \\ 0 & 0 & 0 & 0 & X_{w_1} & Y_{w_1} & Z_{w_1} & 1 & -v_1 X_{w_1} & -v_1 Y_{w_1} & -v_1 Z_{w_1} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ X_{w_n} & Y_{w_n} & Z_{w_n} & 1 & 0 & 0 & 0 & 0 & -u_n X_{w_n} & -u_n Y_{w_n} & -u_n Z_{w_n} \\ 0 & 0 & 0 & 0 & X_{w_n} & Y_{w_n} & Z_{w_n} & 1 & -v_n X_{w_n} & -v_n Y_{w_n} & -v_n Z_{w_n} \end{bmatrix} \begin{bmatrix} L_1 \\ L_2 \\ \vdots \\ L_{10} \\ L_{11} \end{bmatrix} \quad (2.27)$$

The n scene point coordinates $(X_{w_n}, Y_{w_n}, Z_{w_n})$ and their camera-captured image point coordinates (u_n, v_n) are all known. Therefore, the system above can be solved using the least squares technique. First, the above equation is rewritten into the following form,

$$AL = B, \quad (2.28)$$

where

$$A = \begin{bmatrix} X_{w_1} & Y_{w_1} & Z_{w_1} & 1 & 0 & 0 & 0 & 0 & -u_1 X_{w_1} & -u_1 Y_{w_1} & -u_1 Z_{w_1} \\ 0 & 0 & 0 & 0 & X_{w_1} & Y_{w_1} & Z_{w_1} & 1 & -v_1 X_{w_1} & -v_1 Y_{w_1} & -v_1 Z_{w_1} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ X_{w_n} & Y_{w_n} & Z_{w_n} & 1 & 0 & 0 & 0 & 0 & -u_n X_{w_n} & -u_n Y_{w_n} & -u_n Z_{w_n} \\ 0 & 0 & 0 & 0 & X_{w_n} & Y_{w_n} & Z_{w_n} & 1 & -v_n X_{w_n} & -v_n Y_{w_n} & -v_n Z_{w_n} \end{bmatrix}$$

$$L = \begin{bmatrix} L_1 \\ L_2 \\ \vdots \\ L_{10} \\ L_{11} \end{bmatrix}$$

$$B = \begin{bmatrix} u_1 \\ v_1 \\ \vdots \\ u_n \\ v_n \end{bmatrix}$$

Therefore, the least squares solution for DLT parameters are found to be

$$L = (A^T A)^{-1} (A^T B). \quad (2.29)$$

Equation (2.29) can only be solved when we have an over determined system, that is the number of unknowns must be less or equal to total number of equations. Hence, more than 11 equations are required which implies that more than 6 scene points are needed.

2.9.2 Linear Modified DLT Camera Calibration Algorithm

The classic DLT method can estimate parameters for the linear DLT camera model, but unfortunately it has problems. In equation (2.16), there are actually only 10 unknowns instead of 11. This is because one can define parameters $t_u = \frac{f}{\lambda_u}$, $t_v = \frac{f}{\lambda_v}$ to reduce focal length f , λ_u , and λ_v into two variables t_u and t_v [23]. After this reduction, equation (2.16) becomes

$$\begin{aligned} u - u_o &= -t_u \frac{[r_{11}(X_w - X_o) + r_{12}(Y_w - Y_o) + r_{13}(Z_w - Z_o)]}{[r_{31}(X_w - X_o) + r_{32}(Y_w - Y_o) + r_{33}(Z_w - Z_o)]} \\ v - v_o &= -t_v \frac{[r_{21}(X_w - X_o) + r_{22}(Y_w - Y_o) + r_{23}(Z_w - Z_o)]}{[r_{31}(X_w - X_o) + r_{32}(Y_w - Y_o) + r_{33}(Z_w - Z_o)]} \end{aligned} \quad (2.30)$$

It can be observed clearly that the 10 unknowns are $u_o, v_o, X_o, Y_o, Z_o, t_u, t_v$ and Euler angles roll-pitch-yaw. Equation (2.17) and (2.30) both represent the same system of equations but (2.17) has 11 DLT parameters and (2.30) has 10 unknowns. Therefore, one of the 11 DLT parameters must be dependent on the others. When solving equation (2.17) using the least squares method described in Section 2.9.1, this constraint was ignored and the orthogonality for the transformation matrix I_c^w in equation (2.4) is also not ensured.

Hatze [23] has proposed two modified DLT methods, Linear and Nonlinear Modified DLT Methods, to correct this problem. His modified DLT algorithms improve the accuracy of back-projection compared with the classic DLT method. The Linear

modified DLT method (LMDLT) improves a camera calibration result by satisfying the orthogonality constraint of the transformation matrix I_c^w in equation (2.4). In Hatze's nonlinear modified DLT (NLMDLT), the calibration accuracy and stability were improved by properly eliminating implicit variables from the left side of (2.30) and reformulating the objective function to be minimized [23]. This thesis focuses on his linear modified DLT method. In LMDLT method, Hatze first uses an orthogonality constraint for the rotation matrix I_c^w to find a nonlinear equation describing the relationship between the DLT parameters. He then expresses one DLT parameter in terms of the other 10 parameters based on this relationship. Because this relationship is non-linear, a linearization step is required in order to solve the system of equations. This linearization step is explained in detail later in this section. After linearizing the system, the equations can be solved iteratively using the linear least squares method for the 10 DLT parameters.

The following is a summary of LMDLT method proposed by Hatze [23]:

1. Use the classic DLT method to estimate the 11 DLT parameters to be used as an initial guess;
2. Apply the orthogonality constraint for the rotation matrix I_c^w and find the nonlinear relationship between the 11 DLT parameters;
3. Choose one DLT parameter and express it in terms of the other 10 DLT parameters;
4. Use the result from step 3 to eliminate one DLT parameter from the system of equations and keep only 10 DLT parameters;
5. Linearize the system of equations with 10 DLT parameters;

6. Apply the linear least squares method to solve the system for the 10 DLT parameters;
7. Calculate the parameter eliminated in step 4 using the relationship identified in step 3 based on the 10 parameters found in step 6; and
8. Repeat step 3 to step 7 until convergence to final DLT parameters. The intrinsic and extrinsic camera parameters can be recovered from the final DLT parameters.

Step 1 Use classic DLT method to estimate the 11 DLT parameters to be used as an initial guess

This step is the same as the classic DLT method described in section 2.9.1.

Step 2 Apply the orthogonality constraint for the rotation matrix I_c^w and find the nonlinear relationship between the 11 DLT parameters

From the DLT parameters defined in equation (2.18), obtain

$$\begin{aligned}
(L_1 d)(L_9 d) &= (u_o r_{31} - t_u r_{11}) r_{31} = u_o r_{31}^2 - t_u r_{11} r_{31} \\
(L_2 d)(L_{10} d) &= (u_o r_{32} - t_u r_{12}) r_{32} = u_o r_{32}^2 - t_u r_{12} r_{32} \\
(L_3 d)(L_{11} d) &= (u_o r_{33} - t_u r_{13}) r_{33} = u_o r_{33}^2 - t_u r_{13} r_{33} \\
(L_5 d)(L_9 d) &= (v_o r_{31} - t_v r_{21}) r_{31} = v_o r_{31}^2 - t_v r_{21} r_{31} \\
(L_6 d)(L_{10} d) &= (v_o r_{32} - t_v r_{22}) r_{32} = v_o r_{32}^2 - t_v r_{22} r_{32} \\
(L_7 d)(L_{11} d) &= (v_o r_{33} - t_v r_{23}) r_{33} = v_o r_{33}^2 - t_v r_{23} r_{33}
\end{aligned} \tag{2.31}$$

Combine these terms to get

$$\begin{aligned}
&(L_1 d)(L_9 d) + (L_2 d)(L_{10} d) + (L_3 d)(L_{11} d) \\
&= (r_{31}^2 + r_{32}^2 + r_{33}^2) u_o - (r_{11} r_{31} + r_{12} r_{32} + r_{13} r_{33}) t_u \\
&(L_5 d)(L_9 d) + (L_6 d)(L_{10} d) + (L_7 d)(L_{11} d) \\
&= (r_{31}^2 + r_{32}^2 + r_{33}^2) v_o - (r_{21} r_{31} + r_{22} r_{32} + r_{23} r_{33}) t_v
\end{aligned} \tag{2.32}$$

Now apply the orthogonality constraints for the transformation matrix I_c^w

$$\begin{aligned}\sum_{k=1}^3 r_{mk} r_{nk} &= \delta_{mn} \\ \sum_{k=1}^3 r_{km} r_{kn} &= \delta_{mn}\end{aligned}\tag{2.33}$$

$$\text{where } \delta_{mn} = \begin{cases} 1, m = n \\ 0, m \neq n \end{cases}.$$

When $m = 3, n = 3$, equation (2.33) becomes

$$\sum_{k=1}^3 r_{mk} r_{nk} = \sum_{k=1}^3 r_{3k} r_{3k} = r_{31}^2 + r_{32}^2 + r_{33}^2 = \delta_{33} = 1.$$

When $m = 1, n = 3$, equation (2.33) becomes

$$\sum_{k=1}^3 r_{mk} r_{nk} = \sum_{k=1}^3 r_{1k} r_{3k} = r_{11} r_{31} + r_{12} r_{32} + r_{13} r_{33} = \delta_{13} = 0.$$

When $m = 2, n = 3$, equation (2.33) becomes

$$\sum_{k=1}^3 r_{mk} r_{nk} = \sum_{k=1}^3 r_{2k} r_{3k} = r_{21} r_{31} + r_{22} r_{32} + r_{23} r_{33} = \delta_{23} = 0.$$

Hence, one can reduce equation (2.32) to

$$\begin{aligned}(L_1 d)(L_9 d) + (L_2 d)(L_{10} d) + (L_3 d)(L_{11} d) &= u_o \\ (L_5 d)(L_9 d) + (L_6 d)(L_{10} d) + (L_7 d)(L_{11} d) &= v_o\end{aligned}\tag{2.34}$$

Similarly,

$$\begin{aligned}&(L_1 d)(L_5 d) + (L_2 d)(L_6 d) + (L_3 d)(L_7 d) \\ &= (u_o r_{31} - t_u r_{11})(v_o r_{31} - t_v r_{21}) + (u_o r_{32} - t_u r_{12})(v_o r_{32} - t_v r_{22}) + (u_o r_{33} - t_u r_{13})(v_o r_{31} - t_v r_{23}) \\ &= u_o v_o\end{aligned}\tag{2.35}$$

Rearrange the left hand sides of the equations in (2.34); multiply these two equations

together to get⁵

$$(L_1 L_9 + L_2 L_{10} + L_3 L_{11})(L_5 L_9 + L_6 L_{10} + L_7 L_{11}) d^4 = u_o v_o\tag{2.36}$$

⁵ Note that due to orthogonality constraint $L_9^2 + L_{10}^2 + L_{11}^2 = \frac{1}{d^2}$.

Finally, combining equations (2.35) and (2.36) into the following equation that describes the nonlinear relationship between the 11 standard DLT parameters based on the orthogonality constraints.

$$(L_1L_9 + L_2L_{10} + L_3L_{11})(L_5L_9 + L_6L_{10} + L_7L_{11}) = (L_1L_5 + L_2L_6 + L_3L_7)(L_9^2 + L_{10}^2 + L_{11}^2) \quad (2.37)$$

Step 3 Choose one DLT parameter and express it in terms of the other 10 DLT parameters

Rewrite (2.37) and obtain

$$\begin{aligned} L_1 &= \Phi(L_2 \cdots L_{11}) \\ &= \frac{(-L_2L_{10}L_5L_9 - L_2L_{10}L_7L_{11} - L_3L_{11}L_5L_9 - L_3L_{11}L_6L_{10} + L_2L_6L_9^2 + L_2L_6L_{11}^2 + L_3L_7L_9^2 + L_3L_7L_{10}^2)}{(L_9L_6L_{10} + L_9L_7L_{11} - L_5L_{10}^2 - L_5L_{11}^2)} \end{aligned} \quad (2.38)$$

Step 4 Use the result from step 3 to eliminate one DLT parameter from the system of equations and keep only 10 DLT parameters

Substitute equation (2.38) into equation (2.17) and obtain the following system of nonlinear equations

$$\begin{aligned} u &= \frac{\Phi(L_2 \cdots L_{11})X_w + L_2Y_w + L_3Z_w + L_4}{L_9X_w + L_{10}Y_w + L_{11}Z_w + 1} \\ v &= \frac{L_5X_w + L_6Y_w + L_7Z_w + L_8}{L_9X_w + L_{10}Y_w + L_{11}Z_w + 1} \end{aligned} \quad (2.39)$$

Step 5 Linearize the system of equations with 10 DLT parameters

Because the above equation is nonlinear, a linear least squares method cannot be used here. Although there are many non-linear least squares methods available, a linearization technique was used [24][25]. The linearization technique introduced in [24][25] was used to solve the nonlinear system described by equation (2.39). Using a

truncated Taylor series expansion to linearize the system at the current parameter

values $(L_2^{cur}, L_3^{cur}, \dots, L_{11}^{cur})$, we obtain

$$\begin{aligned} u &= u^{cur} + \frac{\partial u^{cur}}{\partial L_2} (L_2^{new} - L_2^{cur}) + \frac{\partial u^{cur}}{\partial L_3} (L_3^{new} - L_3^{cur}) + \dots + \frac{\partial u^{cur}}{\partial L_{11}} (L_{11}^{new} - L_{11}^{cur}) \\ v &= v^{cur} + \frac{\partial v^{cur}}{\partial L_2} (L_2^{new} - L_2^{cur}) + \frac{\partial v^{cur}}{\partial L_3} (L_3^{new} - L_3^{cur}) + \dots + \frac{\partial v^{cur}}{\partial L_{11}} (L_{11}^{new} - L_{11}^{cur}) \end{aligned} \quad (2.40)$$

where the partial derivatives are calculated with the current parameter estimates.

With a total of n control points, we can re-write (2.40) into matrix form as follows,

$$\begin{bmatrix} u_1 - u_1^{cur} \\ v_1 - v_1^{cur} \\ u_2 - u_2^{cur} \\ v_2 - v_2^{cur} \\ \vdots \\ u_n - u_n^{cur} \\ v_n - v_n^{cur} \end{bmatrix} = \begin{bmatrix} \frac{\partial u_1^{cur}}{\partial L_2} & \frac{\partial u_1^{cur}}{\partial L_3} & \dots & \frac{\partial u_1^{cur}}{\partial L_{11}} \\ \frac{\partial v_1^{cur}}{\partial L_2} & \frac{\partial v_1^{cur}}{\partial L_3} & \dots & \frac{\partial v_1^{cur}}{\partial L_{11}} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial v_n^{cur}}{\partial L_2} & \frac{\partial v_n^{cur}}{\partial L_3} & \dots & \frac{\partial v_n^{cur}}{\partial L_{11}} \end{bmatrix} \begin{bmatrix} L_2^{new} - L_2^{cur} \\ L_3^{new} - L_3^{cur} \\ \vdots \\ L_{11}^{new} - L_{11}^{cur} \end{bmatrix} \quad (2.41)$$

We can then simplify (2.41) into

$$A = B \Delta L \quad (2.42)$$

$$\text{where } A = \begin{bmatrix} u_1 - u_1^{cur} \\ v_1 - v_1^{cur} \\ u_2 - u_2^{cur} \\ v_2 - v_2^{cur} \\ \vdots \\ u_n - u_n^{cur} \\ v_n - v_n^{cur} \end{bmatrix}, B = \begin{bmatrix} \frac{\partial u_1^{cur}}{\partial L_2} & \frac{\partial u_1^{cur}}{\partial L_3} & \dots & \frac{\partial u_1^{cur}}{\partial L_{11}} \\ \frac{\partial v_1^{cur}}{\partial L_2} & \frac{\partial v_1^{cur}}{\partial L_3} & \dots & \frac{\partial v_1^{cur}}{\partial L_{11}} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial v_n^{cur}}{\partial L_2} & \frac{\partial v_n^{cur}}{\partial L_3} & \dots & \frac{\partial v_n^{cur}}{\partial L_{11}} \end{bmatrix}, \Delta L = \begin{bmatrix} L_2^{new} - L_2^{cur} \\ L_3^{new} - L_3^{cur} \\ \vdots \\ L_{11}^{new} - L_{11}^{cur} \end{bmatrix}$$

Step 6 Apply the linear least squares method to solve the system for the 10 DLT parameters

After the system has been linearized, the standard linear least squares technique can be applied to get,

$$\Delta L = (B^T B)^{-1} B^T A. \quad (2.43)$$

The solution for the 10 DLT parameters can be found by calculating

$$\begin{aligned} L_2^{new} &= L_2^{cur} + \Delta L_2 \\ L_3^{new} &= L_3^{cur} + \Delta L_3 \\ &\vdots \\ L_{11}^{new} &= L_{11}^{cur} + \Delta L_{11} \end{aligned} \quad (2.44)$$

Step 7 Calculate the parameter eliminated in step 4 using the relationship identified in step 3 based on the 10 parameters found in step 6

The parameter eliminated in step 4 can now be calculated using (2.38) by substituting the 10 parameter estimates found in step 6.

Step 8 Repeat step 3 to step 7 until convergence to final DLT parameters

Step 3 to step 7 can be repeated using the new estimates of DLT parameters found in step 6 and step 7 as the current estimates and continue until a set of converged DLT parameters are found. The iterations are terminated when ΔL is small enough to represent an insignificant update in parameter estimates. Convergence properties of this approach are presented by Draper and Smith [25]. They note that with reasonable initial estimates, convergence will usually occur, however, for some systems, this process may diverge.

After finding the 11 DLT parameters, now we can proceed to recover the camera position and the image centre. From (2.18) we can obtain

$$\begin{aligned}
& L_4 \\
& = \frac{(t_u r_{11} - u_o r_{31})X_o + (t_u r_{12} - u_o r_{32})Y_o + (t_u r_{13} - u_o r_{33})Z_o}{d}, \\
& = \frac{t_u r_{11} - u_o r_{31}}{d} X_o + \frac{t_u r_{12} - u_o r_{32}}{d} Y_o + \frac{t_u r_{13} - u_o r_{33}}{d} Z_o \\
& = -(L_1 X_o + L_2 Y_o + L_3 Z_o)
\end{aligned} \tag{2.45}$$

$$\begin{aligned}
& L_8 \\
& = \frac{(t_v r_{21} - v_o r_{31})X_o + (t_v r_{22} - v_o r_{32})Y_o + (t_v r_{23} - v_o r_{33})Z_o}{d}, \\
& = \frac{v_o r_{31} - t_v r_{21}}{d} X_o + \frac{v_o r_{32} - t_v r_{22}}{d} Y_o + \frac{v_o r_{33} - t_v r_{23}}{d} Z_o \\
& = -(L_5 X_o + L_6 Y_o + L_7 Z_o)
\end{aligned} \tag{2.46}$$

and

$$\begin{aligned}
& d = -(r_{31}X_o + r_{32}Y_o + r_{33}Z_o) \\
& \Rightarrow -\left(\frac{r_{31}}{d}X_o + \frac{r_{32}}{d}Y_o + \frac{r_{33}}{d}Z_o\right) = 1. \\
& \Rightarrow -(L_9 X_o + L_{10} Y_o + L_{11} Z_o) = 1
\end{aligned} \tag{2.47}$$

We can combine (2.45), (2.46), and (2.47) to recover the camera position using

$$\begin{bmatrix} X_o & Y_o & Z_o \end{bmatrix}^T = \Lambda^{-1} \mathbf{M}, \tag{2.48}$$

where

$$\Lambda = \begin{bmatrix} L_1 & L_2 & L_3 \\ L_4 & L_5 & L_6 \\ L_7 & L_8 & L_9 \end{bmatrix},$$

$$\text{and } \mathbf{M} = -\begin{bmatrix} L_4 & L_8 & 1 \end{bmatrix}^T.$$

The principal point (u_o, v_o) is calculated from (2.34).

Next from (2.18), we can obtain

$$\begin{aligned}
& L_1 \\
&= \frac{u_o r_{31} - t_u r_{11}}{d} \\
&= u_o L_9 - t_u \left(\frac{r_{11}}{d} \right) \\
&\Rightarrow r_{11} = \frac{d(u_o L_9 - L_1)}{t_u}
\end{aligned} \tag{2.49}$$

Similarly, the elements of the rotation matrix can be expressed by

$$\begin{aligned}
r_{12} &= \frac{d(u_o L_{10} - L_2)}{t_u}, \\
r_{13} &= \frac{d(u_o L_{11} - L_3)}{t_u}, \\
r_{21} &= \frac{d(v_o L_9 - L_5)}{t_v}, \\
r_{22} &= \frac{d(v_o L_{10} - L_6)}{t_v}, \\
r_{23} &= \frac{d(v_o L_{11} - L_7)}{t_v}, \\
r_{31} &= L_9 d, \\
r_{32} &= L_{10} d, \text{ and} \\
r_{33} &= L_{11} d.
\end{aligned} \tag{2.50}$$

Next, we apply the orthogonality constraint defined in (2.33) to compute the values for

t_u and t_v using the following

$$\begin{aligned}
& r_{11}^2 + r_{12}^2 + r_{13}^2 = 1 \\
& \Rightarrow d^2 \left(\frac{u_o L_9 - L_1}{t_u} \right)^2 + d^2 \left(\frac{u_o L_{10} - L_2}{t_u} \right)^2 + d^2 \left(\frac{u_o L_{11} - L_3}{t_u} \right)^2 = 1, \\
& \Rightarrow t_u = \text{sign}(d) |d| \sqrt{(u_o L_9 - L_1)^2 + (u_o L_{10} - L_2)^2 + (u_o L_{11} - L_3)^2}
\end{aligned}$$

and

$$\begin{aligned}
r_{21}^2 + r_{22}^2 + r_{23}^2 &= 1 \\
\Rightarrow d^2 \left(\frac{v_o L_9 - L_5}{t_v} \right)^2 + d^2 \left(\frac{v_o L_{10} - L_6}{t_v} \right)^2 + d^2 \left(\frac{v_o L_{11} - L_7}{t_v} \right)^2 &= 1, \\
\Rightarrow t_v &= \text{sign}(d) |d| \sqrt{(v_o L_9 - L_5)^2 + (v_o L_{10} - L_6)^2 + (v_o L_{11} - L_7)^2}
\end{aligned}$$

$$\text{where } |d| = \sqrt{\frac{1}{L_9^2 + L_{10}^2 + L_{11}^2}}.$$

Let $t_u > 0$ and $t_v > 0$, calculate the following determinant D,

$$D = \begin{vmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{vmatrix}.$$

The sign of d is determined using the test below.

If ($D > 0$) {
Right hand system
 $d > 0$
 $t_u > 0$
 $t_v > 0$ }
Else if ($D < 0$) {
Left hand system
 $d < 0$
 $t_u < 0$
 $t_v < 0$ }

Now we know the signs and magnitudes for t_u and t_v . Substituting t_u , t_v , the principal point (u_o, v_o) , and the 11 DLT parameters into (2.49) and (2.50), the rotation matrix is found.

2.9.3 Tsai's Non-coplanar Camera Calibration Algorithm

Nonlinear camera calibration methods usually require a large number of iterations to obtain accurate parameter estimation. To improve estimation performance, Tsai proposed a two-step method. Tsai's camera calibration method uses a parallelism

constraint and assumes that some camera parameters are available from the manufacturers. These assumptions reduce the number of parameters that need to be computed in the first step and result in faster parameter convergence in the second step. Tsai's non-coplanar camera calibration algorithm used in this thesis is based on V. C. Aitken's implementation [19] to make Tsai's algorithm more accurate. In Tsai's first step, a linear optimization technique is used to find some of the camera parameters. Aitken added an additional step after Tsai's first step. In this additional step, a constrained minimization is performed to orthogonalize the estimated rotation matrix I_c^w . All step1 parameters are also updated in this additional step. In the second step, parameters estimated in Aitken's additional step are used as an initial guess for an iterative nonlinear optimization process to compute the rest of the parameters. However, parameters obtained in step 1 are held fixed in the second step. Only parameters such as focal length f , distortion coefficients k_1 and k_2 , and the T_z component of the translation matrix are optimized in the second step.

In step 1 of Tsai's algorithm, the following parameters are estimated:

1. Rotation matrix I_c^w ;
2. Translation matrix components T_x and T_y ; and
3. Horizontal scale factor s_x .

In step 2, Tsai's method computes the parameters below:

1. Effective focal length f ;
2. Distortion coefficients k_1 and k_2 ; and
3. T_z component of the translation vector.

Step 1 of Tsai's Algorithm

In step 1, Tsai uses the “parallelism constraint”. Tsai assumes that the camera lens only has radial lens distortion. Hence, according to the perspective projection model in

Figure 2.7, the vectors $\overline{PP_d}$ and $\overline{P'P_w}$, are parallel to each other where P is the centre of the image, and $\overline{P'P_w}$ is perpendicular to the optical axis. The cross product of two parallel vectors is zero. Hence, we can write

$$\overline{PP_d} \times \overline{P'P_w} = 0 \quad (2.51)$$

where $\overline{PP_d} = (x_d, y_d)$ and $\overline{P'P_w} = (X_c, Y_c)$.

From (2.51), we obtain

$$x_d Y_c = y_d X_c. \quad (2.52)$$

From (2.19), one can get

$$\begin{aligned} X_c &= r_1 X_w + r_2 Y_w + r_3 Z_w + T_x \\ Y_c &= r_4 X_w + r_5 Y_w + r_6 Z_w + T_y \end{aligned} \quad (2.53)$$

Substituting X_c and Y_c into (2.52), we get

$$x_d (r_4 X_w + r_5 Y_w + r_6 Z_w + T_y) = y_d (r_1 X_w + r_2 Y_w + r_3 Z_w + T_x) \quad (2.54)$$

Equation (2.54) is rearranged into matrix form to get

$$x_d = \begin{bmatrix} y_d X_w & y_d Y_w & y_d Z_w & y_d & -x_d X_w & -x_d Y_w & -x_d Z_w \end{bmatrix} \begin{bmatrix} r_1 T_y^{-1} \\ r_2 T_y^{-1} \\ r_3 T_y^{-1} \\ T_x T_y^{-1} \\ r_4 T_y^{-1} \\ r_5 T_y^{-1} \\ r_6 T_y^{-1} \end{bmatrix}. \quad (2.55)$$

Note that it is assumed that $T_y \neq 0$.⁶

The distorted image points (x_d, y_d) can be calculated by rewriting (2.23) into

$$\begin{aligned} x_d &= \frac{\tilde{d}_x(u - u_o)}{s_x} \\ y_d &= d_y(v - v_o) \end{aligned} \quad (2.56)$$

Substitute x_d and y_d into (2.55), rearrange terms and obtain

$$\begin{bmatrix} d_y(v - v_o)X_w & d_y(v - v_o)Y_w & d_y(v - v_o)Z_w & d_y(v - v_o) & -\tilde{d}_x(u - u_o)X_w & -\tilde{d}_x(u - u_o)Y_w & -\tilde{d}_x(u - u_o)Z_w \end{bmatrix} \begin{bmatrix} s_x r_1 T_y^{-1} \\ s_x r_2 T_y^{-1} \\ s_x r_3 T_y^{-1} \\ s_x T_x T_y^{-1} \\ r_4 T_y^{-1} \\ r_5 T_y^{-1} \\ r_6 T_y^{-1} \end{bmatrix} = \tilde{d}_x(u - u_o) \quad (2.57)$$

If m scene points are used in the calibration, we can define

$$A = \begin{bmatrix} d_y(v^{(1)} - v_o)X_w^{(1)} & d_y(v^{(1)} - v_o)Y_w^{(1)} & d_y(v^{(1)} - v_o)Z_w^{(1)} & d_y(v^{(1)} - v_o) & -\tilde{d}_x(u^{(1)} - u_o)X_w^{(1)} & -\tilde{d}_x(u^{(1)} - u_o)Y_w^{(1)} & -\tilde{d}_x(u^{(1)} - u_o)Z_w^{(1)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ d_y(v^{(m)} - v_o)X_w^{(m)} & d_y(v^{(m)} - v_o)Y_w^{(m)} & d_y(v^{(m)} - v_o)Z_w^{(m)} & d_y(v^{(m)} - v_o) & -\tilde{d}_x(u^{(m)} - u_o)X_w^{(m)} & -\tilde{d}_x(u^{(m)} - u_o)Y_w^{(m)} & -\tilde{d}_x(u^{(m)} - u_o)Z_w^{(m)} \end{bmatrix}_{m \times 7} \quad (2.58)$$

$$X = \begin{bmatrix} s_x r_1 T_y^{-1} \\ s_x r_2 T_y^{-1} \\ s_x r_3 T_y^{-1} \\ s_x T_x T_y^{-1} \\ r_4 T_y^{-1} \\ r_5 T_y^{-1} \\ r_6 T_y^{-1} \end{bmatrix}, \quad (2.59)$$

and

⁶ When $T_y = 0$ or close to zero, it causes singularity in (2.55). When this situation occurs, an artificial rotation of the camera frame, or an artificial translation of the world frame, can be introduced to make $T_y \neq 0$.

$$b = \begin{bmatrix} \tilde{d}_x(u^{(1)} - u_o) \\ \vdots \\ \tilde{d}_x(u^{(m)} - u_o) \end{bmatrix}_{m \times 1} \quad (2.60)$$

to obtain the linear system

$$AX = b. \quad (2.61)$$

Assuming $s_x = 1$ and $m > 7$, (2.61) is overdetermined and the least squares solution is

$$\hat{X} = [A^T A]^{-1} A^T b \quad (2.62)$$

that minimizes the cost function

$$E(X) = \frac{1}{2} [AX - b]^T [AX - b]. \quad (2.63)$$

Recall in (2.19), the definition for I_c^w is

$$I_c^w = \begin{bmatrix} r_1 & r_2 & r_3 \\ r_4 & r_5 & r_6 \\ r_7 & r_8 & r_9 \end{bmatrix}. \quad (2.64)$$

In Tsai's paper, he did not enforce an orthogonality constraint for the rotation matrix I_c^w .

Aitken modified Tsai's algorithm to add the following constrained minimization to orthogonalize the rotation matrix I_c^w . Because the third row of I_c^w is estimated based on the cross product of first and second row, to ensure the orthogonal I_c^w , we only need to satisfy

$$\begin{bmatrix} r_1 & r_2 & r_3 \end{bmatrix} \begin{bmatrix} r_4 \\ r_5 \\ r_6 \end{bmatrix} = 0. \quad (2.65)$$

Hence, using (2.59) to provide elements of the first and second rows of I_c^w , we can

rewrite the orthogonality constraint (2.65) as

$$X^T QX = 0 \quad (2.66)$$

where $Q = \begin{bmatrix} 0_{3 \times 4} & I_3 \\ 0_{4 \times 7} & \end{bmatrix}_{7 \times 7}$ and I_3 denotes the 3×3 identity matrix.

Now our original minimization problem becomes the following new constrained minimization:

$$\text{Minimize the cost function } E(X) = \frac{1}{2} [AX - b]^T [AX - b] \text{ subject to } X^T QX = 0. \quad (2.67)$$

The method of Lagrange multipliers [26] can be used to find the minimum for $E(X)$ in (2.67). One can convert the constrained minimization in (2.67) into one Lagrange function

$$E(X, \lambda) = E(X) + \frac{1}{2} \lambda X^T QX \quad (2.68)$$

which is the sum of the cost function $E(X)$ and the product of the Lagrange multiplier λ and the constraint $X^T QX = 0$. Solving for the constrained minimum of $E(X)$ in (2.67) is equivalent to finding the minimum for (2.68) and requires that

$$\left[\frac{\partial E(X, \lambda)}{\partial X} \right]^T = [A^T A + \lambda Q^T] X - A^T b = 0_{7 \times 1}, \text{ and} \quad (2.69)$$

$$\frac{\partial E(X, \lambda)}{\partial \lambda} = X^T QX = 0_{1 \times 1}.$$

Rewrite (2.69) into

$$F(X_s) = \begin{bmatrix} [A^T A + \lambda Q^T] X - A^T b \\ X^T QX \end{bmatrix} = 0_{8 \times 1} \quad (2.70)$$

where $X_s = [X^T, \lambda]^T$ and apply Newton's iterative method [19][26] to find a solution for (2.70).

The Newton iterations are

$$\widehat{X}_s^{(n+1)} = \widehat{X}_s^{(n)} - \alpha \left[J(\widehat{X}_s^{(n)}) \right]^{-1} F(\widehat{X}_s^{(n)}), \quad (2.71)$$

where $\alpha \in (0, 1]$ is the step size to prevent overshoot [26] and n is the number of iterations.

If we let $\lambda = 0$, equation (2.68) reduces to (2.63). Hence, we can use the least squares

solution in (2.62) as the initial value to start Newton's iterations. For each Newton's

iteration, we need to check the current solution for $\widehat{X}_s^{(n)}$ and verify that

$$\widehat{X}_s^{(n)} \rightarrow [A^T A + \lambda Q^T]^{-1} A^T b$$

and

$$(\widehat{X}_s^{(n)})^T Q \widehat{X}_s^{(n)} \rightarrow 0,$$

and continue to iterate until the estimation for $\widehat{X}_s^{(n)}$ can no longer be improved. When the

optimal solution for $\widehat{X}_s^{(n)}$ is finally obtained at the n th iteration, we have

$$\widehat{X}^{(n)} = \begin{bmatrix} X(1) \\ X(2) \\ X(3) \\ X(4) \\ X(5) \\ X(6) \\ X(7) \end{bmatrix} = \begin{bmatrix} s_x r_1 T_y^{-1} \\ s_x r_2 T_y^{-1} \\ s_x r_3 T_y^{-1} \\ s_x T_x T_y^{-1} \\ r_4 T_y^{-1} \\ r_5 T_y^{-1} \\ r_6 T_y^{-1} \end{bmatrix}. \quad (2.72)$$

From $X(5)$, $X(6)$, and $X(7)$ in (2.72), we obtain

$$\begin{aligned} & X(5)^2 + X(6)^2 + X(7)^2 \\ &= (r_4 T_y^{-1})^2 + (r_5 T_y^{-1})^2 + (r_6 T_y^{-1})^2 \\ &= \frac{(r_4^2 + r_5^2 + r_6^2)}{(T_y)^2} \end{aligned} \quad (2.73)$$

Using the orthogonality constraint for I_c^w , one can get $r_4^2 + r_5^2 + r_6^2 = 1$ and substitute the result into (2.73) to find the magnitude of T_y ,

$$|T_y| = \frac{1}{\sqrt{X(5)^2 + X(6)^2 + X(7)^2}} \quad (2.74)$$

Next the sign of T_y has to be decided. Select the scene point P_j that is the furthest away from the image center. Let the 3D coordinates for P_j in the world frame, F_w , be

$[X_j \ Y_j \ Z_j]_w$ and P_j in the camera frame, F_c , be $[X_j \ Y_j \ Z_j]_c$. From (2.19), we

know

$$\begin{aligned} [X_j]_c &= r_1[X_j]_w + r_2[Y_j]_w + r_3[Z_j]_w + T_x \\ [Y_j]_c &= r_4[X_j]_w + r_5[Y_j]_w + r_6[Z_j]_w + T_y \end{aligned} \quad (2.75)$$

Previously we assumed that $s_x = 1$, but to calculate the sign of T_y we only need the fact that s_x is positive; hence set $s_x = 1$ here to get

$$\begin{aligned} \psi_1 &= X(1)[X_j]_w + X(2)[Y_j]_w + X(3)[Z_j]_w + X(4) \\ &= \frac{s_x r_1[X_j]_w + s_x r_2[Y_j]_w + s_x r_3[Z_j]_w}{T_y} + \frac{s_x T_x}{T_y} \\ &= \frac{r_1[X_j]_w + r_2[Y_j]_w + r_3[Z_j]_w + T_x}{T_y} \\ &= \text{sign}(T_y) \frac{r_1[X_j]_w + r_2[Y_j]_w + r_3[Z_j]_w + T_x}{|T_y|} \\ &= \text{sign}(T_y) \frac{[X_j]_w}{|T_y|} \end{aligned} \quad (2.76)$$

and

$$\begin{aligned}
\psi_2 &= X(5) \left[X_j \right]_w + X(6) \left[Y_j \right]_w + X(7) \left[Z_j \right]_w + 1 \\
&= \frac{r_4 \left[X_j \right]_w + r_5 \left[Y_j \right]_w + r_6 \left[Z_j \right]_w}{T_y} + 1 \\
&= \text{sign}(T_y) \frac{\left[Y_j \right]_w}{\left| T_y \right|}
\end{aligned} \tag{2.77}$$

According to Tsai's parallelism constraint, (2.20), (2.21), (2.75), (2.76), and (2.77) the sign of T_y can be decided by assumssing the sign for T_y is positive and do the test below

$$\begin{aligned}
&\text{If } (\text{sign}(\psi_1) == \text{sign}(x_{d_j}) \text{ and } (\text{sign}(\psi_2) == \text{sign}(y_{d_j}))) \\
&\quad T_y \text{ is positive} \\
&\text{Else} \\
&\quad T_y \text{ is negative} \\
&\text{End of test.}
\end{aligned} \tag{2.78}$$

After knowing the sign and magnitude of T_y , one can use the following to calculate the remaining parameters. The scale factor, s_x , is given by

$$s_x = \left| T_y \right| \sqrt{[X(1)^2 + X(2)^2 + X(3)^2]} \tag{2.79}$$

Let

$$I_c^w = \begin{bmatrix} r_1 & r_2 & r_3 \\ r_4 & r_5 & r_6 \\ r_7 & r_8 & r_9 \end{bmatrix} = \begin{bmatrix} \text{row1} \\ \text{row2} \\ \text{row3} \end{bmatrix}, \tag{2.80}$$

Then the elements of I_c^w can be calculated using

$$\begin{aligned}
&\text{row1} \\
&= [r_1 \quad r_2 \quad r_3] \\
&= \left[X(1) \frac{T_y}{s_x} \quad X(2) \frac{T_y}{s_x} \quad X(3) \frac{T_y}{s_x} \right]
\end{aligned} \tag{2.81}$$

$$\begin{aligned}
& \text{row2} \\
& = [r_4 \quad r_5 \quad r_6] \\
& = [X(5)T_y \quad X(6)T_y \quad X(7)T_y]
\end{aligned} \tag{2.82}$$

$$\begin{aligned}
& \text{row3} \\
& = \frac{[r_1 \quad r_2 \quad r_3] \times \begin{bmatrix} r_4 \\ r_5 \\ r_6 \end{bmatrix}}{\left\| [r_1 \quad r_2 \quad r_3] \times \begin{bmatrix} r_4 \\ r_5 \\ r_6 \end{bmatrix} \right\|} .
\end{aligned} \tag{2.83}$$

The T_x component of the translation vector is then computed as

$$T_x = X(4) \frac{T_y}{s_x} . \tag{2.84}$$

Finally, at the end of step 1, I_c^w , T_x , T_y , and s_x are all recovered. Now we can proceed to

Tsai's next step.

Step 2

In the second step, we estimate the T_z component of the translation vector, the focal length f , and the radial distortion coefficients k_1 and k_2 . The key for step 2 is to minimize the squares of errors between the predicted and undistorted image points on the image plane. The parameters T_z , f , k_1 , and k_2 are optimized to minimize the cost function

$$\sum_{j=1}^m (x_u^{(j)} - \widetilde{x_u^{(j)}})^2 + \sum_{j=1}^m (y_u^{(j)} - \widetilde{y_u^{(j)}})^2 , \tag{2.85}$$

where j = the number of scene points used in the calibration.

According to (2.19), (2.20), and based on the known scene point coordinates

$[X_w \ Y_w \ Z_w]_w$ in the world frame, the predicted undistorted image points (x_u, y_u) are

$$\begin{aligned} x_u &= f \frac{r_1 X_w + r_2 Y_w + r_3 Z_w + T_x}{r_7 X_w + r_8 Y_w + r_9 Z_w + T_z} \\ y_u &= f \frac{r_4 X_w + r_5 Y_w + r_6 Z_w + T_y}{r_7 X_w + r_8 Y_w + r_9 Z_w + T_z} \end{aligned} \quad (2.86)$$

and the undistorted image points calculated from the observed image points $(\tilde{x}_d, \tilde{y}_d)$

according to the radial distortion model are

$$\begin{aligned} \tilde{x}_u &= \tilde{x}_d + \tilde{x}_d(k_1 r^2 + k_2 r^4) \\ \tilde{y}_u &= \tilde{y}_d + \tilde{y}_d(k_1 r^2 + k_2 r^4) \end{aligned} \quad (2.87)$$

where $r = \sqrt{(\tilde{x}_d)^2 + (\tilde{y}_d)^2}$.

Hence the image errors between the predicted and computed undistorted image points are

$$\begin{aligned} F_x(\theta) &= x_u - \tilde{x}_u = f \frac{r_1 X_w + r_2 Y_w + r_3 Z_w + T_x}{r_7 X_w + r_8 Y_w + r_9 Z_w + T_z} - [\tilde{x}_d + \tilde{x}_d(k_1 r^2 + k_2 r^4)], \\ F_y(\theta) &= y_u - \tilde{y}_u = f \frac{r_4 X_w + r_5 Y_w + r_6 Z_w + T_y}{r_7 X_w + r_8 Y_w + r_9 Z_w + T_z} - [\tilde{y}_d + \tilde{y}_d(k_1 r^2 + k_2 r^4)]. \end{aligned} \quad (2.88)$$

Here, θ is the parameter vector to be estimated, $\theta = [f \ T_z \ k_1 \ k_2]^T$.

Assuming m 3D scene and 2D image points are available, one can define

$$F(\theta) = [F^{(1)}(\theta) \ F^{(2)}(\theta) \ \dots \ F^{(m)}(\theta)]^T, \quad (2.89)$$

where

$$F^{(j)}(\theta) = [F_x^{(j)}(\theta) \ F_y^{(j)}(\theta)]^T, \quad (2.90)$$

and

$$\begin{aligned}
F_x^{(j)}(\theta) &= x_x^{(j)} - \widetilde{x_x^{(j)}} = f \frac{r_1 X_w^{(j)} + r_2 Y_w^{(j)} + r_3 Z_w^{(j)} + T_x}{r_7 X_w^{(j)} + r_8 Y_w^{(j)} + r_9 Z_w^{(j)} + T_z} - \left[\widetilde{x_d^{(j)}} + \widetilde{x_d^{(j)}} (k_1 (r^{(j)})^2 + k_2 (r^{(j)})^4) \right], \\
F_y^{(j)}(\theta) &= y_x^{(j)} - \widetilde{y_x^{(j)}} = f \frac{r_4 X_w^{(j)} + r_5 Y_w^{(j)} + r_6 Z_w^{(j)} + T_y}{r_7 X_w^{(j)} + r_8 Y_w^{(j)} + r_9 Z_w^{(j)} + T_z} - \left[\widetilde{y_d^{(j)}} + \widetilde{y_d^{(j)}} (k_1 (r^{(j)})^2 + k_2 (r^{(j)})^4) \right].
\end{aligned} \tag{2.91}$$

In the perfect case, the errors between the predicted and observed undistorted image points are zero and $F(\theta) = 0$. In practice we wish to estimate the parameters, θ , to minimize a quadratic cost function. Aitken suggested to apply the Gauss-Newton iteration technique to minimize equation (2.85). He started by finding the initial estimates for T_z , f , k_1 , and k_2 in order to allow Gauss-Newton iterations to start. These initial values are estimated by first converting the nonlinear system in (2.91) to a linear one and then the least squares method is employed to find an initial guess. Radial lens distortion coefficients k_1 and k_2 are scaled by large constants C_1 and C_2 respectively, and equation (2.91) becomes

$$\begin{aligned}
F_x^{(j)}(\theta) &= x_x^{(j)} - \widetilde{x_x^{(j)}} = f \frac{a_x^{(j)}}{b^{(j)} + T_z} - \left[\widetilde{x_d^{(j)}} + \widetilde{x_d^{(j)}} (C_1 \widetilde{k}_1 (r^{(j)})^2 + C_2 \widetilde{k}_2 (r^{(j)})^4) \right] \\
F_y^{(j)}(\theta) &= y_x^{(j)} - \widetilde{y_x^{(j)}} = f \frac{a_y^{(j)}}{b^{(j)} + T_z} - \left[\widetilde{y_d^{(j)}} + \widetilde{y_d^{(j)}} (C_1 \widetilde{k}_1 (r^{(j)})^2 + C_2 \widetilde{k}_2 (r^{(j)})^4) \right]
\end{aligned} \tag{2.92}$$

where

$$\begin{aligned}
k_1 &= C_1 \widetilde{k}_1, \\
k_2 &= C_2 \widetilde{k}_2, \\
a_x^{(j)} &= r_1 X_w^{(j)} + r_2 Y_w^{(j)} + r_3 Z_w^{(j)} + T_x, \\
a_y^{(j)} &= r_4 X_w^{(j)} + r_5 Y_w^{(j)} + r_6 Z_w^{(j)} + T_y, \text{ and} \\
b^{(j)} &= r_7 X_w^{(j)} + r_8 Y_w^{(j)} + r_9 Z_w^{(j)}
\end{aligned}$$

The constants C_1 and C_2 are defined fairly large (10^4 and 10^8 respectively) to make $\widetilde{k}_1$ and $\widetilde{k}_2$ small in order to improve the conditioning of matrix inversion in the Gauss Newton method. The initial estimates are computed assuming that $k_1 = 0$ and $k_2 = 0$. In this case equation (2.92) reduces to

$$\begin{aligned} F_x^{(j)}(\theta) &= f \frac{a_x^{(j)}}{b^{(j)} + T_z} - \widetilde{x}_d^{(j)} = 0 \\ F_y^{(j)}(\theta) &= f \frac{a_y^{(j)}}{b^{(j)} + T_z} - \widetilde{y}_d^{(j)} = 0 \end{aligned} \quad (2.93)$$

and these can be rewritten into the following matrix form,

$$AZ = B,$$

where

$$A = \begin{bmatrix} -\widetilde{x}_d^{(1)} & a_x^{(1)} \\ -\widetilde{y}_d^{(1)} & a_y^{(1)} \\ \vdots & \vdots \\ -\widetilde{x}_d^{(m)} & a_x^{(m)} \\ -\widetilde{y}_d^{(m)} & a_y^{(m)} \end{bmatrix}, \quad (2.94)$$

$$Z = \begin{bmatrix} T_z \\ f \end{bmatrix}^T, \quad (2.95)$$

and

$$B = \begin{bmatrix} b^{(1)} x_d^{(1)} \\ b^{(1)} y_d^{(1)} \\ \vdots \\ b^{(m)} x_d^{(m)} \\ b^{(m)} y_d^{(m)} \end{bmatrix}. \quad (2.96)$$

If m is the number of the observed image points in the frame grabber memory, then we have an overdetermined system with $2m$ equations when m is greater than 1. The least squares solution for Z is

$$\hat{Z} = [A^T A]^{-1} A^T B. \quad (2.97)$$

Now the initial guess for the Gauss-Newton technique is

$$\theta_0 = \begin{bmatrix} \hat{Z} \\ 0 \\ 0 \end{bmatrix}. \quad (2.98)$$

Rewrite the cost function in (2.85) into

$$E(\theta) = \frac{1}{2} F^T(\theta) F(\theta). \quad (2.99)$$

Now we can use Gauss-Newton's iteration technique to find an optimal $\hat{\theta}$ that minimizes $E(\theta)$. Details about Gauss-Newton's method can be found in [7][19][27][28]. The

Gauss-Newton iterations are

$$\hat{\theta}_{k+1} = \hat{\theta}_k - \alpha \left[\frac{\partial F^T(\hat{\theta}_k)}{\partial \theta} \frac{\partial F(\hat{\theta}_k)}{\partial \theta} \right]^{-1} \frac{\partial F^T(\hat{\theta}_k)}{\partial \theta} F(\hat{\theta}_k),$$

where

$\frac{\partial F(\theta)}{\partial \theta}$ is the Jacobian matrix, and the step size $\alpha \in (0,1]$.

The solution for T_z , f , k_1 , and k_2 are

$$T_z = \hat{\theta}(1), \quad (2.100)$$

$$f = \hat{\theta}(2), \quad (2.101)$$

$$k_1 = C_1 \hat{\theta}(3), \quad (2.102)$$

and

$$k_2 = C_2 \hat{\theta}(4). \quad (2.103)$$

Finally, we can combine parameters estimated in step 1, I_c^w , s_x , T_x , and T_y , with parameters, T_z , f , k_1 and k_2 estimated in step 2 to obtain the final solutions for the camera calibration.

2.10 Comparison of Linear DLT, Linear Modified DLT and Tsai

Camera calibration Methods

In this section, the DLT, LMDLT, and Tsai's camera calibration methods are summarized and compared. Table 2.2 provides a comparison overview of these camera calibration methods.

Line jitter effect and radial lens distortion: Tsai's method is the only method of the three that models the line jitter effect using an "uncertainty image scale factor s_x ".

The LMDLT method has the ability to include lens distortions by increasing the number of DLT parameters. However, the inclusion of lens distortions results in a nonlinear system requiring full scale nonlinear search and is out of the scope of this comparison study. The DLT and LMDLT methods implemented for this thesis only used 11 DLT parameters without considering any lens distortions.

Orthogonal constraint is the requirement that the rows of a rotation matrix are orthogonal to each other. The rotation matrix estimated by the classic DLT method does not give an orthogonal rotation matrix. The LMDLT improves DLT's lack of orthogonality by estimating the orthogonal rotation matrix using the method described in Section 2.9.2. Tsai's method includes an orthogonality constraint into its solution finding

process by solving a constrained minimization problem⁷. Both LMDLT and Tsai include the orthogonal constraint and hence parameters estimated are more accurate than the ones obtained by the classic DLT method.

Parameter estimation: The classic DLT method solves its overdetermined system using the linear least squares method. Its 11 DLT parameters are solved all at once. Extrinsic and intrinsic parameters are recovered from its estimated DLT parameters. The parameters for LMDLT method are estimated using a Newton-like iterative approach. It applies the classic DLT method to calculate the 11 DLT parameters as an initial solution and then enforces the orthogonality constraint for the rotation matrix I_c^w and finds the nonlinear relationship between the 11 DLT parameters to reduce the number of DLT parameters to 10. Then the system is linearized to apply the linear least squares method to find new estimates. The previously eliminated parameter is then calculated based on the 10 newly estimated parameters. The steps above are repeated until the set of converged DLT parameters are found. Finally, the estimated principal point, focal length, projection centre, rotation matrix and translation vector are calculated from these DLT parameters. Tsai uses a two-step approach to calculate the camera parameters. In step one, assuming that the parallelism constraint holds and that the camera only suffers from radial lens distortion, the orthogonality constraint is added to the parameter estimation process to produce the orthogonal rotation matrix. In Tsai's second step, he solves and optimizes the unrecovered parameters based on those found in the first step. Unlike the DLT method, Tsai's calibration method does not estimate the

⁷ Tsai indicated that row 3 of rotation matrix is orthogonal to row1 and 2 but without enforcing the orthogonality of row 1 and row 2. The implementation of Tsai method in this thesis is based on the improvement proposed by V. Aitken that ensures the orthogonal rotation matrix.

location of the image center. Tsai assumes that the principal point is located at the center of the camera's CCD array.

Table 2.2 Comparisons of DLT, LMDLT, and Tsai Camera Calibration Methods

Camera Calibration Methods	DLT	LMDLT	Tsai
Radial lens distortion	No, not modeled	Depends on number of parameters used	Yes, modeled
Line jitter effect	No, not modeled	No, not modeled	Yes, modeled
Orthogonality constraint for rotation matrix	No	Yes	Yes (V.Aitken's modified Tsai [19])
Method used to solve equations	Least squares method	Linearization and Least squares method	Two-step approach: Least squares method, Newton's method and Gauss-Newton's optimization
Minimum number of control points required to solve equations	6	6	7
Principal point	Yes	Yes	Assumed at the centre of CCD
Scaling parameters (from meters to pixels)	Yes	Yes	Assumed Available from manufacturers
Focal length	Yes	Yes	Yes
Extrinsic Parameters (position, orientation)	Yes	Yes	Yes

Chapter 3 Experiments and Discussion

As discussed in section 2.5, there are many sources that may cause errors in a camera calibration. Experiments were conducted with different calibration setups based on the possible error sources discussed in Section 2.5. The purpose of these experiments is to evaluate and compare the linear modified DLT and the nonlinear Tsai camera calibration methods. These experiments were based on a computer generated camera and calibration jig. This allows easy access to true camera parameters. Evaluations and comparisons were based on the differences between true and estimated camera parameters. These experiments are divided into three major experiment groups that study:

1. Effect of 3D scene points coordinates measurement errors,
2. Effect of radial distortions,
3. Effect of resolution noise,
4. Effect of digitization error,
5. Effect of combination of all the above error sources from 1 to 4, and
6. Effect of changing the number of Control Points with all the above error sources from 1 to 4.

Each experiment group was further divided into different test cases when necessary. In each group, both LMDLT and Tsai camera calibration methods were performed to provide results for study. When Monte Carlo Simulation is required, for each test, 1000 simulation runs were executed. The comparison between LMDLT and Tsai methods was done based on the intrinsic parameters (focal length f) and the extrinsic parameters (the rotation matrix and x-y-z components of the translation vector).

Because the rotation matrix is actually characterized by the roll-pitch-yaw angles, the estimated rotation matrix is converted into roll-pitch-yaw angles for comparison. The LMDLT calibration technique does not yield the horizontal scaling factor s_x , and the radial distortion coefficients k_1 and k_2 . But these parameters are important components of Tsai's technique; these estimated parameters are also listed with other parameters to help understand the performance of Tsai's algorithm.

In this Chapter, Section 3.1 explains how noise was generated when required for the experiments. Section 3.2 gives an overview of Monte Carlo Simulation. Section 3.3 explains the computer generated virtual calibration jig and camera. Sections 3.4 to 3.10 discuss various test cases with changing calibration environments.

3.1 Gaussian Noise Generation

The Gaussian distribution plays an important role in probability and statistics theory. The main reason for this is that the sum of independent observations tends to a Gaussian distribution very quickly [29]. Hence, Gaussian random numbers are chosen to simulate resolution noise or measurement errors. Figure 3.1 [30] illustrates the Gaussian distribution. In Figure 3.1, "s" represents the standard deviation.

Figure 3.1: Zero-mean Gaussian distribution marked with 68%, 95%, and 99.7% Empirical Rule [30]

The mean is $\bar{x}$ and $(\bar{x} - s, \bar{x} + s)$ is the range of one standard deviation to the left and right of the mean $\bar{x}$. It is well known that 68%, 95%, and 99.7% of the distribution is located within 1, 2, and 3 standard deviations of the mean, respectively. According to [10], outliers in the data can also change the outcome of a camera calibration. The impact of outliers depends on the calibration method used to minimize the error of an estimate. Some methods (e.g. least-squares method) are more sensitive to outliers than other methods. Initial simulation results also showed that when the noise statistics are too wide spread, it causes numerical instability for some of the simulation runs. To avoid the effect of outliers [10], the MatLab [31] function “randn” is called to generate raw Gaussian random noise samples that lie within 1 standard deviation of the distribution’s zero mean. These raw noise samples were converted to the desired noise samples within $(-noise_level, noise_level)$ and were added to the 3D scene point coordinates using the following equations:

$$\begin{aligned} X_{noise} &= X + (noise_level) * randn, \\ Y_{noise} &= Y + (noise_level) * randn, \\ Z_{noise} &= Z + (noise_level) * randn, \end{aligned} \tag{3.1}$$

where (X, Y, Z) are the true 3D world point coordinates, and $noise_level$ is the desired standard deviation (e.g. 2mm or 5mm), and $randn$ is the MatLab Gaussian random number generator.

3.2 Monte Carlo Simulation

Monte Carlo Simulation involves repeated execution of an experiment by repeatedly sampling values from the probability distributions for the uncertain variables during each run. At the end of the simulation, the statistics for a few hundreds or

thousands of simulation runs are collected. Figure 3.2 shows an overview of the Monte Carlo Simulation for the effect of jig errors on Tsai's camera parameters. In this example, 3D scene point coordinate measurement noise is generated using the scheme described in Section 3.1 and added to the "perfect scene points coordinates" to emulate errors caused by an imperfectly manufactured calibration grid. These "noisy scene points" are then captured by MatCam to find their associated "noisy 2D distorted image points". These 2D distorted image points along with perfect scene points coordinates are passed to LMDLT or Tsai's calibration algorithm described in Section 2.9 to find "noise polluted camera parameters". This process is repeated over and over again to understand the variability of the camera parameters.

Figure 3.2: Overview of one test case that applied Monte Carlo Simulation

3.3 Virtual Camera and Calibration Jig

As mentioned in Chapter 1, the purpose of this thesis is to examine and compare how parameters of DLT and Tsai camera calibration methods interact with the calibration environment. Because the results of this study are to be used by the autonomous robotic vehicle, TART, the virtual calibration environment was generated similar to Figure 1.2. MatCam [7] was used to generate the virtual camera and calibration jig. MatCam allows the operator to input: 1. the desired intrinsic and extrinsic parameters for the virtual camera; and 2. the desired location and specification of the virtual calibration jig. The original code for the MatCam was modified to make it suitable for Monte Carlo Simulation and to enable it to interface with DLT, LMDLT, and Tsai's camera calibration algorithms for the comparison study.

Figure 3.3: Computer generated camera and calibration jig in world reference frame

Figure 3.3 shows an illustration of the virtual calibration jig and camera generated by MatCam in the world reference frame. The calibration jig is placed at three different z-stages to provide a scene volume for a non-coplanar camera calibration. The jig seen in Figure 3.3 is located at the first z-stage, the ground level. The x-y plane of the world reference frame is assumed to be parallel to and overlap with the upper surface of the calibration jig. As shown in Figure 3.3, this setting was used to simulate a camera mounted on a vehicle looking to the front at a scene that occupies 2 by 2 square meters. As shown in Figure 3.3, the camera was assumed to be located at world coordinates (3.5, 2.5, 4.0) m and the jig center is located at (4.0, 4.0, 0.0) m. This gives a distance of 4.30 meters between the calibration jig centre and the camera. The horizontal distance between the camera and jig centre is 1.58 meters. The vehicle is assumed to travel at a maximum speed of 2.5 km/hr. Therefore, 1.58 meters gives around 2.27 seconds of response time after seeing the object and before colliding with it.

According to Tsai's paper, in machine vision only two terms of radial distortion have the most significant effect on calibration results. Therefore, the virtual camera used in the study had first and second order radial distortions. Other lens distortions were not included. Digitization error and resolution errors are very common in off-the-shelf CCD cameras. They are included in the software camera as well. A high quality calibration jig is difficult to build and its cost is too high. Therefore, it is worthwhile to investigate the effect of a low cost calibration jig on calibration results. The virtual calibration jig simulates a low cost wood calibration board as shown in Figure 3.4. This jig is made of wood and the control points are manually measured and labeled with paper stickers.

Using this jig, control point coordinate measurement errors are unavoidable and are considered in this study.

Both LMDLT and Tsai's methods used in the experiments require non-coplanar jig points inside the field of view of the camera. To satisfy this requirement, virtual jig points were generated at 3 different z-stages to provide these non-coplanar points for camera calibration. This setup simulates the low cost wood calibration jig in Figure 3.4 being placed at three different heights, 0, 0.15, and 0.35 meters respectively to allow the camera to capture images for the calibration.

Figure 3.4: Low cost wood calibration jig

3.4 DLT and Tsai Camera Calibration in a Perfect Calibration

Environment

The first experiment, Case #0, verifies the implementation of LMDLT and Tsai calibration software. These two experiments assumed that there are no noises, no errors, and no lens distortions presented in the calibration environment. The parameters of interest are the extrinsic parameters and intrinsic camera parameters. As shown in Table

3.1, these extrinsic parameters are the rotation matrix and translation vector and the intrinsic parameters are the focal length f , scale factor s_x , and the distortion coefficients k_1 and k_2 . The rotation matrix is the rotation from the world to the camera frame and was converted into roll-pitch-yaw angles for comparison. The translation vector, $[T_x \ T_y \ T_z]^T$, is the translation from the world to the camera frame described in the world frame.

The distortion coefficients k_1 and k_2 are not used because Case #0 is performed in an error-free and distortion-free calibration environment. The scale factor s_x is only yielded by Tsai's method and is used for comparison between different test cases for Tsai's method itself. Because no Gaussian distributed noise was present for this test case Monte Carlo Simulation was not needed. Hence, The Root Mean Squares (RMS) errors are simply the absolute value of errors because only one run is performed in Case #0 for LMDLT and Tsai's methods. As illustrated in Table 3.1, experimental results showed that both LMDLT and Tsai calibration software was working properly because all the errors between the true and estimated camera parameters are on the order of round off. It is interesting to note that LMDLT estimates f with a somewhat larger error than Tsai; however, for the z-component of the translation vector, LMDLT yields a better prediction than Tsai.

Table 3.1 Calibration results for LMDLT and Tsai in Case #0

Case #0 calibration software verification								
LMDLT Method					Tsai Method			
Parameters	units	True Parameters	Estimated Parameters	RMS Error	Parameters	True Parameters	Mean	RMS Error
T_x	m	3.5000	3.5000	4.4409e-15	T_x	3.5000	3.5000	2.3981e-014
T_y	m	2.5000	2.5000	4.8850e-015	T_y	2.5000	2.5000	1.3500e-013
T_z	m	4.0000	4.0000	0	T_z	4.0000	4.0000	1.5365e-013
roll	rad	0.0000	0.0000	1.1462e-015	Roll	0.0000	0.0000	4.8324e-014
pitch	rad	1.1944	1.1944	2.2204e-16	Pitch	1.1944	1.1944	1.2212e-014
Yaw	rad	1.2490	1.2490	1.3323e-015	Yaw	1.2490	1.2490	5.0182e-014
f	m	0.0120	0.0120	1.8705e-005	f	0.0120	0.0120	6.9042e-016
s_x	none	N/A	N/A	N/A	s_x	1.0000	1.0000	2.0650e-014
k_1	m^{-2}	No lens distortion	No lens distortion	No lens distortion	k_1	0.0000	7.4932e-11	7.4932e-11
k_2	m^{-4}	No lens distortion	No lens distortion	No lens distortion	k_2	0.0000	2.8541e-006	2.8541e-006

3.5 Effect of 3D Scene Point Measurement Errors

To calibrate a camera requires accurate 3D scene point coordinates and their associated camera captured image point coordinates. Without these accurate 3D and 2D coordinates, camera calibration methods may produce undesired results. Nevertheless, finding exact 3D scene point coordinates is not an easy task because there are usually human measurement errors involved. For example, manufacturing an ideal calibration board is very expensive and difficult. Obtaining accurate scene points coordinates is also very challenging. There are devices such as theodolite systems to help reduce measurement errors; however, these devices are very expensive. Coordinate errors caused by human measurement errors and a low quality calibration board are unavoidable, and hence it is important to investigate their effects on camera calibration. The purpose

of experiments in this Section is to observe the effect of 3D coordinate errors on camera parameters and to compare calibration results from the LMDLT and Tsai methods.

3.5.1 Human Measurement Error

Figure 3.5: Simulation overview for human measurement errors test cases

Sometimes, because of a limited budget or the setup of a camera calibration environment, it is not possible to use a coordinate measuring machine to measure locations of 3D scene points. Under these circumstances, it is required to hand measure scene point coordinates. Experiments in this section compare the variability of LMDLT and Tsai camera parameters with changing scene coordinates noise levels resulting from human measurement errors.

Table 3.2 summarizes the true experimental setup for these test cases #1 and #2. As shown in the Table, the experimental set up for Cases #1 and #2 excludes digitization error, resolution noises, and radial lens distortions. Without other sources of errors and focus on human measurement error only, this setup allows one to clearly observe how LMDLT and Tsai's camera parameters interact with human measurement error and to provide comparisons of its effect. For each test case, 1000 Monte Carlo Simulation runs were conducted to simulate human measurement error, and statistics of the camera calibration parameters were collected from these simulation runs. The dimension of the virtual calibration jig used in Cases #1 and #2 was 2 meters by 2 meters. There were 25 scene points on the jig board and the point spacing is 0.5 meters. The virtual jig was moved in the vertical direction in the world reference frame and placed at three different heights, 0, 0.15, 0.35 meters, respectively. Other details of this calibration environment setup can also be found in Table 3.2. For Cases #1 and #2 it was assumed that there is only one calibration board and a camera calibration operator has to measure control points on the board by hand relative to the world frame. Therefore, the scene point coordinates obtained this way were polluted with Gaussian random errors. The differences between these two test cases are their scene coordinate noise level; the noise level for Cases #1 is 2 mm and for #2 is 5mm. As shown in Figure 3.5, the experiment procedure is as follows. For each Monte Carlo Simulation run, Gaussian noises were added to true 3D jig coordinates and these noisy coordinates were passed to the camera calibration algorithms to be used as the 3D scene points. However, the true 3D scene coordinates were used by MatCam to generate the 2D image coordinates. In this case the virtual camera produces a true image of the actual scene but the noisy measured 3D

points are used in the calibration algorithms. During the test, the following assumptions were made:

1. the camera was fixed and looked at the jig centre;
2. the jig was moved in the z-direction of the world reference frame and placed at three different z-stages;
3. during the test, the same calibration jig was used; and
4. for each test case, the things that changed between each run are the hand measured 3D scene coordinates.

Tables 3.3 and 3.4 summarize the results from simulation Case #1 and #2. The units for components of translation vector are meters, and units for roll-pitch-yaw angles are radians. These results revealed that LMDLT's and Tsai's translation and rotation matrices are both influenced by scene coordinate noise. At the same noise level, some parameters estimated by Tsai are not as accurate as LMDLT method. This may be because LMDLT method is based on a linear model without distortion and Tsai's method is nonlinear with radial distortion. In these two test cases the virtual camera does not include any lens distortion. However, Tsai's method still assumed radial lens distortion existed and tried to estimate radial lens distortion coefficient k_1 and k_2 . In reality, off-the-shelf cameras have radial lens distortions. When radial lens distortion is present, Tsai's method can still compute camera parameters with high accuracy because it is specifically designed to deal with radial lens distortion. This can be verified by results from Section 3.6.

Table 3.2 Summary of experiment setup for test Cases #1 and #2

Test Cases	Case #1		Case #2	
Scene point coordinates Noise Levels	2mm	2mm	5mm	5mm
Camera calibration Method performed	LMDLT	Tsai	LMDLT	Tsai
Number of Calibration Points	Total 75 points (25 points per z stage)		Total 75 points (25 points per z stage)	
Jig Size, Point spacing	2mx2m, point spacing 0.5 meters		2mx2m, point spacing 0.5 meters	
Height of Z stage0,Zstage1,Z stage3	0m, 0.15m, 0.35m		0m, 0.15m, 0.35m	
Origin of World Reference Frame, O_w	$O_w(0,0,0)$ w.r.t. world frame, in meters		$O_w(0,0,0)$ w.r.t. world frame, in meters	
Origin of Camera Reference Frame, O_c	$O_c(3.5,2.5,4.0)$ w.r.t. world frame, in meters		$O_c(3.5,2.5,4.0)$ w.r.t. world frame, in meters	
Centre of Calibration jig	$O_j(4,4,0)$ w.r.t. world frame, in meters		$O_j(4,4,0)$ w.r.t. world frame, in meters	
Focal length f	0.012m		0.012m	
Horizontal Uncertainty Image Scale Factor S_x	N/A	1	N/A	1
1 st order Radial Distortion Coefficient k_1	N/A		N/A	
2nd order Radial Distortion Coefficient k_2	N/A		N/A	
Frame grabber Resolution	640x480 pixels by pixels		640x480 pixels by pixels	
CCD Array Resolution	640x480 pixels by pixels		640x480 pixels by pixels	
CCD Array Size	1/3 inch square		1/3 inch square	
Assumptions Made	1. Camera and jig are fixed during the whole calibration process 2. No rotation from world reference frame Fw to jig reference frame Fj 3. zero roll between Fw and camera reference frame Fc 4. Camera sits at origin of Fc and points directly at jig centre			

Table 3.3 Simulation results for Case #1

Case #1 Gaussian Noise Level 2mm, 1000 Monte Carlo Simulation runs, 75 control points, no lens distortion								
LMDLT Method					Tsai Method			
Parameters	units	True Parameters	Mean	RMS Error	Parameters	True Parameters	Mean	RMS Error
T_x	m	3.5000	3.5000	0.0031	T_x	3.5000	3.4999	0.0067
T_y	m	2.5000	2.5001	0.0066	T_y	2.5000	2.5001	0.0094
T_z	m	4.0000	4.0002	0.0149	T_z	4.0000	4.0000	0.0122
roll	rad	0.0000	-6.6104e-004	6.6104e-004	roll	0.0000	0.0066	0.0066
pitch	rad	1.1944	1.1940	3.9239e-004	pitch	1.1944	1.1945	1.8593e-004
Yaw	rad	1.2490	1.2483	7.1361e-004	yaw	1.2490	1.2556	0.0065
f	m	0.0120	0.0120	4.9955e-005	f	0.0120	0.0120	4.8564e-005
s_x	none	N/A	N/A	N/A	s_x	1.0000	1.0000	6.5992e-004
k_1	m^{-2}	No lens distortion	No lens distortion	No lens distortion	k_1	0.0000	-6.5419	133.7030
k_2	m^{-4}	No lens distortion	No lens distortion	No lens distortion	k_2	0.0000	1.7723e+005	6.0846e+006

Table 3.4 Simulation results for Case #2

Case #2 Gaussian Noise Level 5mm, 1000 Monte Carlo Simulation runs, 75 control points, no lens distortion								
LMDLT Method					Tsai Method			
parameters	units	True Parameters	Mean	RMS Error	Parameters	True Parameters	Mean	RMS Error
T_x	m	3.5000	3.4997	0.0077	T_x	3.5000	3.5009	0.0164
T_y	m	2.5000	2.4990	0.0162	T_y	2.5000	2.5011	0.0227
T_z	m	4.0000	4.0036	0.0378	T_z	4.0000	3.9995	0.0303
Roll	rad	0.0000	0.0034	0.0034	roll	0.0000	0.0066	0.0066
Pitch	rad	1.1944	1.1962	0.0018	pitch	1.1944	1.1966	0.0022
Yaw	rad	1.2490	1.2531	0.0040	yaw	1.2490	1.2555	0.0065
f	m	0.0120	0.0120	1.1768e-004	f	0.0120	0.0120	1.2000e-004
s_x	none	N/A	N/A	N/A	s_x	1.0000	1.0001	0.0016
k_1	m^{-2}	No lens distortion	No lens distortion	No lens distortion	k_1	0.0000	-93.5588	1.1483e+003
k_2	m^{-4}	No lens distortion	No lens distortion	No lens distortion	k_2	0.0000	3.8316e+006	1.7086e+007

3.5.2 Effect of Calibration Jig Errors

As mentioned previously, manufacturing an accurate calibration jig according to predefined specifications is not easy and is expensive. Due to a limited budget or hardware limitations such as machining, assembly,...etc, the calibration jig quality may be compromised and calibration scene points are therefore dislocated from their theoretical positions. Simulations presented in this Section investigate the outcomes of such jig manufacturing error. Figure 3.6 is an overview of this experiment.

Figure 3.6: Experiment overview for jig errors

The experiment is very similar to Cases #1 and Case #2 stated in Table 3.1 and the two test cases here were called Case #3 and Case #4. The only difference for Cases #3 and Case #4 is that Gaussian noises were added to real world jig coordinates and these

noisy coordinates were captured by the camera. This allows the experiment to simulate the situation that a different calibration jig was used for each camera calibration. As shown in Figure 3.6, for each Monte Carlo simulation run, two groups of coordinates, the true scene coordinates and noise contaminated image coordinates were passed to LMDLT or Tsai camera calibration algorithm. For Case #3, maximum Gaussian jig manufacturing errors were chosen to be 2mm and for Case #4 they were chosen to be 5mm. A total of 1000 simulation runs were conducted for test Cases #3 and #4. Comparison of simulation results is shown in Table 3.5 and 3.6. The experimental results are very close to the ones obtained in Tables 3.3 and 3.4. These test results suggest that human measurement errors and calibration jig manufacturing errors affect camera parameter estimation similarly.

Table 3.5 Simulation results for LMDLT and Tsai methods when jig errors at 2mm noise level

Case #3 Gaussian Noise Level 2mm, 1000 Monte Carlo Simulation runs, 75 control points, no lens distortion								
LMDLT Method					Tsai Method			
Parameters	units	True Parameters	Mean	RMS Error	Parameters	True Parameters	Mean	RMS Error
T_x	m	3.5000	3.5000	0.0031	T_x	3.5000	3.5001	0.0067
T_y	m	2.5000	2.4999	0.0065	T_y	2.5000	2.5001	0.0094
T_z	m	4.0000	4.0001	0.0148	T_z	4.0000	3.9999	0.0122
roll	rad	0.0000	6.3665e-004	6.3665e-004	Roll	0.0000	-0.0066	0.0066
pitch	rad	1.1944	1.1947	3.6527e-004	Pitch	1.1944	1.1943	1.0714e-004
Yaw	rad	1.2490	1.2497	6.8715e-004	Yaw	1.2490	1.2425	0.0065
f	m	0.0120	0.0120	5.9469e-005	f	0.0120	0.0120	4.8608e-005
s_x	none	N/A	N/A	N/A	s_x	1.0000	1.0000	6.6012e-004
k_1	m^{-2}	No lens distortion	No lens distortion	No lens distortion	k_1	0.0000	-18.8737	134.8754
k_2	m^{-4}	No lens distortion	No lens distortion	No lens distortion	k_2	0.0000	7.9435e005	6.1317e006

Table 3.6 Simulation results for LMDLT and Tsai methods when jig errors at 5mm noise level

Case #4 Gaussian Noise Level 5mm, 1000 Monte Carlo Simulation runs, 75 control points, no lens distortion								
LMDLT Method					Tsai Method			
Parameters	units	True Parameters	Mean	RMS Error	Parameters	True Parameters	Mean	RMS Error
T_x	m	3.5000	3.5000	0.0077	T_x	3.5000	3.4995	0.0164
T_y	m	2.5000	2.4997	0.0164	T_y	2.5000	2.5001	0.0228
T_z	m	4.0000	4.004	0.0371	T_z	4.0000	4.0001	0.0304
roll	rad	0.0000	0.0016	0.0016	Roll	0.0000	-0.0066	0.0066
pitch	rad	1.1944	1.1953	9.0676e-004	Pitch	1.1944	1.1922	0.0021
Yaw	rad	1.2490	1.2507	0.0017	Yaw	1.2490	1.2427	0.0064
f	m	0.0120	0.0120	1.2147e-004	f	0.0120	0.0120	1.2032e-004
s_x	none	N/A	N/A	N/A	s_x	1.00000	1.0000	0.0016
k_1	m^{-2}	No lens distortion	No lens distortion	No lens distortion	k_1	0.0000	-65.8585	356.6018
k_2	m^{-4}	No lens distortion	No lens distortion	No lens distortion	k_2	0.0000	2.2617e006	1.6102e007

3.6 Effect of Radial Distortion on Camera Parameters

Experiments in this section investigate the calibration outcomes for LMDLT and Tsai under influence of radial lens distortion. Digitization error, resolution noise, and 3D scene point coordinates errors were removed from the experiments to allow a clear observation of interactions between radial distortion and parameters of LMDLT and Tsai. Table 3.7 summarizes the test setup. Two test cases, Cases #5 and #6, with different radial distortions were carried out. For Case #5, first order radial distortion coefficient $k_1 = 1000$ was used by MatCam to generate radial lens distortion and Case #6 first and second order radial distortion coefficients $k_1 = 1000, k_2 = 1 \times 10^7$ were used.

Table 3.7 Experiment setup for effect of radial distortion test cases: Cases #5 and Case #6

Test Cases	Case #5		Case #6	
Scene point coordinates Noise Levels	N/A	N/A	N/A	N/A
Camera calibration Method performed	LMDLT	Tsai	LMDLT	Tsai
Number of Calibration Points	Total 75 points (25 points per z stage)		Total 75 points (25 points per z stage)	
Jig Size, Point spacing	2mx2m, point spacing 0.5 meters		2mx2m, point spacing 0.5 meters	
Height of Z stage0,Zstage1,Z stage3	0m, 0.15m, 0.35m		0m, 0.15m, 0.35m	
Origin of World Reference Frame, O_w	$O_w(0,0,0)$ w.r.t. world frame, in meters		$O_w(0,0,0)$ w.r.t. world frame, in meters	
Origin of Camera Reference Frame, O_c	$O_c(3.5,2.5,4.0)$ w.r.t. world frame, in meters		$O_c(3.5,2.5,4.0)$ w.r.t. world frame, in meters	
Centre of Calibration jig	$O_j(4,4,0)$ w.r.t. world frame, in meters		$O_j(4,4,0)$ w.r.t. world frame, in meters	
Focal length f	0.012m		0.012m	
Horizontal Uncertainty Image Scale Factor S_x	N/A	1	N/A	1
1 st order Radial Distortion Coefficient k_1	$k_1 = 1000$		$k_1 = 1000$	
2nd order Radial Distortion Coefficient k_2	$k_2 = 0$		$k_2 = 1 \times 10^7$	
Frame grabber Resolution	640x480 pixels by pixels		640x480 pixels by pixels	
CCD Array Resolution	640x480 pixels by pixels		640x480 pixels by pixels	
CCD Array Size	1/3 inch square		1/3 inch square	
Assumptions Made	<div>1. Camera and jig are fixed during the whole calibration process</div> <div>2. No rotation from world reference frame Fw to jig reference frame Fj</div> <div>3. zero roll between Fw and camera reference frame Fc</div> <div>4. Camera sits at origin of Fc and points directly at jig centre</div>			

According to Table 3.8 and 3.9, the test results for both cases indicated that when radial lens distortion is present, Tsai's method estimated camera parameters much more accurately than LMDLT. Another observation from Table 3.8 and 3.9 is that when radial distortion increased, Tsai's algorithm can still estimate camera parameters accurately. Comparing the RMS errors in Table 3.5, and 3.6 for the DLT method, it can be seen that DLT's camera parameter estimation becomes less and less reliable when the magnitude of radial lens distortion is increased. However, pitch and focal length predicted by DLT are less sensitive to radial distortion than other parameters yielded by DLT.

Table 3.8 LMDLT and Tsai methods test results comparison for Case #5

Case #5 Radial lens distortion k1=1000, 75 control points, 2 simulation runs								
LMDLT Method					Tsai Method			
Parameters	units	True Parameters	Mean	RMS Error	Parameters	True Parameters	Mean	RMS Error
T_x	m	3.5000	3.4947	0.0053	T_x	3.5000	3.5000	7.8160e-014
T_y	m	2.5000	2.4793	0.0207	T_y	2.5000	2.5000	2.3626e-013
T_z	m	4.0000	4.0717	0.0717	T_z	4.0000	4.0000	5.9242e-013
roll	rad	0.0000	-0.0025	0.0025	Roll	0.0000	-9.4176e-015	9.4176e-015
pitch	rad	1.1944	1.1937	6.1664e-004	Pitch	1.1944	1.1944	4.6629e-015
Yaw	rad	1.2490	1.2463	0.0028	Yaw	1.2490	1.2490	5.1070e-015
f	m	0.0120	0.0121	6.8127e-005	f	0.0120	0.0120	9.3987e-015
s_x	none	N/A	N/A	N/A	s_x	1.0000	1.0000	3.9968e-015
k_1	m^{-2}	N/A	N/A	N/A	k_1	1000	1000	7.6492e-008
k_2	m^{-4}	N/A	N/A	N/A	k_2	N/A	N/A	N/A

Table 3.9 LMDLT and Tsai methods test results comparison for Case #6

Case #6 Radial lens distortion k1=1000 and k2=1e07, 75 control points, 2 simulation runs								
LMDLT Method					Tsai Method			
Parameters	units	True Parameters	Mean	RMS Error	Parameters	True Parameters	Mean	RMS Error
T_x	m	3.5000	3.4936	0.0064	T_x	3.5000	3.5000	9.9032e-014
T_y	m	2.5000	2.4744	0.0256	T_y	2.5000	2.5000	2.8422e-013
T_z	m	4.0000	4.0879	0.0879	T_z	4.0000	4.0000	6.2705e-013
roll	rad	0.0000	-0.0033	0.0033	Roll	0.0000	-6.3213e-015	6.3213e-015
pitch	rad	1.1944	1.1937	6.2890e-004	Pitch	1.1944	1.1944	1.2879e-014
Yaw	rad	1.2490	1.2455	0.0036	Yaw	1.2490	1.2490	5.7732e-015
f	m	0.0120	0.0121	1.0134e-004	f	0.0120	0.0120	9.0813e-015
s_x	none	N/A	N/A	N/A	s_x	1.0000	1.0000	3.4417e-015
k_1	m^{-2}	N/A	N/A	N/A	k_1	1000	1000	7.4310e-008
k_2	m^{-4}	N/A	N/A	N/A	k_2	1e07	1e07	0.0031

3.7 Effect of Resolution Noise on Camera Parameters

As described previously in Section 2.5, resolution noise comes from manually selecting the center of a circle by the camera calibration operator. Usually, Gaussian error is introduced to the 2D image coordinates of the manually selected points. The accuracy of this centre-selection task mainly depends on the CCD array resolution, the frame grabber resolution, the computer screen resolution, and the carefulness of the operator. In this experiment, the computer screen was assumed to have 14 inches diagonal length with 640×480 pixel-by-pixel screen resolution. To simplify our calculation, the monitor is also assumed to have a ratio, 3:4 (height : width). It is equivalent to around 8.4:11.2 inches. The resolution error was limited to below 2 pixels in the positive and negative directions. This results in an error zone of 4×4 pixels by pixels. According to these assumptions, 4×4 pixels by pixels is equivalent to an area of 0.07×0.07 inch squared. Experience showed that in most cases, when a calibration operator manually selects the centre of a circle, his mistake does not exceed this 4×4 error zone. Therefore, as shown in Table 3.10, the Gaussian noise level for this test case was chosen to be 2 pixels.

The purpose of this test case is to show how Gaussian resolution noise in the on 2D image coordinates affect the estimated parameters of LMDLT and Tsai camera models. Because the resolution error is Gaussian distributed, 1000 Monte Carlo Simulation runs are performed for LMDLT and Tsai methods to collect the statistics of the camera parameters. This test case only investigates the influence of resolution noise on the camera calibration. Other sources of errors as outlined in Section 2.5 are not included. The estimated camera parameters were compared with true camera parameters

to find parameter error statistics. As listed in Table 3.11, the results indicate that Tsai's method produced better roll-pitch-yaw estimation than the LMDLT method because their mean values are closer to the true roll-pitch-yaw and their RMS errors are also smaller than LMDLT's. LMDLT predicted the focal length slightly more accurately than Tsai. Comparing with Tsai, the translation vector estimated by LMDLT is also closer to the true translation vector.

Table 3.10 Experiment setup for Case #7

Test Cases	Case #7	
Camera calibration Method performed	LMDLT	Tsai
Scene Point Coordinates Noise Levels	0 mm	0mm
Resolution Noise Levels	2 pixels	2 pixels
Number of Calibration Points	Total 75 points (25 points per z stage)	
Jig Size, Point spacing	2mx2m, point spacing 0.5 meters	
Height of Z stage0,Zstage1,Z stage3	0m, 0.15m, 0.35m	
Origin of World Reference Frame, O_w	$O_w(0, 0, 0)$ w.r.t. world frame, in meters	
Origin of Camera Reference Frame, O_c	$O_c(3.5, 2.5, 4.0)$ w.r.t. world frame, in meters	
Centre of Calibration jig	$O_j(4, 4, 0)$ w.r.t. world frame, in meters	
Focal length f	0.012m	
Horizontal Uncertainty Image Scale Factor S_x	N/A	1
1 st order Radial Distortion Coefficient k_1	N/A	
2nd order Radial Distortion Coefficient k_2	N/A	
Frame grabber Resolution	640x480 pixels by pixels	
CCD Array Resolution	640x480 pixels by pixels	
CCD Array Size	1/3 inch square	
Assumptions Made	<ol style="list-style-type: none"> 1. Camera and jig are fixed during the whole calibration process. 2. No rotation from world reference frame Fw to jig reference frame Fj.. 3. Zero roll between Fw and camera reference frame Fc. 4. Camera sits at origin of Fc and points directly at jig centre 	

Table 3.11 Experiment results for Case #7

Case #7 Effect of Gaussian resolution noise @ 2 pixels level, 1000 Monte Carlo Simulation runs, 75 control points, no lens distortion								
LMDLT Method					Tsai Method			
Parameters	units	True Parameters	Mean	RMS Error	Parameters	True Parameters	Mean	RMS Error
T_x	m	3.5000	3.5057	0.0152	T_x	3.5000	3.5321	0.0478
T_y	m	2.5000	2.5176	0.0338	T_y	2.5000	2.4479	0.0752
T_z	m	4.0000	3.9917	0.0673	T_z	4.0000	4.0775	0.1027
roll	rad	0.0000	-0.0131	0.0131	roll	0.0000	3.4964e-004	3.4964e-004
pitch	rad	1.1944	1.1765	0.0179	pitch	1.1944	1.1870	0.0074
Yaw	rad	1.2490	1.2364	0.0127	yaw	1.2490	1.2498	7.4630e-004
f	m	0.0120	0.0119	2.2669e-004	f	0.0120	0.0123	3.9860e-004
s_x	none	N/A	N/A	N/A	s_x	1.00000	0.9982	0.0040
k_1	m^{-2}	No lens distortion	No lens distortion	No lens distortion	k_1	0.0000	389.7188	777.1198
k_2	m^{-4}	No lens distortion	No lens distortion	No lens distortion	k_2	0.0000	-1.8470e007	3.5435e007

3.8 Effect of Digitization Error on Camera Parameters

When an image of a scene object is captured by the camera CCD array, the image is digitized and stored. During the digitization process, the image is divided into pixels. When an image point is digitized, the point can only be located at the centre of a pixel. This may cause an image point to be dislocated from its ideal position, and error of this kind is called digitization error. In this test case, we examine the effect of digitization error on the camera calibration. The experimental setup for Case #8 is generally the same as the one listed in Table 3.10 for Case #7. The differences between test cases, #7 and #8, are that the resolution noise levels are changed to zeros for both LMDLT and Tsai in Case #8. In addition, in order to focus on the study of digitization error, the MatCam code is set to include only digitization error when it generates camera images for our calibration and exclude the other kinds of error sources.

The test results for Case #8 are listed in table 3.12. It can be observed from Table 3.12 that when the digitization error is presented in the calibration environment, the following parameters:

1. Tsai T_x and T_z components of the translation vector;
2. roll and yaw components of the rotation matrix; and
3. the focal length f ,

estimated by Tsai are more accurate than the ones predicted by LMDLT method. Although LMDLT method did well in estimating T_y and the pitch angle, it does not outperform Tsai much. For example, the difference for the mean value of T_y between LMDLT and Tsai's methods is 0.0362 meters. The difference 0.0362 meters is around 0.84% of the distance between the camera and the jig centre (around 4.3012 meters).

Table 3.12 Experiment results for Case #8

Case #8 Effect of digitization error, 100 runs, 75 control points, no lens distortion								
LMDLT Method					Tsai Method			
Parameters	units	True Parameters	Mean	RMS Error	Parameters	True Parameters	Mean	RMS Error
T_x	m	3.5000	3.5067	0.0067	T_x	3.5000	3.5056	0.0056
T_y	m	2.5000	2.5130	0.0130	T_y	2.5000	2.4768	0.0232
T_z	m	4.0000	3.9514	0.0486	T_z	4.0000	4.0152	0.0152
roll	rad	0.0000	0.0031	0.0031	Roll	0.0000	-0.0025	0.0025
pitch	rad	1.1944	1.1956	0.0012	Pitch	1.1944	1.1926	0.0018
Yaw	rad	1.2490	1.2520	0.0029	Yaw	1.2490	1.2463	0.0028
f	m	0.0120	0.0117	2.5942e-004	f	0.0120	0.0119	1.1782e-004
s_x	none	N/A	N/A	N/A	s_x	1.00000	0.9984	0.0016
k_1	m^{-2}	No lens distortion	No lens distortion	No lens distortion	k_1	0.0000	-1.5824e003	1.5824e003
k_2	m^{-4}	No lens distortion	No lens distortion	No lens distortion	k_2	0.0000	5.0939e007	5.0939e007

3.9 Effect of 3D Scene Point Coordinate Measurement Errors, Radial Lens Distortion, Resolution Noise, and Digitization Error on Camera Parameters

Each of the test cases from Section 3.5 to 3.8 focuses on one particular error source. Although these experimental setups allow us to observe the individual effect of each error source on the camera calibration results, these test cases do not provide experimental results that can give an overall effect on the calibration results when all the possible sources of errors appear in the calibration environment. In real life, off-the-shelf cameras usually have significant radial lens distortion. Other than radial lens distortions, digitization error, resolution noises and 3D scene point coordinate measurement errors are all common sources of errors that reduce the reliability of the camera calibration results. For example, a common physical camera calibration environment such as the one in Figure 1.1 usually contains radial lens distortion, digitization error, resolution noise and Gaussian coordinate noises. The purpose of test cases, #9 and #10, is to investigate the impact of noise and errors in the real calibration environment on LMDLT and Tsai's calibration methods.

Table 3.13 provides a summary of the test setup. As seen in the table, the radial lens distortion was specified by the distortion coefficients. Resolution noise was Gaussian distributed and set at 2 pixels standard deviation. The distribution of 3D scene point coordinate measurement error was also Gaussian distributed and set at 2mm standard deviation. 3D-scene-point-coordinate errors used in these test cases only result from human measuring errors. This is because simulation results in Section 3.5.1 and 3.5.2 showed that human measurement error and jig error produce identical outcomes on

camera parameter estimation. Hence, it is not necessary to study the effect of jig error again. Monte Carlo simulation was carried out and the statistics for the estimated camera parameters were gathered. The difference between Cases #9 and #10 is the fact that Case #9 has less radial lens distortion. The radial lens distortion was modeled using (2.22). The radial lens distortion coefficients (k_1, k_2) were set to $(k_1 = 1000, k_2 = 1 \times 10^7)$ for Case #9 and $(k_1 = 3000, k_2 = 1 \times 10^7)$ for Case #10.

The lists of experimental results are shown in Tables 3.14 and 3.15. Comparing Tsai's parameter statistics for Cases #9 and Case #10, we can see that when the radial lens distortion was made more significant, all the parameters estimated by Tsai were not very sensitive to this change; in addition, the LMDLT's parameter estimation becomes less reliable than Tsai's algorithm. LMDLT's parameters are generally more sensitive to radial lens distortion than Tsai's. For example, the mean and RMS errors for roll and yaw angles were increased around 10 times and the errors for translation vector components, T_x , T_y , and T_z , were about doubled when radial lens distortion increased from $(k_1 = 1000, k_2 = 10^7)$ to $(k_1 = 3000, k_2 = 10^7)$. From Tables 3.14 and 3.15, it is also indicated that the focal length f is not sensitive to noise, errors, and radial lens distortion because in both test cases, #9 and #10, the RMS errors for f remained very small. There is an interesting observation for T_x , roll, pitch, yaw, and the scale factor s_x . The estimation RMS errors for these parameters are even closer to their true values when the calibration environment contains more radial lens distortion. This may further imply that using Tsai's algorithm can minimize the errors caused by radial lens distortion successfully.

Table 3.13 Summary of Experimental setup for test cases #9 and #10

Test Cases	Case #9		Case #10	
Scene point coordinates Noise Levels	2mm	2mm	2mm	2mm
Camera calibration Method performed	LMDLT	Tsai	LMDLT	Tsai
Number of Calibration Points	Total 75 points (25 points per z stage)		Total 75 points (25 points per z stage)	
Jig Size, Point spacing	2mx2m, point spacing 0.5 meters		2mx2m, point spacing 0.5 meters	
Height of Z stage0,Zstage1,Z stage3	0m, 0.15m, 0.35m		0m, 0.15m, 0.35m	
Origin of World Reference Frame, O_w	$O_w(0,0,0)$ w.r.t. world frame, in meters		$O_w(0,0,0)$ w.r.t. world frame, in meters	
Origin of Camera Reference Frame, O_c	$O_c(3.5,2.5,4.0)$ w.r.t. world frame, in meters		$O_c(3.5,2.5,4.0)$ w.r.t. world frame, in meters	
Centre of Calibration jig	$O_j(4,4,0)$ w.r.t. world frame, in meters		$O_j(4,4,0)$ w.r.t. world frame, in meters	
Focal length f	0.012m		0.012m	
Horizontal Uncertainty Image Scale Factor s_x	N/A	1	N/A	1
1 st order Radial Distortion Coefficient k_1	1000		3000	
2nd order Radial Distortion Coefficient k_2	1e07		1e07	
Frame grabber Resolution	640x480 pixels by pixels		640x480 pixels by pixels	
CCD Array Resolution	640x480 pixels by pixels		640x480 pixels by pixels	
CCD Array Size	1/3 inch square		1/3 inch square	
Digitization Error	Yes		Yes	
Resolution noise	Gaussian noise @ 2 pixels level		Gaussian noise @ 2 pixels level	
Assumptions Made	1. Camera and jig are fixed during the whole calibration process 2. No rotation from world reference frame Fw to jig reference frame Fj 3. zero roll between Fw and camera reference frame Fc 4. Camera sits at origin of Fc and points directly at jig centre			

Table 3.14 Experiment results for Case #9 Effect of digitization Error, resolution noises, radial distortion and human measurement error on camera parameters

Case #9 Digitization error, resolution noise, radial lens distortion k1=1000 and k2=1e07, 75 control points, 1000 simulation runs								
LMDLT Method					Tsai Method			
Parameters	units	True Parameters	Mean	RMS Error	Parameters	True Parameters	Mean	RMS Error
T_x	m	3.5000	3.4758	0.0294	T_x	3.5000	3.4696	0.0473
T_y	m	2.5000	2.4348	0.0746	T_y	2.5000	2.5200	0.0568
T_z	M	4.0000	4.2473	0.2606	T_z	4.0000	3.9466	0.0867
roll	rad	0.0000	-0.0035	0.0035	Roll	0.0000	0.0077	0.0077
pitch	rad	1.1944	1.1923	0.0021	Pitch	1.1944	1.1926	0.0018
Yaw	rad	1.2490	1.2461	0.0029	Yaw	1.2490	1.2571	0.0081
f	m	0.0120	0.0123	4.1522e-004	f	0.0120	0.0116	4.6976e-004
s_x	None	N/A	N/A	N/A	s_x	1.0000	1.0024	0.0044
k_1	m^{-2}	N/A	N/A	N/A	k_1	1000	-678.6143	1.8460e+003
k_2	m^{-4}	N/A	N/A	N/A	k_2	1e07	5.7190e+007	5.8933e+007

Table 3.15 Experiment results for Case #10 Effect of digitization Error, resolution noises, radial distortion and human measurement error on camera parameters

Case #10 Digitization error, resolution noise, radial lens distortion k1=3000 and k2=1e07, 75 control points, 1000 simulation runs								
LMDLT Method					Tsai Method			
Parameters	units	True Parameters	Mean	RMS Error	Parameters	True Parameters	Mean	RMS Error
T_x	m	3.5000	3.4380	0.0644	T_x	3.5000	3.4720	0.0451
T_y	m	2.5000	2.3725	0.1326	T_y	2.5000	2.5287	0.0592
T_z	m	4.0000	4.4745	0.4823	T_z	4.0000	3.9386	0.0906
roll	rad	0.0000	-0.0300	0.0300	Roll	0.0000	-0.0073	0.0073
pitch	rad	1.1944	1.1971	0.0027	Pitch	1.1944	1.1961	0.0017
Yaw	rad	1.2490	1.2175	0.0315	Yaw	1.2490	1.2424	0.0066
f	m	0.0120	0.0125	5.8528e-004	f	0.0120	0.0116	4.9875e-004
s_x	none	N/A	N/A	N/A	s_x	1.0000	1.0021	0.0042
k_1	m^{-2}	N/A	N/A	N/A	k_1	3000	1.2645e+003	1.9324e+003
k_2	m^{-4}	N/A	N/A	N/A	k_2	1e07	5.5407e+007	6.1273e+007

3.10 Effect of Number of Calibration Points on Calibration Results

The study performed by [12] indicates that increasing the number of scene points improves the calibration accuracy for the stereo DLT camera calibration. The experiments described in this section were conducted to:

1. Investigate if increasing the number of scene points can improve the parameter estimation for LMDLT and Tsai's method; and
2. Investigate if this is also true when digitization error, resolution noise, radial distortion, and 3D scene point coordinate error are presented in the calibration environment.

The experimental setup for test Cases, #11, #12, #13, #14, #15, and #16 is the same as what is described in test case #10 in Table 3.13. The only modification is that test cases #11 and #14 use 27 scene points, test cases #12 and #15 use 75 points, and test cases #13 and #16 use 243 points. Test cases #11, #12, and #13 are for Tsai's method. The calibration method used for test cases #14, #15, and #16 is the LMDLT method.

Experimental results are listed in Tables 3.16 and 3.17. It can be seen from Table 3.16 that when Tsai's method was used, increasing the number of scene points has a positive impact on parameter estimation because the RMS errors decreased when the number of scene points was increased. The only exception for Tsai's method occurred in the scale factor s_x in Cases #12 and #13. But the RMS error for s_x in Case #12 is only slightly less than the one in Case #13. This anomaly may be caused by the noise in the system. When comparing Cases #14 and Case #15 for LMDLT method, the positive effect of using more scene points is not very clear. But when the number of scene points

was increased to 243 points, Case #16 provided better parameter estimates than Case #14 and #15.

Table 3.16 Experiment results for Tsai method: Cases #11, #12, and #13

1000runs, Tsai's Method, k1=3000, k2=e+07								
Parameters	units	True parameters	Mean, Tsai's method			RMS Errors, Tsai's method		
			Case #11	Case #12	Case #13	Case #11	Case #12	Case #13
			27	75	243	27	75	243
			Control	Control	Control	Control	Control	Control
			Points	Points	Points	Points	Points	Points
T_x	m	3.5000	3.4670	3.4720	3.5340	0.0675	0.0451	0.0393
T_y	m	2.5000	2.5322	2.5287	2.5082	0.0947	0.0592	0.0300
T_z	m	4.0000	3.9247	3.9386	3.9817	0.1355	0.0906	0.0420
Roll	rad	0.0000	0.0496	-0.0073	0.0021	0.0496	0.0073	0.0021
Pitch	rad	1.1944	1.1976	1.1961	1.1952	0.0032	0.0017	8.7210e-004
Yaw	rad	1.2490	1.2977	1.2424	1.2518	0.0487	0.0066	0.0028
f	m	0.0120	0.0112	0.0116	0.0116	9.9784e-004	4.9875e-004	3.8448e-004
s_x	m	1.0000	1.0040	1.0021	1.0042	0.0073	0.0042	0.0047
K1	m^{-2}	3000	-5.6099e03	1.2645e003	-6.8695	9.2156e003	1.9324e003	3.0473e003
K2	m^{-4}	1e07	3.6736e008	5.5407e007	1.5025e008	3.8516e008	6.1273e007	1.4291e008

Table 3.17 Experiment results for LMDLT method: Cases #14, #15, and #16

1000runs, LMDLT Method, k1=3000, k2=e+07								
Parameters	units	True parameters	Mean, LMDLT method			RMS Errors, LMDLT method		
			27 Control Points	75 Control Points	243 Control Points	27 Control Points	75 Control Points	243 Control Points
T_x	m	3.5000	3.4708	3.4380	3.4893	0.0388	0.0644	0.0142
T_y	m	2.5000	2.3792	2.3725	2.4252	0.1332	0.1326	0.0778
T_z	m	4.0000	4.4706	4.4745	4.2723	0.4871	0.4823	0.2770
Roll	rad	0.0000	0.0088	-0.0300	-9.2127e-004	0.0088	0.0300	9.2127e-004
Pitch	rad	1.1944	1.2017	1.1971	1.1930	0.0074	0.0027	0.0014
Yaw	rad	1.2490	1.2601	1.2175	1.2476	0.0111	0.0315	0.0014
f	m	0.0120	0.0123	0.0125	0.0121	4.9580e-004	5.8528e-004	1.6054e-004
s_x	none	N/A	N/A	N/A	N/A	N/A	N/A	N/A
K1	m^{-2}	N/A	N/A	N/A	N/A	N/A	N/A	N/A
K2	m^{-4}	N/A	N/A	N/A	N/A	N/A	N/A	N/A

3.11 Discussion of Results

In order to provide an overall parameter estimation accuracy comparison for the LMDLT and Tsai's algorithms, the RMS errors for the rotation angles, translation vector components, and focal length were compared. The comparison was done according to the following criteria:

1. If either LMDLT or Tsai's algorithm had smaller RMS errors for two or more out of the three roll-pitch-yaw rotation angles, the algorithm was decided to have higher parameter estimation accuracy and its name was assigned into the associated slots in Table 3.18.
2. A similar criteria to that used in Item 1 above was also applied to the components of the translation vector T_x , T_y , and T_z . The name of the more accurate algorithm was also allocated into the appropriate slots in Table 3.18.
3. Either the LMDLT or Tsai's method with the least RMS errors for focal length estimation was entered in Table 3.18.

Table 3.18 Summary of parameter comparison between LMDLT and Tsai's method

Test Cases		Rotation	Translation	Focal Length
Case #3, 3D error	2mm	LMDLT	LMDLT	Tsai
Case #4, 3D error	5mm	LMDLT	LMDLT	Tsai
Case #5, radial distortion	$k_1=1000$ $k_2=0$	Tsai	Tsai	Tsai
Case #6, radial distortion	$k_1=1000$ $k_2=1 \times 10^7$	Tsai	Tsai	Tsai
Case #7, resolution noise		Tsai	LMDLT	LMDLT
Case #8, digitization error		Tsai	Tsai	Tsai
Case #9, all errors and noise	$k_1=1000$, $k_2=1 \times 10^7$	LMDLT	Tsai	LMDLT
Case #10, all errors and noise	$k_1=3000$, $k_2=1 \times 10^7$	Tsai	Tsai	Tsai
Cases #11 ~ #16, Changing Number of jig points used $k_1=3000, k_2=1e07$	27 pts	LMDLT	Tsai	LMDLT
	75 pts	Tsai	Tsai	Tsai
	243 pts	LMDLT	Tsai	LMDLT

Table 3.18 suggests that the LMDLT method is less sensitive to 3D scene point coordinates errors and resolution noise. Tsai's method appears to be more robust to

radial lens distortion and digitization errors. The comparison is less conclusive for the test cases where the calibration environment contains 3D measurement errors, radial distortion, resolution noise, and digitization error. As seen in Table 3.18, the LMDLT yielded more accurate parameter estimation in Case # 9, however, in Case #10 when radial distortion is increased Tsai's method outperformed LMDLT. This observation seems to verify the importance of including radial lens distortion in the camera model.

The results from Cases #11 to #16, however, showed an interesting twist. The LMDLT's deficiency in dealing with radial distortion seems to be compensated by the improvement obtained by increasing the number of calibration points to 243. This raises an interesting question in the parameter study: When the LMDLT method estimates the rotation angles and the focal length more accurately, but without the ability to correct radial lens distortions, can the LMDLT method give better back projection accuracy than Tsai's method? To answer this question, a back projection study based on this specific test case (243 points) was carried out. The back projection study used the mean parameter estimates obtained in Cases #13 and #16 in Section 3.10. The back projection was performed for scene points located at zero height ($[Z]_w = 0$) in the world frame. With this assumption, we have the depth information required for back projection. More information about how the back projection is performed can be found in [27].

Figure 3.7 is the back projected 3D scene points described in the world frame for the LMDLT and Tsai's methods. The true scene points are located at grid intersections and at midpoints of grid intersections. Backprojection from Tsai's algorithm are shown as dots, and those from the LMDLT are shown as crosses. It can be observed that scene points estimated by Tsai's method are generally closer to their true positions.

Figure 3.7: The back projected 3D scene points for LMDLT and Tsai's method

In order to observe how the back projected errors are distributed relative to the camera origin O_c , the average back projection errors for each row in the X_w direction were plotted in Figure 3.8. The upper plot in Figure 3.8 shows the average horizontal distance of each scene row from the camera origin O_c . The middle plot shows the average backprojection errors for each row estimated by the LMDLT method. The lower plot shows the average backprojection errors for each row estimated by Tsai's method. According to Figure 3.8, when a row in the scene is further away from the camera centre O_c , the back projected scene points for the LMDLT method were further dislocated from their actual positions. However, the back projected results from Tsai's showed the opposite situation.

Figure 3.8: The average backprojection errors for each row in the X_w direction of the world frame

The back projection study suggests that the LMDLT method can estimate the position and the orientation of the camera more accurately than Tsai's method. However, the LMDLT method lacks the ability to estimate lens distortion and therefore has larger backprojection error than Tsai's method. These results emphasize the importance of including lens distortion in the camera model.

Chapter 4 Conclusions and Future Work

This thesis compares the LMDLT and Tsai's camera calibration methods. The comparison was done based on:

1. The accuracy of the estimated intrinsic and extrinsic camera parameters; and
2. The sensitivity of the camera parameters to noise and errors in the calibration environment.

In order to have the true camera parameters to be used in parameter accuracy comparisons, the calibration algorithms for LMDLT and Tsai's methods were performed using virtual images and calibration jig generated by MatCam. Monte Carlo Simulation was performed when needed to collect parameter statistics for comparison.

Chapter 1 provided an overview of camera calibration, the general problems with a geometrical camera calibration, and the significance of a calibration comparison study. The literature review highlighted problems of existing comparison studies in that true parameters are generally not known, and hence a study based on a virtual camera, virtual images and a virtual calibration jig was determined to be a sound approach to compare algorithm performance.

In Chapter 2, literature regarding camera calibration was reviewed. The DLT, LMDLT, and Tsai camera models were presented and the respective algorithms used to estimate parameters were reviewed and compared.

In Chapter 3, several experiments were conducted. These experiments explored the interaction between the estimated camera parameters with different types of errors and noise in the simulated camera calibration environment. The impact of the number of scene points on parameter estimation was also investigated. The simulation analysis in

Chapter 3 showed that Tsai's implementation is sensitive to noises in the image plane, especially in estimation of the distortion parameters k_1 and k_2 . The study shows that Tsai's method can estimate radial lens distortion reasonably well provided that these noise levels are kept as small as possible. This result may suggest that Tsai is more suitable for autonomous vehicle applications because camera systems with wide angle lenses, and hence more distortion, are commonly used to provide image information for path planning and collision avoidance. As discussed by Aitken [19], his implementation of Tsai's algorithm has been tested in applications of real unmanned vehicle systems and produced satisfactory results.

Sensitivity of distortion parameters to errors in Tsai's algorithm may be caused by the way k_1 and k_2 are estimated. During the Gauss-Newton iterations in the 2nd stage of Tsai's approach, the initial estimates of k_1 and k_2 were set to zero and are far away from their true values. This can cause nonlinear optimization to settle into a local minimum and hence the algorithm may not find a global optimal solution. In the future, as a further improvement to Tsai's algorithm, a technique to obtain more accurate initial estimates for k_1 and k_2 may improve estimation performance in the presence of noise and errors.

Another finding from the study is that with current implementations, both LMDLT and Tsai methods may produce results, under noisy or error-prone calibration environments, that are not suitable for scene measurement applications. Improvements have to be made to both LMDLT and Tsai methods to make their parameter estimation more accurate. For example, extensions to the LMDLT algorithm have been proposed that can include lens distortions [23]. However this results in nonlinear optimization for which reasonable initial estimates are required.

Another improvement that can be made to Tsai's approach is to implement a full scale non-linear parameter optimization following the 2nd stage. In the current implementation of Tsai's method, parameters estimated in stage 1 are held fixed during the stage 2 optimization. If a final optimization stage is introduced then all of the parameters can be adjusted to minimize the errors in the image plane.

The current study may provide useful information for camera calibration operators in the field of autonomous vehicle navigation. In the future, a new parameter study could be conducted to provide more analysis in terms of the applications of the calibration methods for high accuracy measurement or object localization. It is recommended that further studies implement the LMDLT with lens distortions, augment Tsai's method to include a final optimization for all parameters, and could also consider comparing the performance of these approaches to another alternative known as Weng's method [8].

The MatCam software used in the current parameter study requires code modification when the camera calibration method is changed. As a further improvement, the new study can build a better Graphical User Interface (GUI) to allow easy interaction with different camera calibration methods. In this way, the software package can be distributed to researchers or camera calibration operators to help them improve camera calibration techniques or help them to select the most appropriate calibration method from those available.

References

- [1] X. Armangué, J. Salvi, J. Batle, "A Comparative Review of Camera Calibrating Methods with Accuracy Evaluation," *Pattern Recognition*, Vol. 35, No. 7, July 2002, pp.1617-1635.
- [2] Hynek Bakstein, "A Complete DLT-based Camera Calibration with a Virtual 3D Calibration Object," M.sc. Dissertation, Faculty of Mathematics and Physics, Charles University, 1999.
- [3] G.E. Healey, R. Kondepudy, "Radiometric CCD Camera Calibration and Noise Estimation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp.267-276, Vol.16, March 1994.
- [4] Y.I. Abdel-Aziz, H.M. Karara, "Direct Linear Transformation into Object Space Coordinates in Close-range Photogrammetry," *Proceedings of the Symposium on Close-Range Photogrammetry*, Falls Church, 1971, pp.1-18.
- [5] R. Y. Tsai, "A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using off-the-shelf TV Cameras and Lenses," *IEEE Journal of Robotics and Automation*, Vol. RA-3, No.4, 1987, pp.323-344.
- [6] O. D. Faugeras, G. Toscani, "Calibration Problem for Stereo," *Proc. Int. Conf. Computer Vision Pattern Recognition*, Miami Beach, FL, June 1986, pp.15-20.
- [7] A. Wykurz, "Evaluation of Camera Calibration Performance with a Computer Based Camera Model," M.sc. Dissertation, Department of Systems and Computer Engineering, Carleton University, 2002.
- [8] J. Weng, P. Cohen, M. Herniou, "Camera Calibration with Distortion Models and Accuracy Evaluation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 14, pp. 965-980, October 1992.

- [9] S. Koppurapu, P. Corke, "*The Effect of Noise on Camera Calibration Parameters*," Graphical Mode, Vol. 63, No. 5, pp. 277-303, September 2001.
- [10] R. Kumar, A. R. Hanson, "*Robust Estimation of Camera Location and Orientation from Noisy Data Having Outliers*," Proceedings of Workshop on Interpretation of 3D Scenes, November 1989, pp. 52-60.
- [11] R. Kumar, "*Robust Methods for Estimating Pose and a Sensitivity Analysis*," GVGIP: Image Understanding, Vol. 60, No.3, November 1994, pp. 313-342.
- [12] J. Koffman, D. I. Miller, G. K. Knopf, A. Zecevic, "*Calibration and Measurement Accuracy of a Stereophotogrammetric System Using the Direct Linear Transformation*," Proceedings of the Third American Congress on Biomechanics, Waterloo, August 1998, pp.187-188.
- [13] C.C. Slama, C. Theurer, S.W. Henriksen, *Manual of Photogrammetry*, American Society of Photogrammetry, Falls Church, 4th Edition, 1980.
- [14] S.K. Gosh, *Analytical Photogrammetry*, Pergamon Press, 2nd Edition, 1998, pp. 115-121.
- [15] Jonathan M. Walton, <http://www.Gordian.com/users/jonboy/thesis/node1.html>, "Terrain Mapping from a Mobile Platform Through Optical Flow Techniques."
- [16] D.C. Brown, "*De-centering Distortion of Lenses*", Photogrammetric Engineering, Vol.32, No.3, 1966, pp. 444-462.
- [17] W.J. Smith, *Modern Optical Engineering*, SPIE press, New York, 3rd Edition, 2000.
- [18] P. Payeur, "ELG5163 Machine Vision Course Notes," School of Information Technology and Engineering, University of Ottawa, 2003.

- [19] V. C. Aitken, "*Calibration, navigation, spatial correspondence and data fusion for the Improved Landmine Detection Project D6300*," Defense Research Establishment Suffield, Contract No. W7702-7-R664.
- [20] G. Wei, S.D. Ma, "*Implicit and Explicit Camera Calibration: theory and experiments*", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 16, issue 5, May 1994, pp. 469-480.
- [21] J. Lavest, M. Viala, and M. Dhome, "*Do we Really Need an Accurate Calibration Pattern to Achieve a Reliable Camera Calibration?*" Proceedings of 5th European Conference on Computer Vision, June 1998, pp.158-174.
- [22] K. Astrom, B. Wittenmark, *Adaptive Control*, Addison Wesley, 2nd edition, 1995.
- [23] H. Hatze, "*High-precision Three-dimensional Photographic Calibration and Object Space Reconstruction Using a Modified DLT-approach*", J. Biomechanics, Vol. 21, No. 7, 1988, pp. 533-538.
- [24] N. R. Miller, "*A Technique for Obtaining Spatial Kinetic Parameters of Segments of Cinematographic Data*", J. Biomechanics, Vol 13, August 1980, pp.535-547.
- [25] N.R. Draper, H. Smith, *Applied Regression Analysis*, Wiley-Interscience, 3rd Edition, 1998, pp.505-512.
- [26] B.D. Bunday, *Basic Optimization Methods*, Edward Arnold Ltd, 1984.
- [27] V.C. Aitken, K. Russell, J. Mcfee, "*Geometrical and Optical Calibration of a Vehicle-mounted IR Imager for Landmine Localization*," in Proceedings SPIE 14th Annual International Symposium on Aerospace/Defense. Sensing, Simulation and Controls, AeroSense 2000, Detection and Remediation Technologies for Mines and Minelike Targets V, Orlando FL USA, April 24-28, 2000.

- [28] V.C. Aitken, “*Sliding Mode State Estimation for Non-linear Discrete-time systems: Applications in Image Sequence Analysis*,” PhD Dissertation, Department of Systems and Computer Engineering, Carleton University, 1995, pp. 245-246, 283-284.
- [29] M. H. DeGroot, *Probability and Statistics*, Addison-Wesley, 2nd Edition, 1986.
- [30] S. Hollis, http://www.math.armstrong.edu/statsonline/3/empirical_rule.html, “Empirical Rule”.
- [31] MatLab, Version 6.5, The MathWorks Inc., Natick MA, 2002.