MODELING ALTERNATIVES FOR IMPLEMENTING THE POINT-BASED BUNDLE BLOCK ADJUSTMENT

by

Chen Ma

A Dissertation

Submitted to the Faculty of Purdue University

In Partial Fulfillment of the Requirements for the degree of

Doctor of Philosophy

Lyles School of Civil Engineering

West Lafayette, Indiana

May 2021

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF COMMITTEE APPROVAL

Dr. James S. Bethel, Co-Chair

School of Civil Engineering

Dr. Ayman F. Habib, Co-Chair

School of Civil Engineering

Dr. Avinash C. Kak

School of Electrical and Computer Engineering

Dr. Michael Y. Zhu

Department of Statistics

Approved by:

Dr. Dulcy M. Abraham

ACKNOWLEDGMENTS

I want to sincerely thank my advisor, Professor James Bethel, for his long-term training, guidance, and encouragement during my Ph.D. study. The teaching of his knowledge, academic research and discussion, and the cultivation of skills have left me a deep memory. I am deeply thankful to my co-advisor, Professor Ayman Habib, for his strict and careful supervision of my work.

I would like to acknowledge my advisory committee: Professor Avinash C. Kak and Michael Yu Zhu, for their kind support, insightful comments, and encouragement.

I still want to thank Professor Jie Shan for letting me participate in his group meeting. I still want to thank Professor Wolfgang Förstner for the talking and discussion during his academic visit.

I want to thank Dr.Jenette Ashtekar, Professor David Ebert, and Professor Edward J. Delp for their financial support.

I also want to thank my friends Dr.Guizhen Wang and Dr. Jieqiong Zhao for their programming skills teaching and design experience sharing. I also want to thank my friends Yaxiong Chen, Fei Tao, and Qi Zhong for their encouragement.

Finally,I want to acknowledge my parents Weimin Ma and Xiuqin Zhou, and my sister Jie Ma for their endless family support.

I am also grateful to the Geomatics department for awarding me a one-year Roland S. Corning II Memorial Fellowship.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

BA	Block Adjustment
DOF	Degree of Freedom
EOP	Exterior Orientation Parameter
IOP	Interior Orientation Parameter
RANSAC	Random Sample Consensus
SFM	Structure from Motion
SLAM	Simultaneous Localization and Mapping
TFM	Three-Frame Model

ABSTRACT

This thesis examines the multilinear equations of the calibrated pinhole camera. The multilinear equations describe the linear relations between camera parameters and image observations in matrix or tensor formats. This thesis includes derivations and analysis of the trilinear equations through the point feature relation. For the four-frame and more than four frame cases, this paper gives derivations and analysis using a combination of the bilinear and trilinear equations to represent general multi-frame point geometry. As a result, a three-frame model (TFM) for general multi-frame point geometry is given. This model provides a concise set of minimal and sufficient equations and minimal unknowns.

Based on the TFM, there are two bundle adjustment (BA) approaches developed. The TFM does not involve the object parameters/coordinates necessary and indispensable for the collinearity equation employed by BA. The two methods use TFM as the condition equation fully and partially, replacing the collinearity equation. One operation using both TFM and the collinearity equation is designed to engage the object structures' prior knowledge. The synthetical and real data experiments demonstrate the rationality and validity of the TFM and the two TFM based methods. When the unstable estimate of the object structures appears, the TFM-based BA methods have a higher acceptance ratio of the adjustment results.

1. INTRODUCTION

Photogrammetry is an image-based 3D reconstruction technique. It is widely used in industrial structure inspection, aerial topographic mapping, satellite mapping, and resource management. The types of cameras vary with the application, the scale, and the platform. The complete photogrammetry system involves the camera, auxiliary sensors, and carrier platform. These techniques accommodate multi-model, multi-sensor, and multi-source data. Today's algorithm developments allow the reconstruction of large scenes in short periods of time. This extreme adaptability, high efficiency, and high precision give such techniques a prominent place in the Geomatics toolbox.

1.1 Research Background and Motivation

Photogrammetric techniques have been studied by researchers from many domains, such as survey, computer vision, and robot navigation. All the researchers make significant contributions to photogrammetric computer vision. Some image-based applications are still being studied and developed, and progress continues. For instance, structure from motion(SFM) and simultaneous location and mapping(SLAM) are recently developed techniques which are quite successful.

With the popularity of smartphones and personal computers, photogrammetric applications have become much more prevalent. Photography and video have become people's daily habits. Smartphones unify cameras, auxiliary sensors, and computation processors as a mini photogrammetry system. It brings massive data and takes the techniques anywhere.

Although having mature theory, well-developed hardware, and wide application possibilities, some practical challenges remain. The platform maybe unstable. The scene may not be static. The object structures may not be well distributed in the scene. The control points may not be visible or practical to create. For photogrammetric tasks, it is customary to have a particular camera model and operating model for a specific application. Scenes in close-range and aerial mapping tasks often have a short depth range for object features. The monocular SLAM is expected to work under many different conditions in one task, indoor and outdoor, close range and far range. So the demands of the task require much flexibility and adaptability from the processing system.

The prime motivation is to make the image-based techniques more adaptable. One example of the badly distributed object features is the distant object point. It can make the estimation and optimization numerically unstable. It is one of the problems studied in this research. The creation of control points may be a problem. Direct observation of camera exterior orientation may not be practical. However, the images sometimes provide useful object geometry, such as planar surfaces and some regular shapes. This information could be utilized in the algorithm to enhance the estimation and to resolve a datum defect. This is another consideration of this research.

An overview of the background and motivation for the research have been given here. The research involves the development and application of the multilinear equations and their use in the context of BA. Some relevant prior work is described in the literature review chapter.

1.2 Research Outline and Objectives

The most commonly used framework for analytical photogrammetry is the pinhole camera model. If the interior orientation is known, the camera is called a calibrated camera and uncalibrated otherwise. The interior orientation includes knowledge of the lens distortion and is assumed for the calibrated camera in photogrammetry. This research considers only the calibrated pinhole camera model.

In photogrammetric computer vision, the projection equations and multilinear relations are essential. They play significant roles in the algorithms, both non-iterative estimation and iterative optimization.

This study has three progressive levels. The first one is the derivation and analysis of the three-frame geometry. The second one is the derivation and analysis of the four-frame and more than four frame cases. And the third level is the BA method study. For each of these three levels, objectives will be given.

The first stage of study is the derivation and analysis of the three-frame geometry. New derivations of the trilinear equations are given. And a concise model for the three-frame case is given. The bilinear and trilinear equations are good filters to detect outliers in image matches.

The second phase includes new derivations and analysis of the four-frame geometry and more than four frame cases. The derivations and analysis are given for both of them. The three-frame work is expanded to the four-frame case. The work alludes to a possible way to represent the three-frame and the four-frame cases together. It underscores the utility of the bilinear and trilinear equations in general cases. As a result, the three-frame model is expanded and given for the general situation. It is assumed that the bilinear equation (coplanarity) is well known to everyone.

The final level is the BA algorithm study. Two TFM-based BA methods are developed to (1) fully and (2) partially replace the collinearity equations. One procedure uses both TFM and the collinearity equations to exploit prior knowledge of the object geometry. Some researchers have advocated similar BA approaches.

In the experiment chapter, the synthetic and real data experiments are presented. The purposes of the BA experiments are to check the validity of the methods first; secondly, to improve upon prior results; thus, displaying the advantages of this kind of BA approach.

1.3 Dissertation Structure

Chapter one is the introduction of the dissertation. Chapter two is the literature review giving much more details about prior, related studies.

Chapter three includes the original works, derivations, and contributions of this dissertation. The three levels are introduced as described in the previous section: (1) three frame geometry, (2) four-frame geometry, and (3) integration of these techniques into BA.

Chapter four presents experimental results of using these techniques in solving practical block adjustment problems. One auxiliary experiment is added for the weak geometry. The developed BA methods are evaluated and discussed in the two subsections. And real data experiments are given for practical validity checking.

Chapter five includes the conclusions and recommendations for future work.

The appendix includes all of the derivations of the trilinear equations and necessary derivatives.

1.4 Notation

Using conventional variable names, the first camera matrix is $P_1 = I[I|\mathbf{0}]$. The $\mathrm{i}-th$ camera matrix is $P_\mathrm{i} = M_\mathrm{i}[I| - \widetilde{\mathbf{C}}_\mathrm{i}]$. The object point $\mathbf{X} = [\widetilde{\mathbf{X}}, 1]$ is the homogeneous coordinate of the 3-dimensional point $\widetilde{\mathbf{X}} = [x, y, z]$. The image point $\mathbf{x} = [\tilde{\mathbf{x}}, 1]$ is the homogeneous coordinate of the 2-dimensional point $\tilde{\mathbf{x}} = [x, y]$. In the expression $\mathbf{x} \sim P\mathbf{X}$, the symbol $\sim$ means proportional relation. ${}^s\mathbf{x}$ and ${}^\mathrm{i}\mathbf{x}$ are the image points in the sensor and image coordinate systems. M_i and $\widetilde{\mathbf{C}}_\mathbf{i}$ denote the 3 by 3 homography matrix and optical center of the $\mathrm{i} - th$ camera. $\bar{\mathbf{C}}_\mathrm{i}$ is a normalized vector. No two cameras occupy the same location. The studied object point is not at infinity.

I is the identity matrix. $[\mathbf{u}]_\times$ is the three by three skew symmetric cross product matrix of $\mathbf{u}$. The cross product of vectors $\mathbf{u}$ and $\mathbf{v}$ is $\mathbf{u} \times \mathbf{v} = [\mathbf{u}]_\times \mathbf{v}$.

In this paper, bold letters, and lower case letters denote, respectively, vectors and elements of the matrices. Upper case letters without bold font represent matrices. Points represented by upper case are object space, lower case indicates image space.

A condition equation here refers to the equation which relates the measured value and the unknowns. A constraint equation here is an equation between the unknowns.

This paper uses knowledge of photogrammetry and computer vision. There is some overlap between the two of them. Some concise computer vision expressions are recommended to give brief and short expressions of geometry. For instance, there are nine trilinear equations, and each one has twenty-seven components. Because all equations have the same format, using the commonly used expression makes the equations very short and brief.

1.5 Summary of Original Contributions of this Research

1. New derivations of the trilinear equations.

This paper gives two new derivations of the trilinear equations for the three-frame and four-frame geometry.

2. A new analysis of multi-frame geometry.

The spatial geometry analysis approach gives a new three-frame model(TFM) for multi-frame geometry. The column vector approach shows that the quadrilinear equations are linear combinations of the trilinear equations.

3. This thesis gives two algorithms for conventional applications.

In the image coordinate prediction algorithm, the new method resolves the ambiguity in the wide-field camera. The relative distance estimation algorithm works well for close and distant object points.

4. There are two BA algorithms using bilinear and trilinear equations different from other researchers.

This thesis implements the new TFM model in the BA algorithm. Another new BA method, which combinates the TFM and collinearity equations with spatial constraints, is given in the thesis. The research discovers that the TFM-based BA algorithm has a higher acceptance ratio than the conventional method.

2. LITERATURE REVIEW

This research is an exploration of photogrammetry as viewed from the vantage point of computer vision. Significant differences in nomenclature may make familiar concepts seem unfamiliar and inaccessible. The author will try to anticipate and clarify ambiguous terms and phrases. The multilinear equations (bilinear, trilinear, quadrilinear) are the linear equations of the coefficients and the image correspondences in matrix or tensor formats. The linear coefficients are defined in terms of the nonlinear camera parameters. The image correspondences are the collections of image observations of the same object structure that occur in different images. The commonly used object structures are points and lines. These coefficients describe linear relations among image observations. The coefficients are nonlinear functions of the physical(EO) camera parameters. For instance, the coplanarity relation is represented by the bilinear equation. As is typical, the number of linear coefficients is always larger than the number of camera parameters.

Given some matched correspondences, there are some natural, valuable, and well-known questions. 1. What kind of geometry relations exist, and what equations describe them? 2. What kind of inner relations exist within each group of coefficients? 3. How to estimate the coefficients from the given correspondences? 4. How to retrieve physical camera information from the coefficients? Answering these questions reveals the properties of the multi-frame geometry. The following sections review the prior research concentrating on the above questions.

The bilinear equation has been extensively exploited in two-fame epipolar research. Thereafter, the linear expression became the prevailing pursuit of multi-frame geometry. While the bilinear, trilinear, and quadrilinear geometry was well studied, the duality existing within the multi-frame geometry was discovered subsequently. A summary of the duality follows the review of these linear equations.

For the question about how to represent the more complex relationships among more than four frames using the three kinds of linear equations, researchers have explored the minimal and sufficient expressions for the general multi-frame geometry.

For the multi-frame issue, the most popular and significant application is the BA. BA is a least squares optimization technique in photogrammetry. Traditionally, in photogrammetry, the collinearity equations were the foundation of BA. Some researchers have recast the BA using the multilinear equations instead. This research continues along the path and makes some new contributions.

The following sections review 1) the bilinear equation; 2) the trilinear equation; 3) the quadrilinear equation; 4) the duality; 5) the BA.

Hartley and Zisserman's(2004)[1]computer vision book is a good general reference for multi-frame geometry. Mikhail's(1976)[2] survey and adjustment book is recommended for knowledge of the adjustment model. Mikhail and Bethel's(2011)[3] photogrammetry book is a good source for photogrammetry knowledge. Trigg's (1998)[4] (2000)[5] BA review papers are a good study of the BA. Cooper and Cross(1998)[6] give a summary paper as a tutorial for survey and adjustment.

2.1 Two-Frame Geometry and the Bilinear Equation

The bilinear equation is the so-called coplanarity equation in photogrammetry. For a pair of cameras, it describes the coplanarity condition among one object point and two perspective centers of two cameras. This condition builds the equation for two image points from the image pair. It is widely used in the relative orientation of two-frame geometry.

The individual works describing the bilinear equation are Longuet-Higgins(1981)[7] and Tsai(1984)[8]. This equation describes the coplanarity equation of the two-frame geometry known as epipolar geometry. The resulting 3 by 3 matrix has rank 2 and is called the essential matrix, E, having 5 degrees of freedom (DOF) for the calibrated camera, and the fundamental matrix, F, having 7 DOF otherwise.

Werman and Shashua(1995)[9] showed it is the only relationship between the two frames. Rank 2 is the only inner constraint of the nine coefficients. The left and right null vectors of the matrix are the epipoles.

Luong and Faugeras(1996)[10] studied particular distributions of the object points, such as the critical surface and object plane. When the object points lie on the critical surface, the coefficients could not be defined uniquely. They[10] designed a six-point algorithm for the planar distribution.

Other non-iterative estimations include the linear eight-point algorithm given by Longuet-Higgins(1981)[7], Tsai[8], and Hartley (1997)[11]. The nonlinear seven-point algorithm was given by Sparr(1994)[12] and Hartley(1994)[13]. Hartly(1994)[13] showed that the six points from three views could provide the estimates of three F-matrices.

Luong et al.(1993) [14] studied the different parametrizations and objective functions for the robust iterative estimation of the F-matrix. When outliers appear in the matched data, Deriche et al.(1994)[15] used the least median square technique instead of least squares estimation. Torr and Murray(1997)[16] reviewed some of the robust algorithms and strategies. The random sample concensus(RANSAC) technique is a commonly used method to deal with the outliers, Fischer and Bolles(1981)[17]. Torr and Murray(1997)[16], Zhang(1998)[18] gave the review and evaluation of the estimation algorithms. Hartley's(1997)[11] non-iterative normalized eight-point algorithm has a comparable performance with the iterative methods.

Longuet-Higgins(1981)[7], Tsai(1984)[8] retrieved the rotation and translation parameters from the E-matrix. Luong et al.(1993) [14] recovered the epipoles from the F-matrix.

Faugeras and Robert(1994)[19] studied the image transfer and predication of point, line, and curvature among three views using the F-matrix. They found that the three F-matrices are not independent. This issue is revisited in the three-frame geometry.

2.2 Three-Frame Geometry and the Trilinear Equation

The trilinear equation can describe the intersection relations involving either a point feature or a line feature from three cameras. This equation expresses geometric information that cannot be expressed with bilinear equations. The well-known trifocal tensor is derived from the three-frame geometry. The physical camera parameters can be retrieved from the trifocal tensor.

In the 1960s, photogrammetrists studied the image triplet in analytical aero-triangulation. The image triplet was used as a unit in the adjustment and strip assembling, McNair(1962)[20], Keller(1967)[21], Fitzgibbon and Zisserman(1998)[22]. In the 1980s, computer vision researchers started the study of the structure from motion, SFM, problem from straight lines from three calibrated cameras, such as Liu and Huang(1988)[23]. This three-frame line study triggered the more general three-frame geometry research.

Researchers discovered the advantages of the image triplet over the stereo pair. For instance, Mikhail(1962)[24], McNair(1962)[20] and Anderson(1966)[25] demonstrated that the triplet gives greater strength in error detection. Hartley(1993)[26] pointed out that the uncalibrated image triplet is the minimal requirement for Euclidean reconstruction. For calibrated cameras, an image pair is sufficient for reconstruction. A line feature on an image pair gives no information or constraint about the camera orientation. These merits of the image triplet attract the attention of researchers.

Spetisakis and Aloimonos(1987)[27] (1990)[28], Liu and Huang(1988)[29], Weng et al.(1992)[30] derived three line-equations for calibrated cameras. The equations have three, three by three rank two matrices. These constitute the so-called trifocal tensor. Shashua(1994)[31] (1995)[32] discovered the nine point-equations among three frames. Hartley(1993)[33] extended Weng's derivation to the uncalibrated camera and he(1997)[34] identified the trifocal tensor equation in Shashua's nine trilinear equations. Spetsakis(1992)[35] also unified the point-equation with his previous line-equations. Faugeras and Papadopoulo(1998)[36] gave the derivations of both point and line features using Grassman Algebra. Heyden(1998)[37] (2000)[38] derived the reduced trifocal tensor with 15 non-zero coefficients for the point feature of the reduced camera. Shashua(1997)[39], Mendonça and Cipolla(1998)[40] studied the trifocal tensor of affine cameras. Hartley and Vidal(2004)[41] studied the multi-trifocal tensor for multi-body movement in scene.

In the three line-equations, there are two linearly independent equations. There are four linearly independent equations in the nine point-equations, which are the point-line equations involving one point and two lines in Shashua(1997)[39]. The 13 lines and 7 points give the linear estimate of the trifocal tensor coefficients. Shashua(1997)[39] and Avidan(1997)[42] (1998)[43]showed that the bilinear equation has the same format and geometry meaning as the point-line equation. He called the bilinear equation the bifocal tensor. Avidan and Shashua(1996)[44] indicated the existence of only 27 coefficients, which are computed by the feature coordinates. The frame order only impacts the arrangement of coefficients.

Shashua and Werman(1995)[45] (1995)[46] (1997)[39], and Avidan(1998)[43] also revealed three functional inner structures of the 27 coefficients. One is the trifocal tensor, and the other two are homography slices. Much research about the internal constraints and retrieval of camera parameters relies on Shashua's discoveries. Faugeras and Papadopoulo showed that once the epipolar geometry is known, there is only one algebraically trilinear relation. Given $(n+2)$ frames, Shashua(1996)[47] and Avidan(1996)[44] declared that n trifocal tensors are sufficient. Avidan and Shashua(1997)[48] developed a trifocal operator converting one tensor to another and illustrated the deep dependency among the n tensors.

Within the 27 coefficients, there are eight inner constraints. Weng et al. (1992)[30] gave the so-called rank and epipolar constraints for calibrated cameras. Faugeras and Papadopoulo(1998)[36] (1998)[49] (1998)[50] gave three sets of constraints. Heyden(1998)[37] (2000)[38] plumbed the constraints for the reduced trifocal tensor. All of these sets have 12 constraints, which are sufficient but neither minimal nor independent.

The pursuit of a sufficient, minimal, and independent set was attempted by many researchers. Canterskis [51] gave the set. Ressl(2003)[52], Alzate and Tortora (2010)[53], Heinrich and Snyder(2011)[54], provided other different sets. Alzate and Tortora(2010)[53]gave the lowest degree constraints.

The estimation of the trifocal tensor was widely studied. Spetisakis and Aloimonos(1987)[27] (1990)[28], Liu and Huang(1988)[29], Weng et al.(1992) [30], Hartley(1994)[55] (1995)[56], and Shashua(1995)[32] brought about non-iterative linear algorithms using the line-equation, point-equations, or both. The non-iterative nonlinear solution algorithms include different algorithms using six-point correspondences, Heyden(1994)[57], Quan(1995) [58], and Hartley, Dano(2000)[59]. Three camera matrices can be retrieved from correspondences using Quan's (1995) [58] six-point-invariant. Heyden's algorithm (1995)[60] involves the coplanarity of object points. Torr and Zisserman(1997)[61] furnished one non-constrained iterative algorithm. Hartley(1998)[62], Kuang te al.(2014)[63] gave the constrained iterative algorithms. Kuang et al. (2014) studied the estimation using line-equations. Hartley and Dano(2000)[59], Schaftalizky(2000)[64] expanded Quan's six-point algorithm from three views to more views. Heinrich and Snyder(2011)[65] evaluated the estimation techniques and developed a RANSAC based robust strategy. They recommended Hartley's algorithm[56]. When given more than 10 points, it provides nearly the maximum likelihood estimate.

Spetisakis and Aloimonos(1987)[27] (1990)[28], Liu and Huang(1988)[29], and Weng et al.(1992)[30] recovered the rotation matrices and translation for a calibrated camera. Then the retrieval of physical camera parameters could be realized. Shashua and Werman(1995)[45] (1995)[46], Hartley(1993)[33] (1994)[55] retrieved epipoles, F-matrices, and camera matrices from the trifocal tensor for uncalibrated cameras.

Avidan and Shashua(1997)[48], Mayer(2002)[66] developed a trifocal tensor based view synthesis technique that overcomes some deficiencies of Faugeras's F-matrix-based method. Shashua and Werman(1995)[45] (1995)[46] indicated the projective reconstruction could be realized using the trifocal tensor without retrieving explicit camera matrices. Shashua(1995)[67] (1997)[39] gave a tensor brightness equation for surface reconstruction. This equation relates spatial position and radiometric level, which is applicable for smooth surfaces, Stein and Shashua(2000)[68]. Luong and Viéville(1996)[69] gave the minimal parameterization of the single, two, and three views for all projective, affine, perspective cameras (projective = uncalibrated, perpective = calibrated, affine = viewing only a planar object). This study is quite helpful for understanding multi-frame geometry.

2.3 Four-Frame Geometry and the Quadrilinear Equation

The quadrilinear equation can describe the intersection relations involving a point or a line feature from four cameras. The physical camera parameters can be retrieved from the quadrifocal tensor. The estimation of quadrifocal tensor involves the observations from four frames, which is numerically more stable than the fundamental matrix and trifocal tensor.

Werman and Shashua(1995)[9] first studied the properties of the four-frame linear equation. They deduced that the quadrilinear equations have 81 coefficients and 16 linearly independent equations. Their analysis predates any quadrilinear formula. Trigg(1995)[70] (1995)[71] derived the first formula of quadrilinear equations for points and lines later. There are a total of 27 equations. Heyden(1995)[72] (1997)[73] (1998)[37] (2000)[38] studied the reduced quadrifocal tensor with 36 non-zero coefficients and the one described previously.

Hartley(1995)[74], Faugeras and Mourrain(1995)[75], and Heyden(2000)[38] also provided the frameworks for all multilinear equations for both point and line features. Faugeras and Mourrain(1995)[75] pointed out that there are only these three types of equations(bilinear, trilinear, and quadrilinear). The necessary and sufficient conditions to estimate the three groups of coefficients are studied, by Heyden(1995)[72] and Hartley(1995)[74]. It requires at least 6 points for four frames, the same as for three frames.

Heyden(1995)[72] (1998)[37] provided the necessary and sufficient constraints for the reduced quadrifocal tensor and necessary constraints for the usual one.

Heyden(1997)[73] devised a six-point algorithm for the reduced quadrifocal tensor, partially avoiding the constraints. Hartley(1998)[76] evolved the algorithm by fully avoiding the constraints. He constructed one linear algorithm and two iterative ones.

Heyden(1998)[37] (2000)[38] retrieved the trifocal tensor from the quadrilinear tensor and then gave a unique representation and minimal parametrization for recovering camera matrices. Hartley(1998)[76] recovered reduced camera matrices from the reduced tensor.

Some researchers considered the relation among the three types of equations. Triggs(1995)[71] thought that the quadrilinear equations are some combinations of the other two. Heyden(1997)[73] thought that some linear transformation could convert reduced quadrilinear equations to other reduced ones. Faugeras and Mourrain(1995)[75] demonstrated that if the bilinear and trilinear equations are satisfied, the quadrilinear equations are guaranteed simultaneously by analyzing one equation. Although the expressions are rich and varied, there is no new information in the quadrilinear equations.

For more camera geometry, Heyden(2000)[38] elaborated the methods of representation using each of the three types individually, where the simplest one is the bilinear representation. Heyden and Åström (1996)[77] (1997)[78] studied the bilinear representation and demonstrated that the trilinear equations are necessary and indispensable for the trifocal plane's object structures.

2.4 Duality Theory

The multilinear equations exist not only in the multi-frame geometry but also in the multi-point geometry. There could be one point on multiple frames or one frame with multiple points. There is a duality between the two types of geometry. The study of this duality may allow a reduction of unknowns.

Carlsson(1993)[79] (1995)[80] discovered the absolute linear invariant from Quan's six-point-invariant. Object point and camera position play symmetrical roles in the invariant equation. There is a duality between these two parameters. In the multilinear coefficient estimation algorithms, the non-iterative linear methods always estimate the coefficients first. They retrieve the camera matrices later and compute object points finally. The non-iterative nonlinear methods always estimate the object points first, recover the camera matrices secondly, and calculate the coefficients finally. The duality gives the two methodologies. For the uncalibrated camera, six object points are necessary, as expressed by the name six-point-invariant. So there are many different approaches for six-point algorithms for three-, four- and multi-frame cases.

Weinshall et al. (1995)[81] (1996)[82] determined that the multi-frame geometry's duality is the multi-point geometry involving the m object points in one view. The two-, three-, and four-point equations are dual to the four-, three-, and two-frame equations respectively.

Carlsson and Weinshall(1998)[83] summarized the duality theory as one algorithm could work on the two dual questions. They develop the idea and provided careful comparisons of these algorithms one by one. Finally, they(1998)[83] said:" equations on scene structure and equations on camera geometry can be used to reduce the number of unknowns to be solved for and should be exploited in similar ways."

2.5 Bundle Adjustment

The bundle adjustment is the assembly of ray bundles from multiple images in photogrammetry. The bundles derive from image observations, and the optimization is usually done by least squares. It accommodates image observations and all kinds of unknowns, such as platform parameters, camera parameters, and lens distortion parameters. It also integrates multi-source, multi-model, and multi-sensor data in the whole optimization. These outstanding properties make this technique the most widely used refinement method in the image-based applications.

Fred Doyle(1964)[84] gives a thorough history of the development of analytical photogrammetry. It was a long process with many contributors from the 1750's through the present. Such contributors include Lambert, Finsterwalder, Pülfrich, von Gruber, Church, Merritt, Schut, Schmid, and Brown. Helmut Schmid and Duane Brown(1958)[85] really developed the bundle adjustment, BA, into its modern concept in the 1950's and 1960's. From that time, BA techniques have grown in scope and variety of application. The two primary branches of conventional photogrammetry are aerial mapping and close-range mapping, Brown(1976)[86] and Granshaw(1980)[87]. From the practical experience of both, the significant core considerations of the photogrammetric task and BA are listed here: 1. Network design; 2. Camera calibration; 3. Gross error detection; 4. Constraints, minimal constraints, and free network adjustment; 5. Recursive, sequential and, online methods; 6. Computation efficiency.

This section summarizes some prior research around the above topics. And the alternative methods are reviewed: 1. Model changing (equation types and adjustment models); 2. Line feature BA; 3. Distant object points.

Network design is the planning stage of the photogrammetric task. Some elements such as camera positions, camera internal geometry, photo overlap, and control point distribution are designed before the fieldwork to get the required accuracy, Fraser(1984)[88]. Mason(1995)[89] developed assist software for network design.

Brown(1976)[86] and Granshaw(1980)[87] demonstrated that self-calibrating approaches could reduce camera system error substantially and bring better accuracy. These methods accommodate camera distortion models for system error. Brown(1971)[90] studied the camera distortion models such as radial and decentering distortion. Faig(1975)[91] provided camera calibration methods. Ackermann(1981)[92] studied the problem of selection and reliability of camera distortion parameters. Alharthy and Bethel(2002)[93] gave the calibration method and lens distortion model for a multi-band camera.

In practice, gross-errors such as image feature measurements always exist. The BA is susceptible to gross errors. Grün(1982)[94] and Förstner(1985)[95] studied gross-error detection and location. El-Hakim(1984)[96] provided a step-by-step detection strategy.

For close-range photogrammetry, the free network BA is required for eliminating uncertainty dependencies on the control point location. Dermanis(1994)[97] gave the derivation of the inner constraints resolving the rank defect of the normal equations. He introduced four different approaches for the implementation of minimal gauge internal constraint. McLauchlan(1999)[98] (1999)[99] provided the free network BA for Euclidean and projective reconstruction. Morris et al.(1999)[100] developed the normal covariance technique reflecting the inner uncertainty and developed an efficient method to estimate the full covariance.

In batch BA, the collection of condition equations is prepared in advance, and unknown parameters are updated numerically without deletion and adding. In some real-time applications, online data editing is required. Mikhail and Helmering(1972)[101] studied the Schur-component based methods in data editing, adding, and reduction. The size of the normal equations may change during the computation. Blas(1983)[102] and Holm(1989)[103] studied the Givens transformation in the sequential adjustment. They showed that the Givens transformation has numerical stability, computation efficiency, and advantages in parallel processing. McLauchlan and Murray(1995)[104] studied recursive algorithms for projective and Euclidean SFM. McLauchlan(2000)[105] researched the condition for removing unknowns and the trade-off between accuracy and efficiency.

Agerwal et al.(2011)[106] furnished a large-scale SFM instance building a city model with more than 150k images, which calls for higher performance on computation efficiency in BA. In photogrammetric reconstruction, most cameras and most object structures have no direct interaction, making the normal matrix's special sparse structure, Elassal(1969)[107];Lourakis and Argyros(2009)[108]. Many sparseness techniques and software algorithms are developed. Elassal(1969)[107] designed a recursive Schur-complement based method for the inverse of the normal matrix. Lourakis and Argyros(2009)[108] developed a software package called SparseBA(SBA), emphasizing flexibility and computation efficiency. This software uses Cholesky factorization of the normal matrix to save computation. Although second-order optimization is recommended for small-scale instances by Trigg et al.(2000)[5], thousands of photos make the direct inverse of the reduced normal matrix impractical. Byröd and Åström(2010)[109] implemented the conjugate gradients BA(CGBA). Agerwal et al.(2010)[110] developed large-scale BA(LBA). Wu et al.(2011)[111] created parallel computing BA(PBA) for the multi-core computer. For large-scale computing, LBA and PBA implement preconditioned conjugates and inexact-Newton-like updating techniques. Hänsch et al.(2016)[112] compared and evaluated some parallel computing optimization methods. The combination of conjugate gradient techniques and the Levenberg-Marquarelt algorithm is recommended for medium-scale instances.

The conventional BA customarily uses the collinearity equation as the condition equation. It involves and refines the object points and camera parameters. Some alternate approaches of BA implement the multilinear equations as the condition equations. Utilizing the multilinear equations causes the changing of the adjustment model and deferring consideration of object structure parameters. Ressl(2000)[113] studied the relative orientation using the trifocal tensor and model changing, but no experiment was given. Liu et al.(2003)[114] and Rodríguez et al.(2011)[115] developed one technique only employing the bilinear equation. Steffen et al.(2010)[116], Indelma(2012)[117], Scheider et al.(2017)[118] carried out the bilinear and trilinear equations based approach. The object structure parameters are disregarded, so Indelma calls this approach light BA.

Taylor and Kriegman(1995)[119]developed the BA using the line feature. For the line feature, they designed a particular camera projection matrix and objective function. McLauchlan and Murray(1995)[104] developed the SFM and BA algorithm incorporating the point and line features.

Distant object points can make the position estimation and BA numerically unstable, so some implementations avoid them. The utilizing of remote object points in real-time mapping and batch BA was studied by Montiel et al.(2006)[120], Civera et al.(2008)[121]. Montiel et al.(2006)[120], and Civera et al.[121] developed inverse distance techniques for distant object points. Schneider et al.(2012)[122] made the BA robust to account for very distant object points. They demonstrated the advantages of employing the distant object points in camera orientation.

3. DERIVATIONS AND ANALYSIS OF MULTI-FRAME GEOMETRY

This chapter gives a new and comprehensive analysis framework for the multilinear point equations. The analysis includes the spatial geometry interpretations and algebraic interpretations of the multilinear equations. This framework has three approaches.

The first approach is algebraic algebraic approach and is the so-called determinant approach, which gives the expressions of condition equations. The second approach is based on spatial geometry and gives interpretations and analysis of these relations. The third approach, also algebraic is the so-called column vector approach, which provides an analysis of the individual types of equations and illuminates the relations among the equations. It provides insight about the dependencies.

The two algebraic approaches exploit the well-known rank defect of the matrix . The determinant approach has the same starting point as Trigg(1995)[71], Hartley(1995)[74], and Heyden(2000)[38]. The column vector approach simplifies the analyses of Trigg(1995)[71], Faugeras and Mourrain(1995)[75]. The determinant approach is used for the derivation of the trilinear equations.

3.1 Camera Projection

The pinhole camera model is the most commonly used and considered in today's photogrammetry and computer vision applications. It is the foundation of this thesis. This camera model makes projective two-dimensional images of the real three-dimensional world via the camera position, rotation, and geometry including lens distortions. During the projection, the spatial object distance is lost. There are some kinds of distortions in real camera systems, such as radial and decentering lens distortion, making cameras and images not always ideal. The calibrated camera functions like a distortion free camera, which is assumed to have perfect projection.

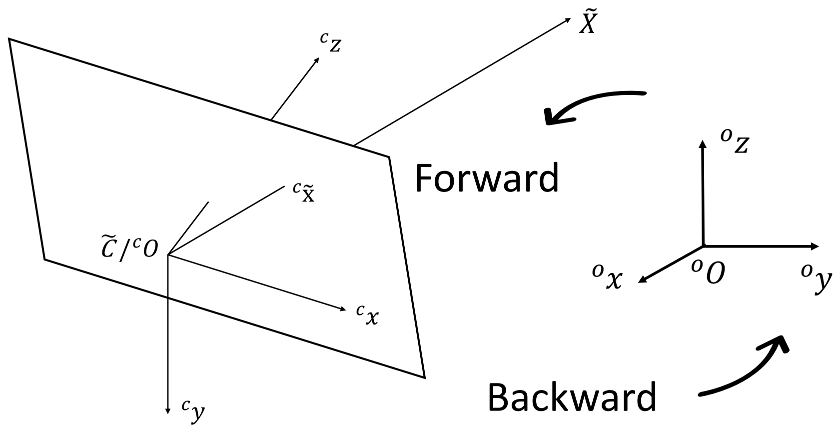


Figure 3.1. The pinhole camera and related coordinate systems.

The collinearity equations are the usual way to express the relationship between object and image coordinates in photogrammetry and computer vision. In computer vision, the projection function is ${}^{s}\mathbf{x} \sim KR[I| - \widetilde{\mathbf{C}}]\mathbf{X}$ where coordinate vectors without tilde are homogenenous, and vectors with tilde are cartesian. The camera projection is defined by the three by four camera matrix up to a scalar involving interior orientation, camera rotation, and camera position. K is the upper triangular matrix having interior orientation parameters. R is the rotation matrix having three parameters. $\widetilde{\mathbf{C}}$ is the camera position vector.

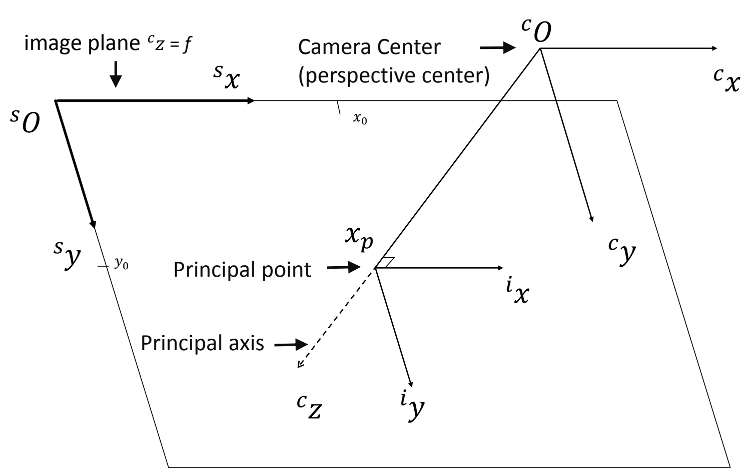


Figure 3.2. The pinhole camera interior orientation.

K has focal length parameters f and principal point location x_0 and y_0.

$$K = \begin{bmatrix} f & 0 & x_0 \\ 0 & f & y_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.1}$$

In photogrammetry, the collinearity eqations are

$$\begin{cases} {}^s x = x_0 + fu/w + \Delta x \\ {}^s y = y_0 + fv/w + \Delta y \end{cases} \quad \text{and} \quad \begin{bmatrix} u \\ v \\ w \end{bmatrix} = R_f^T \left[\widetilde{\mathbf{X}} - \widetilde{\mathbf{C}} \right]. \tag{3.2}$$

In the equations (3.2), Δx and Δy are corrections for lens distortions. The forward rotation from object to image space is represented by R_f, which rotates object space basis vectors into image space basis vectors, all expressed in object space coordinates. To effect the transformation from object space vectors in object space coordinates into images space vectors in image coordinates we use the transpose of R_f, R_f^T. This is a subtle but important distinction. Equations (3.2) are used in the camera calibration, spatial intersection and resection. As mentioned above the rotation matrix used in equations (3.2) is R_f^T. Since $R_f^T = R_f^{-1} = R_b$, we are using the backward rotation matrix in equations (3.2). This slight confusion of language is explained above. In real work, the camera could be calibrated using a printed template. There are mature calibration techniques to ensure that the added lens distortion parameters can be determined. Note that the calibration is only strictly valid at the focus position of the calibration setup. Some researchers fix the focus at that position for all work until a new calibration is done. After calibration, the Δx and Δy are computed to guarantee the calibrated camera has the ideal projection. It is written as $\mathbf{x} \sim R[I| - \widetilde{\mathbf{C}}]\mathbf{X}$ concisely.($\sim$ means proportional to)

One object point gives two equations to one camera in the equations (3.2). There is a modified projection relation here, as shown below.

$$
\begin{aligned}
\mathbf{x} &\sim M[I| - \widetilde{\mathbf{C}}]\mathbf{X} \text{ where } M = KR \text{ or } R \\
M^{-1}\mathbf{x} &\sim [I| - \widetilde{\mathbf{C}}]\mathbf{X} \text{ where } \chi = M^{-1}\mathbf{x} = [u, v, w]^T \\
[\chi]_\times \chi &\sim [\chi]_\times [I| - \widetilde{\mathbf{C}}]\mathbf{X} = \mathbf{0} \\
&\quad \text{where the } [\cdot]_\times \text{ notation indicates the vector cross product} \\
&\quad \text{by matrix mutiplication}
\end{aligned}
$$

The above derivation shows $[M^{-1}\mathbf{x}]_\times [I| - \widetilde{\mathbf{C}}]\mathbf{X} = 0$. If $w = \chi_3 \neq 0(\mathrm{i} = 1)$, or $v = \chi_2 \neq 0, (\mathrm{i} = 2)$, or $u = \chi_1 \neq 0(\mathrm{i} = 3)$, there are three corresponding matrices S_i with $det(S_\mathrm{i}) \neq 0$. For instance, if $\chi_3 = w \neq 0$, there is a matrix S_1 which gives the equation below.

$$
\begin{aligned}
S_1 [\chi]_\times [I| - \widetilde{\mathbf{C}}]\mathbf{X} &= \mathbf{0} \\
\begin{bmatrix} 1 & 0 & -u/w \\ 0 & 1 & -v/w \\ 0 & 0 & 0 \end{bmatrix} [I| - \widetilde{\mathbf{C}}]\mathbf{X} &= \mathbf{0} \\
\begin{bmatrix} 1 & 0 & -u/w & -c_1 + c_3u/w \\ 0 & 1 & -v/w & -c_2 + c_3v/w \\ 0 & 0 & 0 & 0 \end{bmatrix} \mathbf{X} &= \mathbf{0} \\
&\quad \text{or more familiarly,} \\
\begin{bmatrix} 1 & 0 & -u/w \\ 0 & 1 & -v/w \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} &= \begin{bmatrix} c_1 - c_3u/w \\ c_2 - c_3v/w \end{bmatrix} \\
&\quad \text{where } [c_1, c_2, c_3]^T = \widetilde{\mathbf{C}}
\end{aligned}
\tag{3.3}
$$

The expressions of three matrices S_{i} are

$$S_1 = \begin{bmatrix} -uv & u^2+w^2 & -vw \\ -v^2-w^2 & uv & uw \\ uw & vw & w^2 \end{bmatrix} \cdot \frac{1}{(u^2+v^2+w^2)w}$$

$$S_2 = \begin{bmatrix} uw & vw & -u^2-v^2 \\ uv & v^2 & vw \\ v^2+w^2 & -vu & -uw \end{bmatrix} \cdot \frac{1}{(u^2+v^2+w^2)v}$$

$$S_3 = \begin{bmatrix} u^2 & uv & uw \\ -uw & -vw & u^2+v^2 \\ uv & -u^2-w^2 & wv \end{bmatrix} \cdot \frac{1}{(u^2+v^2+w^2)u}$$

The S_{i}^{-1} is the matrix, which replaces the $(4-\mathrm{i})th, \mathrm{i} \in [1,3]$ column of the $[\chi]_\times$ using the vector χ. This is a convenient way to build the matrix S_{i}. Equation (3.3) is rewritten as the one below. The detailed steps of the derivation are in the appendix (A.1).

$$\begin{bmatrix} 1 & 0 & m & n \\ 0 & 1 & p & q \end{bmatrix} \mathbf{X} = \mathbf{0} \text{ and } \begin{bmatrix} m & n \\ p & q \end{bmatrix} = \begin{bmatrix} -u/w & -c_1 + c_3 u/w \\ -v/w & -c_2 + c_3 v/w \end{bmatrix} \tag{3.4}$$

The above equation is noted as $A_{s1}\mathbf{X} = \mathbf{0}$. The "1" in the subscript refers to a single frame. If the camera is the first one, then R is the identity matrix I and $\widetilde{\mathbf{C}} = \mathbf{0}$. This yields,

$$\begin{bmatrix} 1 & 0 & -x & 0 \\ 0 & 1 & -y & 0 \end{bmatrix} \mathbf{X} = \mathbf{0} \text{ and } \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \mathbf{x}. \tag{3.5}$$

One three dimensional object point gives $2n$ equations for n cameras. So the number of redundant equations is $(2n-3)$. This is the starting point of the column vector approach.

3.2 Two-Frame Geometry

This section reviews the two-frame geometry to motivate the following analyses. Although the epipolar geometry is well-studied, we do it in a way to anticipate the extension to more than two frames. The subsequent sections use the properties of two-frame geometry. Using the conventional assumption $P_1 = I[I|\mathbf{0}]$, and the second camera matrix is $P_2 = R_2[I| - \widetilde{\mathbf{C}}_2]$.

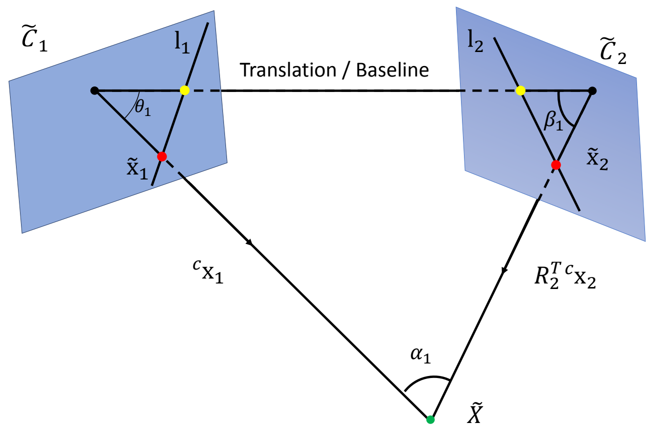


Figure 3.3. The epipolar geometry.

The bilinear equation describes the coplanarity condition, where the three vectors $\mathbf{x}_1$, $R_2^T\mathbf{x}_2$, and $\widetilde{\mathbf{C}}_2$ are coplanar and therefore have a common normal vector.

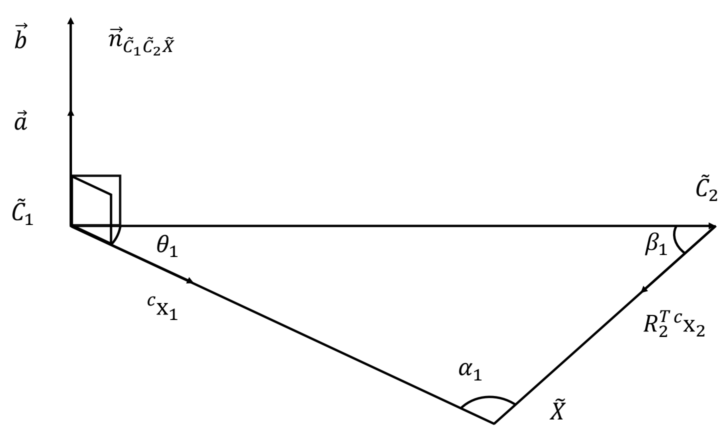


Figure 3.4. The two normal vectors of two frames plane.

In the figure (3.4), the two normal vectors are noted as $\mathbf{a}$ and $\mathbf{b}$, and $\mathbf{a} = [a_1 \quad a_2 \quad a_3]^T$.

$$\mathbf{a} = \mathbf{x}_1 \times R_2^T \mathbf{x}_2 \quad \mathbf{b} = \widetilde{\mathbf{C}}_2 \times R_2^T \mathbf{x}_2 \quad \mathbf{a} \parallel \mathbf{b} \Leftrightarrow \mathcal{B}_{12}$$

where script $\mathcal{B}_{12}$ a bilinear equation betweeen frame 1 and 2.

For the calibrated camera, one way to state the coplanarity equation is $\mathbf{x}_1^T \mathbf{b} = 0$.

$$\mathbf{x}_1^T \mathbf{b} = 0 \Leftrightarrow \mathbf{x}_1^T \left[\widetilde{\mathbf{C}}_2\right]_\times R_2^T \mathbf{x}_2 = 0 \quad \Leftrightarrow \mathcal{B}_{12} \tag{3.6}$$

The three-by-three matrix $E = \left[\widetilde{\mathbf{C}}_2\right]_\times R_2^T$ is known as the essential matrix. If we allow the first frame to have non-identity rotation matrix and non-zero position, then the projection equation is $\mathbf{x}_1 = R_1[I| - \widetilde{\mathbf{C}}_1]\mathbf{X}$, The base vector becomes $\widetilde{\mathbf{C}}_2 - \widetilde{\mathbf{C}}_1$, the first frame object vector is $R_1^T \mathbf{x}_1$, and the revised equation for $\mathbf{b}$ is

$$\begin{aligned} \mathbf{b} &= (\widetilde{\mathbf{C}}_2 - \widetilde{\mathbf{C}}_1) \times R_2^T \mathbf{x}_2 \text{ , or} \\ \mathbf{b} &= \left[\widetilde{\mathbf{C}}_2 - \widetilde{\mathbf{C}}_1\right]_\times R_2^T \mathbf{x}_2 \text{ ,} \end{aligned}$$

and the revised equation for coplanarity in equation (3.6) is

$$\mathbf{x}_1^T R_1 \left[\widetilde{\mathbf{C}}_2 - \widetilde{\mathbf{C}}_1\right]_\times R_2^T \mathbf{x}_2 = 0. \tag{3.7}$$

3.2.1 Analysis By Column Vector Approach

Back to the assumption of zero rotation and zero position for frame 1, one object point gives four equations for these two frames $P_1 = I[I|\mathbf{0}]$ and $P_2 = R_2[I| - \widetilde{\mathbf{C}}_2]$. So the number of redundant equations is 1. Using the simplified projection (3.5) and (3.4), the four equations are below in the stacked matrix format.

$$\begin{bmatrix} 1 & 0 & m_1 & 0 \\ 0 & 1 & p_1 & 0 \\ 1 & 0 & m_2 & n_2 \\ 0 & 1 & p_2 & q_2 \end{bmatrix} \mathbf{X} = \mathbf{0} \tag{3.8}$$

The above equation is noted as $A_{s2}\mathbf{X} = \mathbf{0}$. The subscript 2 denotes the 2-frame equation. Gaussian elimination gives the equation below.

$$\begin{bmatrix} 1 & 0 & m_1 & 0 \\ 0 & 1 & p_1 & 0 \\ \mathbf{0} & \mathbf{0} & \mathbf{U}_2 & \mathbf{V}_2 \end{bmatrix} \mathbf{X} = \mathbf{0} \tag{3.9}$$

$$\text{where } \begin{bmatrix} \mathbf{U}_2 & \mathbf{V}_2 \end{bmatrix} = \begin{bmatrix} m_2 - m_1 & n_2 \\ p_2 - p_1 & q_2 \end{bmatrix} \text{ and } \mathbf{U}_2 \parallel \mathbf{V}_2$$

There are two vectors, $\mathbf{U}_2$ and $\mathbf{V}_2$, in the lower right corner of the matrix in (3.9). The rank of the stacked matrix is less than four making the $\mathbf{U}_2 \parallel \mathbf{V}_2$. This gives the equation below.

$$\mathbf{U} \parallel \mathbf{V} \quad \Leftrightarrow \frac{m_2 - m_1}{p_2 - p_1} = \frac{n_2}{q_2} \quad \Leftrightarrow \mathcal{B}_{12} \tag{3.10}$$

The above equation(3.10) shows that there is only one equation between the two frames, the coplanarity equation. It is proved by the Werman and Shashua(1995)[9]. The assumption, $P_1 = I[I|\mathbf{0}]$, makes the expression concise without losing information.

3.3 Three-Frame Geometry

This section talks about three-frame geometry and the trilinear point-point-point equations. The skeletons of two derivations of the trilinear equations are given here. Full details of the derivation are in the appendix(A.2). Image point coordinates of a space point $\mathbf{X}$, are assumed given. The matrix determinant approach gives the derivation. Two analyses are by the spatial geometry approach and column vector approach sections.

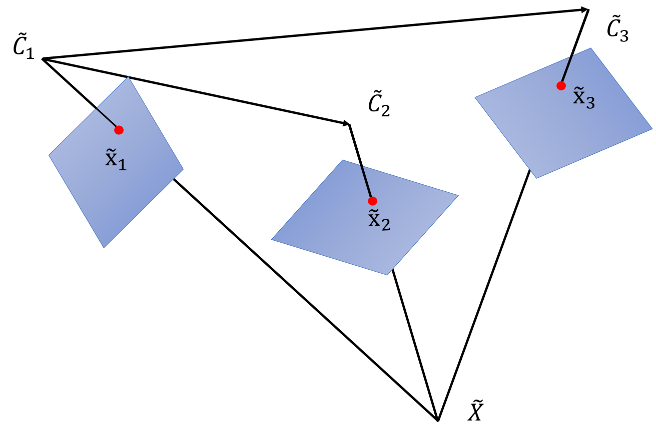


Figure 3.5. The figure of three-frame geometry.

3.3.1 Derivation By Determinant Approach

The matrix determinant approach starts from the situation where three rays intersect in one common object space point $\mathbf{x}_1 \sim I[I|\mathbf{0}]\mathbf{X}$, $\mathbf{x}_2 \sim R_2[I| - \widetilde{\mathbf{C}}_2]\mathbf{X}$ and $\mathbf{x}_3 \sim R_3[I| - \widetilde{\mathbf{C}}_3]\mathbf{X}$. The space point can be represented by the projection elements and relative lengths, $\widetilde{\mathbf{X}} = \lambda_1\mathbf{x}_1 = \widetilde{\mathbf{C}}_2 + \lambda_2 R_2^T\mathbf{x}_2 = \widetilde{\mathbf{C}}_3 + \lambda_3 R_3^T\mathbf{x}_3$.

$$\begin{bmatrix} R_2^T\mathbf{x}_2 & 0 & -\mathbf{x}_1 & \widetilde{\mathbf{C}}_2 \\ 0 & R_3^T\mathbf{x}_3 & -\mathbf{x}_1 & \widetilde{\mathbf{C}}_3 \end{bmatrix} \begin{bmatrix} \lambda_2 \\ \lambda_3 \\ \lambda_1 \\ 1 \end{bmatrix} = \mathbf{0} \tag{3.11}$$

The above equation is noted as $A_{\Lambda 3}\mathbf{\Lambda}_3 = \mathbf{0}$. It is commonly used in the derivations of other researchers. The rank of the 6 by 4 matrix $A_{\Lambda 3}$ is smaller than four. The matrix $A_{\Lambda 3}$ is composed of image points and projection elements only. The homogeneous coordinate vector of the scale elements is in the null space of this matrix. If some of the vectors are unitized, the equations will still hold. The relative scale $\|\widetilde{\mathbf{C}}_3\|/\|\widetilde{\mathbf{C}}_2\| = t_2$ is important, so let $\|\mathbf{x}_1\| = \|R_2^T\mathbf{x}_2\| = \|R_3^T\mathbf{x}_3\| = \|\widetilde{\mathbf{C}}_2\| = 1$ and $\|\widetilde{\mathbf{C}}_3\| = t_2$.

$$\begin{bmatrix} R_2^T\mathbf{x}_2 & 0 & -\mathbf{x}_1 & \widetilde{\mathbf{C}}_2 \\ 0 & R_3^T\mathbf{x}_3 & -\mathbf{x}_1 & \widetilde{\mathbf{C}}_3 \end{bmatrix} \begin{bmatrix} \lambda_2' \\ \lambda_3' \\ \lambda_1' \\ 1 \end{bmatrix} = \mathbf{0} \tag{3.12}$$

The $\lambda's$ represent lengths with the preceding scale assumptions. Since the rank of $A_{\Lambda 3}$ is less than 4, then each choice of 4 rows of the matrix is a square matrix with determinant equal to 0. That is a cumbersome way to proceed and it obscures the projection elements. Multiplying on the left by the transpose of the $A_{\Lambda 3}$ gives a way to deal those elements as a unit. This is the key of the matrix determinant approach. The four by four matrix $A_{\Lambda 3}^T A_{\Lambda 3}$ obtained is as follows.

$$\begin{bmatrix} \mathbf{x}_2^T R_2 R_2^T \mathbf{x}_2 & 0 & -\mathbf{x}_2^T R_2 \mathbf{x}_1 & \mathbf{x}_2^T R_2 \widetilde{\mathbf{C}}_2 \\ 0 & \mathbf{x}_3^T R_3 R_3^T \mathbf{x}_3 & -\mathbf{x}_3^T R_3 \mathbf{x}_1 & \mathbf{x}_3^T R_3 \widetilde{\mathbf{C}}_3 \\ -\mathbf{x}_1^T R_2^T \mathbf{x}_2 & -\mathbf{x}_1^T R_3^T \mathbf{x}_3 & 2\mathbf{x}_1^T \mathbf{x}_1 & -\mathbf{x}_1^T (\widetilde{\mathbf{C}}_2 + \widetilde{\mathbf{C}}_3) \\ \widetilde{\mathbf{C}}_2^T R_2^T \mathbf{x}_2 & \widetilde{\mathbf{C}}_3^T R_3^T \mathbf{x}_3 & -(\widetilde{\mathbf{C}}_2^T + \widetilde{\mathbf{C}}_3^T)\mathbf{x}_1 & \widetilde{\mathbf{C}}_2^T \widetilde{\mathbf{C}}_2 + \widetilde{\mathbf{C}}_3^T \widetilde{\mathbf{C}}_3 \end{bmatrix} \tag{3.13}$$

The determinant of this four by four matrix is zero, which gives the condition of three-frame geometry without needing space points. The components of the matrix are well defined by the space vectors and their intersection angles. These geometric elements are shown in the following figure, and accompanying equations.

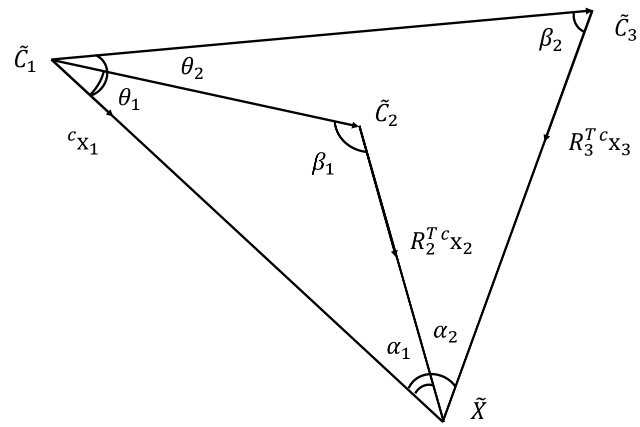


Figure 3.6. The notation of spatial element.

$$\begin{aligned} \mathbf{x}_1^T R_2^T \mathbf{x}_2 &= \cos\alpha_1 & \mathbf{x}_1^T R_3^T \mathbf{x}_3 &= \cos\alpha_2 \\ \widetilde{\mathbf{C}}_2^T R_2^T \mathbf{x}_2 &= \cos(\boldsymbol{\pi} - \beta_1) & \widetilde{\mathbf{C}}_3^T R_3^T \mathbf{x}_3 &= \cos(\boldsymbol{\pi} - \beta_2)\, t_2 \\ \mathbf{x}_1^T \widetilde{\mathbf{C}}_2 &= \cos\theta_1 & \mathbf{x}_1^T \widetilde{\mathbf{C}}_3 &= \cos\theta_2 t_2 \end{aligned}$$

The matrix $M = A_{\Lambda 3}^T A_{\Lambda 3}$ can be partitioned into four 2 by 2 blocks. The determinant of the matrix is computed by the blocks.

$$\det(M) = \det(M_a)\det\left(M_d - M_c M_a^{-1} M_b\right) \tag{3.14}$$

with

$$M = \begin{bmatrix} M_a & M_b \\ M_c & M_d \end{bmatrix}.$$

$\det(M_a) = \left(\mathbf{x}_2^T \mathbf{x}_2\right)\left(\mathbf{x}_3^T \mathbf{x}_3\right) = 1 \neq 0$. So the determinant condition,$\det(M) = 0$, is equivalent to $(M_d - M_c M_a^{-1} M_b) = 0$.

$$M_d - M_c M_a^{-1} M_b = \begin{bmatrix} \sin^2\alpha_1 + \sin^2\alpha_2 & -\sin\alpha_1 \sin\beta_1 - \sin\alpha_2 \sin\beta_2 t_2 \\ -\sin\alpha_1 \sin\beta_1 - \sin\alpha_2 \sin\beta_2 t_2 & \sin^2\beta_1 + \sin^2\beta_2 {t_2}^2 \end{bmatrix}$$

$$\det\left(A_{\Lambda 3}^T A_{\Lambda 3}\right) = \det\left(M_d - M_c M_a^{-1} M_b\right) = (\sin\alpha_2 \sin\beta_1 - t_2 \sin\alpha_1 \sin\beta_2)^2$$

The expression of the determinant of $A_{\Lambda 3}^T A_{\Lambda 3}$ has a quadratic format. The detailed steps of the derivation are in the appendix(A.2). So the determinant condition is equivalent the following equation.

$$\det = 0 \Rightarrow \sin\alpha_2 \sin\beta_1 - t_2 \sin\alpha_1 \sin\beta_2 = 0 \tag{3.15}$$

The $\lambda' s$ can be resolved as below with a value $s > 0$. The constraints of the $\lambda' s > 0$ guarantee that the object point $\widetilde{\mathbf{X}}$ is located in the front of the image planes. This condition is used to check the reconstructed results by many researchers.

$$\begin{aligned}
s &= 2 - \left(\mathbf{x}_2^T R_2 \mathbf{x}_1\right)^2 - \left(\mathbf{x}_3^T R_3 \mathbf{x}_1\right)^2 \\
\mathbf{v}_1 &= \left(\mathbf{x}_2^T R_2 \widetilde{\mathbf{C}}_2\right) R_2^T \mathbf{x}_2 \\
\mathbf{v}_2 &= \left(\mathbf{x}_3^T R_3 \widetilde{\mathbf{C}}_3\right) R_3^T \mathbf{x}_3 \\
\lambda_1' &= \left(\widetilde{\mathbf{C}}_2 + \widetilde{\mathbf{C}}_3 - \mathbf{v}_1 - \mathbf{v}_2\right)^T \mathbf{x}_1 / s \\
\lambda_2' &= \left(\lambda_1' \mathbf{x}_1 - \widetilde{\mathbf{C}}_2\right)^T R_2^T \mathbf{x}_2 \\
\lambda_3' &= \left(\lambda_1' \mathbf{x}_1 - \widetilde{\mathbf{C}}_3\right)^T R_3^T \mathbf{x}_3
\end{aligned} \tag{3.16}$$

The determinant condition (3.15) can also be deduced by a geometric approach using pure geometry only. The geometric analysis yields the condition in a much more concise way.

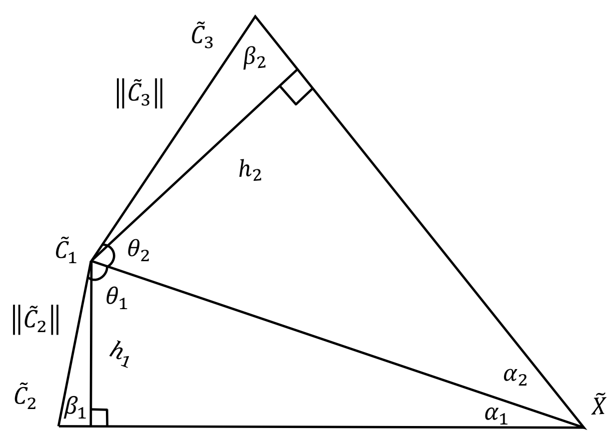


Figure 3.7. The plane plot of the three-frame element.

The two equivalent expressions of $\|\widetilde{\mathbf{C}}_1\widetilde{\mathbf{X}}\|$ give the determinant condition (3.15).

$$
\begin{aligned}
\|\mathbf{h}_1\| &= \|\widetilde{\mathbf{C}}_1\widetilde{\mathbf{X}}\| \sin\alpha_1 = \|\widetilde{\mathbf{C}}_2\| \sin\beta_1 \\
\|\mathbf{h}_2\| &= \|\widetilde{\mathbf{C}}_1\widetilde{\mathbf{X}}\| \sin\alpha_2 = \|\widetilde{\mathbf{C}}_3\| \sin\beta_2 \\
&\Rightarrow \|\widetilde{\mathbf{C}}_1\widetilde{\mathbf{X}}\| = \frac{\|\widetilde{\mathbf{C}}_2\| \sin\beta_1}{\sin\alpha_1} = \frac{\|\widetilde{\mathbf{C}}_3\| \sin\beta_2}{\sin\alpha_2} \\
&\Rightarrow t_2 = \frac{\|\widetilde{\mathbf{C}}_3\|}{\|\widetilde{\mathbf{C}}_2\|} = \frac{\sin\beta_1 \sin\alpha_2}{\sin\alpha_1 \sin\beta 2} \\
&\Rightarrow \sin\beta_1 \sin\alpha_2 - t_2 \sin\alpha_1 \sin\beta_2 = 0
\end{aligned}
$$

The angles can be derived from the cross product of the vectors, which gives the condition expressed by the projection elements.

$$
\begin{aligned}
\sin\alpha_1 &= \frac{\|\mathbf{x}_1 \times R_2^T\mathbf{x}_2\|}{\|\mathbf{x}_1\|\|R_2^T\mathbf{x}_2\|} \quad & \sin\beta_1 &= \frac{\|\widetilde{\mathbf{C}}_2 \times R_2^T\mathbf{x}_2\|}{\|\widetilde{\mathbf{C}}_2\|\|R_2^T\mathbf{x}_2\|} \\
\sin\alpha_2 &= \frac{\|\mathbf{x}_1 \times R_3^T\mathbf{x}_3\|}{\|\mathbf{x}_1\|\|R_3^T\mathbf{x}_3\|} \quad & \sin\beta_2 &= \frac{\|\widetilde{\mathbf{C}}_3 \times R_3^T\mathbf{x}_3\|}{\|\widetilde{\mathbf{C}}_3\|\|R_3^T\mathbf{x}_3\|}
\end{aligned}
$$

The determinant condition (3.15) is equivalent to the following equation, which is also a rearrangement of the equations just above.

$$
\|\mathbf{x}_1 \times R_2^T\mathbf{x}_2\|\|\widetilde{\mathbf{C}}_3 \times R_3^T\mathbf{x}_3\| = \|\widetilde{\mathbf{C}}_2 \times R_2^T\mathbf{x}_2\|\|\mathbf{x}_1 \times R_3^T\mathbf{x}_3\| \tag{3.17}
$$

This equation indicates that the relation does not need the prior length assignments, which were used in this matrix determinant approach. There are four normal vectors in this equation. The normal vectors are introduced in the previous two-frame geometry.

3.3.2 Analysis By Spatial Geometry Approach

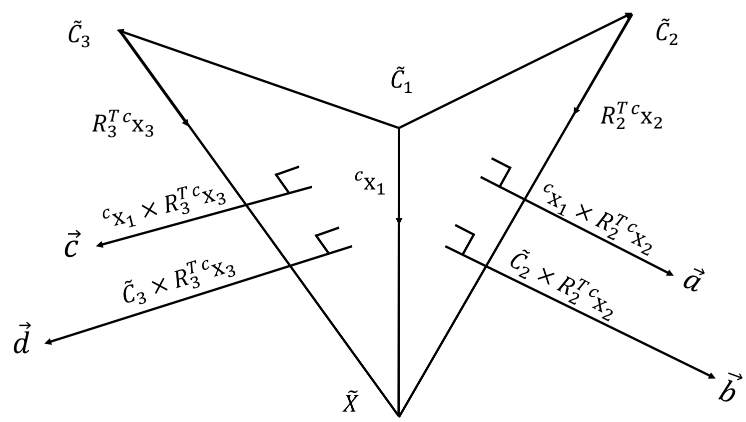


Figure 3.8. The four normal vectors of two planes.

In figure (3.8), the vectors **a** and **b** are the normal vectors of the first $\widetilde{\mathbf{C}}_1\widetilde{\mathbf{C}}_2\widetilde{\mathbf{X}}$ plane. The vectors **c** and **d** are the normal vectors of the second $\widetilde{\mathbf{C}}_1\widetilde{\mathbf{C}}_3\widetilde{\mathbf{X}}$ plane. Therefore vectors **a** and **b** are parallel, the vectors **c** and **d** are parallel. The parallelism of the normal vectors is described by the bilinear equation $\mathcal{B}_{\mathrm{ij}}$. All of them are perpendicular to the vector $\widetilde{\mathbf{C}_1\mathbf{X}} \parallel \mathbf{x}_1$. The vector **a** is $[a_1 \quad a_2 \quad a_3]^T$ for example.

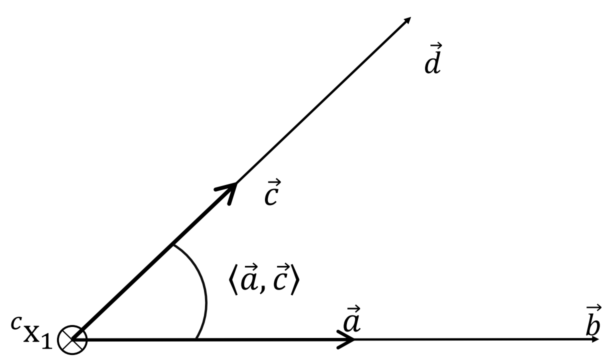


Figure 3.9. The four normal vectors of two planes.

In the figure (3.9), $\mathbf{x}_1$ is perpendicular with the paper. The four normal vectors are perpendicular to the vector $\widetilde{\mathbf{C}}_1\widetilde{\mathbf{X}}$ (or $\parallel \mathbf{x}_1$).

$$\begin{aligned} \mathbf{a} = \mathbf{x}_1 \times R_2^T \mathbf{x}_2 \quad \mathbf{b} = \widetilde{\mathbf{C}}_2 \times R_2^T \mathbf{x}_2 \quad \mathbf{a} \parallel \mathbf{b} \Leftrightarrow \mathcal{B}_{12} \\ \mathbf{c} = \mathbf{x}_1 \times R_3^T \mathbf{x}_3 \quad \mathbf{d} = \widetilde{\mathbf{C}}_3 \times R_3^T \mathbf{x}_3 \quad \mathbf{c} \parallel \mathbf{d} \Leftrightarrow \mathcal{B}_{13} \end{aligned} \tag{3.18}$$

Then equation (3.17) is $\|\mathbf{a}\|\|\mathbf{d}\| = \|\mathbf{b}\|\|\mathbf{c}\|$ in compact form. Considering the parallelism, the two pairs of vectors have a proportionality relationship $\mathbf{a} = \lambda\mathbf{b}, \mathbf{d} = \frac{1}{\lambda}\mathbf{c}$. The relationship guarantees that $\mathbf{a}^T N \mathbf{d} = \mathbf{b}^T N \mathbf{c}$ is true, where N is any three by three matrix .

$$\mathbf{a} = \lambda \mathbf{b} \quad \mathbf{d} = \frac{1}{\lambda}\mathbf{c} \quad \Rightarrow \quad \mathbf{a}^T N \mathbf{d} = \mathbf{b}^T N \mathbf{c} \tag{3.19}$$

The expression $\mathbf{a}^T N \mathbf{d} = \mathbf{b}^T N \mathbf{c}$ is a general equation of the conditions for the four normal vectors. The format of the general equation is identical to the bilinear condition of the point on the second and third images, $\mathbf{x}_2^T E_{23} \mathbf{x}_3 = 0$. This is restated as follows.

$$\mathbf{x}_2^T R_2 \Big([\mathbf{x}_1]_\times N \left[\widetilde{\mathbf{C}}_3\right]_\times - \left[\widetilde{\mathbf{C}}_2\right]_\times N [\mathbf{x}_1]_\times \Big) R_3^T \mathbf{x}_3 = 0 \tag{3.20}$$

This function can be written in the following way.

$$\begin{aligned} &\mathbf{x}_2^T R_2 \left(x_1 H_N^1 + y_1 H_N^2 + z_1 H_N^3 \right) R_3^T \mathbf{x}_3 = 0 \\ &H_N^{\mathrm{i}} = [\mathbf{e}_{\mathrm{i}}]_\times N \left[\widetilde{\mathbf{C}}_3\right]_\times - \left[\widetilde{\mathbf{C}}_2\right]_\times N [\mathbf{e}_{\mathrm{i}}]_\times \end{aligned} \tag{3.21}$$

If the projection matrix of first frame is $\mathbf{x}_1 \sim R_1[I| - \widetilde{\mathbf{C}}_1]\mathbf{X}$, the above equation (3.20) is the one below.

$$\mathbf{x}_2^T R_2 \Big(\left[R_1^T \mathbf{x}_1\right]_\times N \left[\widetilde{\mathbf{C}}_3 - \widetilde{\mathbf{C}}_1\right]_\times - \left[\widetilde{\mathbf{C}}_2 - \widetilde{\mathbf{C}}_1\right]_\times N \left[R_1^T \mathbf{x}_1\right]_\times \Big) R_3^T \mathbf{x}_3 = 0 \tag{3.22}$$

This completes the geometric derivation. The above equation is also the general expression of the trilinear equation. Based on these equivalent derivations, the exploitation and analysis is described in the following section.

There are nine matrices $N_{\mathrm{ij}} = \mathbf{e}_{\mathrm{i}}\mathbf{e}_{\mathrm{j}}^T$ corresponding to the outer products of the three basis vectors. Each of them gives a trilinear equation corresponding to $a_{\mathrm{i}} d_{\mathrm{j}} = b_{\mathrm{i}} c_{\mathrm{j}}$. It is clear that these nine equations are the same as the outer product equations, $\mathbf{a}\mathbf{d}^T = \mathbf{b}\mathbf{c}^T$. The outer product, $\mathbf{a}\mathbf{d}^T = \mathbf{b}\mathbf{c}^T$, is the one below. The full derivation of the outer product(3.23) is in the appendix(A.4.1).

$$R_2^T \left[\mathbf{x}_2\right]_\times R_2 \left(\mathbf{x}_1 \widetilde{\mathbf{C}}_3^T - \widetilde{\mathbf{C}}_2 \mathbf{x}_1^T\right) R_3^T \left[\mathbf{x}_3\right]_\times R_3 = [0]_{3\times 3} \tag{3.23}$$

This expression gives the same equation as the trifocal tensor point-point-point equation in Hartley's(2004)[1] book,

$$\begin{aligned} \left[\mathbf{x}_2\right]_\times R_2 \left(\mathbf{x}_1 \widetilde{\mathbf{C}}_3^T - \widetilde{\mathbf{C}}_2 \mathbf{x}_1^T\right) R_3^T \left[\mathbf{x}_3\right]_\times &= [0]_{3\times 3} \\ \left[\mathbf{x}_2\right]_\times R_2 \left(x_1 H_1 + y_1 H_2 + z_1 H_3\right) R_3^T \left[\mathbf{x}_3\right]_\times &= [0]_{3\times 3} \end{aligned} \tag{3.24}$$

where, $H_{\mathrm{i}} = \mathbf{e}_{\mathrm{i}} \widetilde{\mathbf{C}}_3^T - \widetilde{\mathbf{C}}_2 \mathbf{e}_{\mathrm{i}}^T$. The matrices $H_{\mathrm{i}}, \mathrm{i} = 1, 2, 3$ have a close relation with the trifocal tensor. Using the formulas $T_{\mathrm{i}} = R_2 H_{\mathrm{i}} R_3^T$, equation (3.24) is equivalent to the trifocal tensor as presented in Hartly's(2004)[1] book.

There are three degrees of freedom among the nine equations corresponding to $N_{\mathrm{ij}} \Leftrightarrow a_{\mathrm{i}} d_{\mathrm{j}} = b_{\mathrm{i}} c_{\mathrm{j}}$. Each normal vector, such as $\mathbf{a} = \mathbf{x}_1 \times R_2^T \mathbf{x}_2 = \left[\mathbf{x}_1\right]_\times R_2^T \mathbf{x}_2$, has 2 degrees of freedom. If both $\mathbf{a}$ and $\mathbf{d}$ are not parallel to $\mathbf{e}_3$, each third component of $\mathbf{a}$ and $\mathbf{d}$ is the dependent one. Then $a_1 d_1 = b_1 c_1$, $a_1 d_2 = b_1 c_2$, $a_2 d_1 = b_2 c_1$ and $a_2 b_2 = c_2 d_2$ can represent the other five equations. It is explained in the appendix(A.4.2). Each of these four conditions can be proved by the other three.

$$\frac{(a_1 d_2)(a_2 d_1)}{(a_2 d_2)} = \frac{(b_1 c_2)(b_2 c_1)}{(b_2 c_2)} \Rightarrow a_1 d_1 = b_1 c_1$$

So each three of the four equations will make the remaining one hold. Therefore, there are only three algebraically independent equations among the nine. This correspondes to the three equations which we get by classical photogrametric analysis of a photo triplet. the relative orientations are determined by eleven parameters, and for each point, visible on three photos, we can write three equations : two coplanarity and one scale restraint, Theiss et al.(2000)[123]. For each of the nine equations, $a_{\mathrm{i}} d_{\mathrm{j}} = b_{\mathrm{i}} c_{\mathrm{j}}$, there is a singular configuration which makes the equation trivial. The configuration $\widetilde{\mathbf{C}}_2 \parallel \mathbf{e}_{\mathrm{i}}$ and $\widetilde{\mathbf{C}}_3 \parallel \mathbf{e}_{\mathrm{j}}$ makes the equation

$a_{\mathrm{i}}d_{\mathrm{j}} = b_{\mathrm{i}}c_{\mathrm{j}}$ equivalent to $0 = 0$. The nine equations cannot be used arbitrarily. They should be selected for the reasons above.

Group the nine equations into two groups which are the inner product group $\mathbf{a}^t\mathbf{d} = \mathbf{b}^t\mathbf{c}$ and the cross product group $\mathbf{a} \times \mathbf{d} = \mathbf{b} \times \mathbf{c}$. This is the geometric product of two vectors in geometric algebra, Macdonald(2010)[124]. The nine equations, $a_{\mathrm{i}}d_{\mathrm{j}} = b_{\mathrm{i}}c_{\mathrm{j}}$, are also studied by Indelman(2011)[125]. But He considered the inner product only. The closed form expressions are in the appendix(A.4.3).

The sum of the base equations corresponding to N_{11},N_{22} and N_{33} yields the inner product equation, $\mathbf{a}^T\mathbf{d} = \mathbf{b}^T\mathbf{c}$. When $\mathbf{a}$ is perpendicular to $\mathbf{c}$, the inner product equation is trivial.

$$N_{11} + N_{22} + N_{33} = I \quad \Rightarrow \quad \mathbf{a}^T\mathbf{d} = \mathbf{b}^T\mathbf{c} \tag{3.25}$$

From the figure 3.9, $(\mathbf{a} - \mathbf{c}) \parallel (\mathbf{b} - \mathbf{d}) \Rightarrow (\mathbf{a} - \mathbf{c}) \times (\mathbf{b} - \mathbf{d}) = \mathbf{0}$ is equivalent to $\mathbf{a} \times \mathbf{d} = \mathbf{b} \times \mathbf{c}$. Since the four normal vectors are perpendicular to vector $\widetilde{\mathbf{C}}_1\widetilde{\mathbf{X}}$, one may conclude that $\mathbf{a} \times \mathbf{d} = \mathbf{b} \times \mathbf{c} = -\lambda_0\mathbf{x}_1$ with a scale, λ_0.

The differences between N_{32} and N_{23}, between N_{13} and N_{31}, and between N_{21} and N_{12} generate these cross products. When the four vectors are parallel the cross products are trivial.

$$N_{\mathrm{j}k} - N_{k\mathrm{j}} = [\mathbf{e}_{\mathrm{i}}]_{\times} \Rightarrow \mathbf{a}^T [\mathbf{e}_{\mathrm{i}}]_{\times} \mathbf{d} = \mathbf{b}^T [\mathbf{e}_{\mathrm{i}}]_{\times} \mathbf{c} = -\lambda_0\mathbf{e}_{\mathrm{i}}^T\mathbf{x}_1 \tag{3.26}$$

In the three cross product equations, there are only two linear independent ones. There are still three algebraically independent equations in this grouping analysis.

Some scholars propose their three-frame models instead of using the pure trifocal tensor. They also decompose N-frame geometry into three-frame geometry. Their models are listed in the table below.

Table 3.1. Some three frame models

Author	Model	Remark
Liu	$\mathcal{B}_{\text{ij}}$	
Steffen	$\mathcal{B}_{12}$, $\mathcal{T}_{123}$	line-line-line
Indelman	$\mathcal{B}_{12}$, $\mathcal{B}_{13}$, $\mathcal{T}_{123}$	$\mathbf{a}^T\mathbf{d} = \mathbf{b}^T\mathbf{c}$
Schneider	$\mathcal{B}_{12}$, $\mathcal{B}_{13}$, $\mathcal{T}_{123}$	point-line-line

Liu et al.(2003)[114] uses the $(2n-3)$ $\mathcal{B}$ equations only. Steffen(2010)[116], Indelman(2012)[117], Schneider(2017)[118] propose their mixed models with different $\mathcal{T}_{\text{ij}k}$. The $\mathcal{T}$ of Steffen(2010)[116] are two of the line-line-line equations of the image line, whose closed form expression is not given. His model has one bilinear equation. The $\mathcal{T}$ of Indelman(2012)[117] has a singular case when $\mathbf{a} \perp \mathbf{c}$. His model has two bilinear equations. The $\mathcal{T}$ of Schneider(2017)[118] is the point-line-line relation which needs the two auxiliary image lines. Mayer(2002)[66] studies the method to design the auxiliary image lines efficiently and avoiding singular cases. Heyden and Åström (1996)[77] (1997)[78] indicated that the trilinear equations are indispensable for object points on the trifocal plane. They also show that the necessary number of $\mathcal{B}$ is larger than the redundant equation number $(2n-3)$ if the purely bilinear equation method is used. For instance, for four frames, six $\mathcal{B}$ equations are needed rather than five.

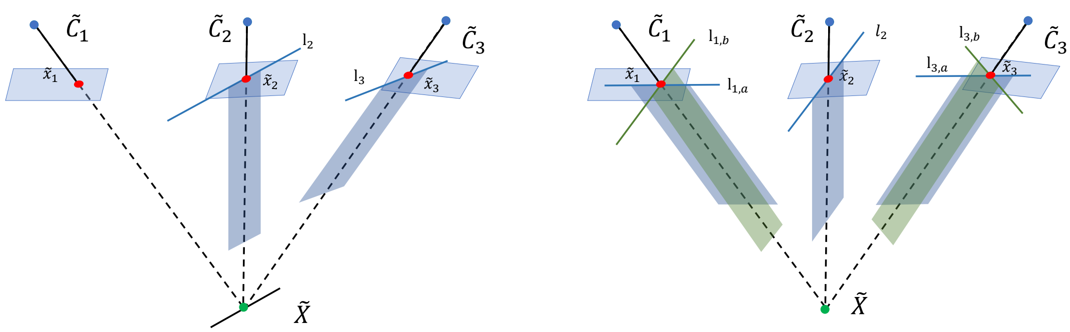


Figure 3.10. Two models: a Schneider's model, and b Steffen's model

Based on the analysis of previous section, a new three-frame model(TFM) is given here. Let $\lambda_{cos} = \cos\langle \mathbf{a}, \mathbf{c} \rangle$.

Table 3.2. The new three-frame model

Equation	Geometry	Condition
$\mathcal{B}_{12}$	$\mathbf{x}_1 \perp \mathbf{a} \parallel \mathbf{b}$	
$\mathcal{B}_{13}$	$\mathbf{x}_1 \perp \mathbf{c} \parallel \mathbf{d}$	
$\mathcal{T}_{123}$	$\mathbf{a}^T \mathbf{d} = \mathbf{b}^T \mathbf{c}$	$\|\lambda_{cos}\| \geq \frac{\sqrt{2}}{2}$
	$\mathbf{a}^T \left[\mathbf{e}_{\mathrm{i}}\right]_\times \mathbf{d} = \mathbf{b}^T \left[\mathbf{e}_{\mathrm{i}}\right]_\times \mathbf{c}$	$\|\lambda_{cos}\| < \frac{\sqrt{2}}{2}$

It is abbreviated as shown below.

$$TFM(1,2,3) = \begin{cases} \mathcal{B}_{12} \\ \mathcal{B}_{13} \\ \mathcal{T}_{123} \end{cases}$$

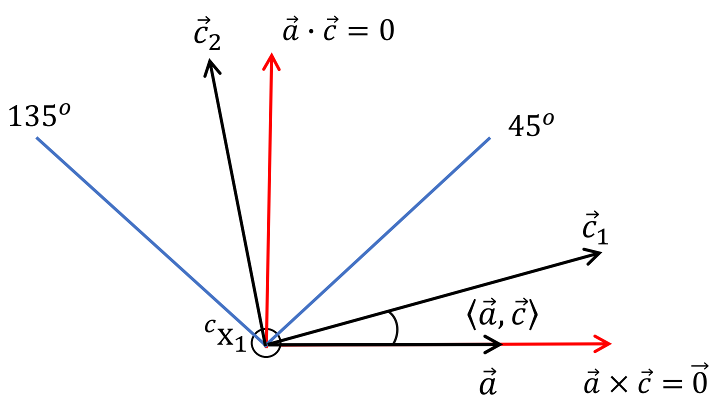


Figure 3.11. The singularity of the trilineary equation.

This figure(3.11) shows the two singularities of the trilinear equation for the inner product and the cross product. When the vector angle is small, the inner product is stable. When the vector angle is large, the cross production is stable. Selection of the trilinear equat depends on the vector angle between $\mathbf{a}$ and $\mathbf{c}$. The upper and lower thresholds are 45 and 135 degrees here.

The new model covers the singular case of $\mathbf{a}^T\mathbf{d} = \mathbf{b}^T\mathbf{c}$ and avoids the extra auxiliary image lines. $\mathcal{B}$ expresses the parallelism of the normal vectors. $\mathcal{T}$ ensures $\|\mathbf{a}\|\|\mathbf{d}\| = \|\mathbf{b}\|\|\mathbf{c}\|$. Considering $\mathbf{a} \times \mathbf{d} = \mathbf{b} \times \mathbf{c} = -\lambda_0 \mathbf{x}_1$, and if the $\mathrm{i}-th$ component of $\mathbf{x}_1$ is nonzero, the cross product $\mathbf{a} \times \mathbf{d} = \mathbf{b} \times \mathbf{c}$ can be represented by $\mathbf{a}^T \left[\mathbf{e}_\mathrm{i}\right]_\times \mathbf{d} = \mathbf{b}^T \left[\mathbf{e}_\mathrm{i}\right]_\times \mathbf{c}$. If the $\mathrm{i}-th$ component of $\mathbf{x}_1$ is the maximal absolute value, the $\mathrm{i}-th$ equation of the cross product is recommended. So this model can be implemented on an image point at infinity whose third component is zero.

In this section, a new derivation and analysis of the trilinear equations are given. Based on the analysis, a new model is restated. The new model overcomes the singular case and avoids the auxiliary image lines.

3.3.3 Analysis By Column Vector Approach

One object point gives six equations for the three frames $P_1 = I[I|\mathbf{0}]$, $P_2 = R_2[I| - \widetilde{\mathbf{C}}_2]$ and $P_3 = R_3[I| - \widetilde{\mathbf{C}}_3]$. So the number of redundant equations is three. Using the simplified projection (3.5) and (3.4), the six equations are below in the stacked matrix format.

$$\begin{bmatrix} 1 & 0 & m_1 & 0 \\ 0 & 1 & p_1 & 0 \\ 1 & 0 & m_2 & n_2 \\ 0 & 1 & p_2 & q_2 \\ 1 & 0 & m_3 & n_3 \\ 0 & 1 & p_3 & q_3 \end{bmatrix} \mathbf{X} = \mathbf{0} \tag{3.27}$$

The above equation is noted as $A_{s3}\mathbf{X} = \mathbf{0}$. Gaussian elimination gives the equation below as the shown in prior section.

$$\begin{bmatrix} 1 & 0 & m_1 & 0 \\ 0 & 1 & p_1 & 0 \\ \mathbf{0} & \mathbf{0} & \mathbf{U}_3 & \mathbf{V}_3 \end{bmatrix} \mathbf{X} = \mathbf{0} \tag{3.28}$$

$$\text{where } \begin{bmatrix} \mathbf{U}_3 & \mathbf{V}_3 \end{bmatrix} = \begin{bmatrix} m_2 - m_1 & n_2 \\ p_2 - p_1 & q_2 \\ m_3 - m_1 & n_3 \\ p_3 - p_1 & q_3 \end{bmatrix} \text{ and } \mathbf{U}_3 \parallel \mathbf{V}_3$$

There are two parallel vectors, $\mathbf{U}_3$ and $\mathbf{V}_3$, in the lower right corner of the matrix in (3.28). The rank of the stacked matrix is less than four making the $\mathbf{U}_3 \parallel \mathbf{V}_3$. The first and the last two rows, noted as $row_{1,2}$ and $row_{3,4}$, of these vectors give two bilinear equations $\mathcal{B}_{12}$ and $\mathcal{B}_{13}$ referring to (3.10). The first and the third row,noted as $row_{1,3}$, give the equation involving three frames. And there are four ways to generate this expression, for instance $row_{1,3}$, $row_{1,4}$, $row_{2,3}$ and $row_{2,4}$.

$$\begin{aligned} &(1)\ row_{1,3} \rightarrow \frac{m_2 - m_1}{m_3 - m_1} = \frac{n_2}{n_3} \\ &(2)\ row_{1,4} \rightarrow \frac{m_2 - m_1}{p_3 - p_1} = \frac{n_2}{q_3} \\ &(3)\ row_{2,3} \rightarrow \frac{p_2 - p_1}{m_3 - m_1} = \frac{q_2}{n_3} \\ &(4)\ row_{2,4} \rightarrow \frac{p_2 - p_1}{p_3 - p_1} = \frac{q_2}{q_3} \end{aligned} \tag{3.29}$$

Selecting two rows $row_{2n-1,2n}$ and one of $row_{2m-1,2m}$ gives two three-dimensional vectors noted as the $\mathbf{u}$ and $\mathbf{v}$. For instance, selecting the row one, two, and three noted as $row_{1,2,3}$ gives two vectors.

$$\begin{bmatrix} \mathbf{u} & \mathbf{v} \end{bmatrix} = \begin{bmatrix} m_2 - m_1 & n_2 \\ p_2 - p_1 & q_2 \\ m_3 - m_1 & n_3 \end{bmatrix} \text{ and } \mathbf{u} \parallel \mathbf{v}$$

The cross product of $\mathbf{v}$ and $\mathbf{u}$ is $[\mathbf{v}]_\times \mathbf{u} = \mathbf{0}$.

$$\begin{bmatrix} 0 & -n_3 & q_2 \\ n_3 & 0 & -n_2 \\ -q_2 & n_2 & 0 \end{bmatrix} \begin{bmatrix} m_2 - m_1 \\ p_2 - p_1 \\ m_3 - m_1 \end{bmatrix} = \begin{bmatrix} -n_3(p_2 - p_1) + q_2(m_3 - m_1) \\ n_3(m_2 - m_1) - n_2(m_3 - m_1) \\ q_2(m_2 - m_1) + n_2(p_2 - p_1) \end{bmatrix} = \mathbf{0} \Leftrightarrow \begin{bmatrix} \mathcal{T}_{123} \text{ of } row_{2,3} \\ \mathcal{T}_{123} \text{ of } row_{1,3} \\ \mathcal{B}_{12} \end{bmatrix}$$

The above relation indicates the dependency between $\mathcal{T}_{123}$ and $\mathcal{B}_{12}$. Given these bilinear equations, there is only one unknown scalar in the three-frame geometry, referring to (3.18) and (3.19). It is proven to be the minimal and sufficient condition of three-frame geometry. Faugeras and Mourrain(1995)[75] show that, given two bilinear equations, there is only one algebraically independent trilinear equation.

The $\mathbf{U}_3 \parallel \mathbf{V}_3$ gives two bilinear equations $\mathcal{B}_{12}$ and $\mathcal{B}_{13}$, and four independent $\mathcal{T}_{123}$. Although there are nine trilinear equations in Hartly's trifocal tensor expression, only four of them are linearly independent but algebraically dependent. It is proved by Werman and Shashua(1995)[9]. Hartley(1995)[74] suggests to choose four equations by a so-called householder matrix method. Also he shows that utilizing the whole equations together gives the best numerical performance. The spatial geometry motivates these four trilinear equations. One equation comes from the inner product, and three equations come from the cross product.

So in the nine equations(3.21), there are only three algebraically independent equations. They are represented by $\mathcal{B}_{12}$, $\mathcal{B}_{13}$ and one $\mathcal{T}_{123}$, which depends on the spatial configuration.

This column vector approach consolidates the former analysis and the three-frame model derivation, and restates the new proposed model.

3.4 Four-Frame Geometry

This section has the same structure as the last one. A simplified derivation and analyses of four-frame geometry are given here. It shows that the bilinear and trilinear equations can guarantee the four-frame geometry. The three-frame geometry is exploited here under the four-frame geometry situation. Utilizing the previous conclusions simplifies the derivation and analysis. Some details of the derivation are in the appendix (A.3).

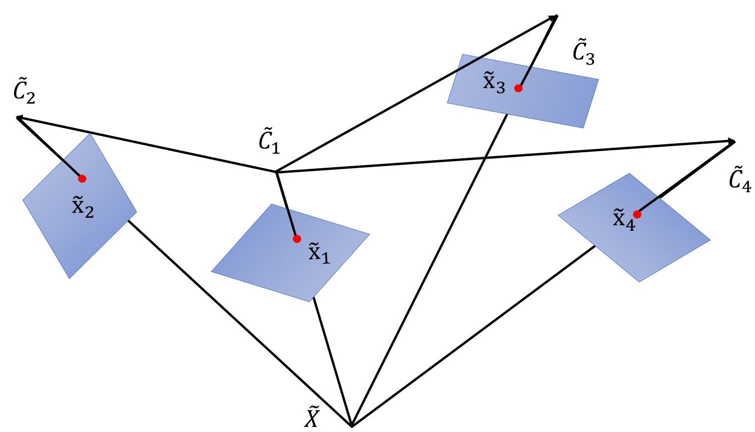


Figure 3.12. The figure of four-frame geometry.

3.4.1 Derivation By Determinant Approach

The matrix determinant approach starts from the situation where four rays intersect in one common object space point $\mathbf{x}_1 \sim I[I|\mathbf{0}]\mathbf{X}$, $\mathbf{x}_2 \sim R_2[I| - \widetilde{\mathbf{C}}_2]\mathbf{X}$, $\mathbf{x}_3 \sim R_3[I| - \widetilde{\mathbf{C}}_3]\mathbf{X}$, and $\mathbf{x}_4 \sim R_4[I| - \widetilde{\mathbf{C}}_4]\mathbf{X}$. The space point can be represented by the projection elements and relative lengths, $\widetilde{\mathbf{X}} = \lambda_1\mathbf{x}_1 = \widetilde{\mathbf{C}}_2 + \lambda_2 R_2^T\mathbf{x}_2 = \widetilde{\mathbf{C}}_3 + \lambda_3 R_3^T\mathbf{x}_3 = \widetilde{\mathbf{C}}_4 + \lambda_4 R_4^T\mathbf{x}_4$.

$$\begin{bmatrix} R_2^T\mathbf{x}_2 & 0 & 0 & -\mathbf{x}_1 & \widetilde{\mathbf{C}}_2 \\ 0 & R_3^T\mathbf{x}_3 & 0 & -\mathbf{x}_1 & \widetilde{\mathbf{C}}_3 \\ 0 & 0 & R_4^T\mathbf{x}_4 & -\mathbf{x}_1 & \widetilde{\mathbf{C}}_4 \end{bmatrix} \begin{bmatrix} \lambda_2 \\ \lambda_3 \\ \lambda_4 \\ \lambda_1 \\ 1 \end{bmatrix} = \mathbf{0} \tag{3.30}$$

The above equation is noted as $A_{\Lambda 4}\mathbf{\Lambda}_4 = \mathbf{0}$. The rank of the nine by five matrix $A_{\Lambda 4}$ is smaller than five. The homogeneous coordinate vector of the scale elements is in the null space of this matrix. It gives the below equation.

$$A_{\Lambda 4}^T A_{\Lambda 4}\mathbf{\Lambda}_4 = \mathbf{0} \tag{3.31}$$

The determinant of the $A_{\Lambda 4}^T A_{\Lambda 4}$ is zero, which gives the condition of four-frame geometry without needing space points. The relative scale $\|\widetilde{\mathbf{C}}_3\|/\|\widetilde{\mathbf{C}}_2\| = t_2$ and $\|\widetilde{\mathbf{C}}_4\|/\|\widetilde{\mathbf{C}}_2\| = t_3$ are important, so let $\|\mathbf{x}_1\| = \|R_2^T\mathbf{x}_2\| = \|R_3^T\mathbf{x}_3\| = \|\widetilde{\mathbf{C}}_2\| = 1$, $\|\widetilde{\mathbf{C}}_3\| = t_2$,and $\|\widetilde{\mathbf{C}}_4\| = t_3$. The expression of $A_{\Lambda 4}^T A_{\Lambda 4}$ is defined by the space vectors and their intersection angles. These geometric elements are shown in the following figure, and accompanying equations.

$$\begin{array}{ccc}
\mathbf{x}_1^T R_2^T \mathbf{x}_2 = \cos\alpha_1 & \mathbf{x}_1^T R_3^T \mathbf{x}_3 = \cos\alpha_2 & \mathbf{x}_1^T R_4^T \mathbf{x}_4 = \cos\alpha_3 \\
\widetilde{\mathbf{c}}_2^T R_2^T \mathbf{x}_2 = \cos(\boldsymbol{\pi} - \beta_1) & \widetilde{\mathbf{c}}_3^T R_3^T \mathbf{x}_3 = \cos(\boldsymbol{\pi} - \beta_2)\, t_2 & \widetilde{\mathbf{c}}_4^T R_4^T \mathbf{x}_4 = \cos(\boldsymbol{\pi} - \beta_3)\, t_3 \\
\mathbf{x}_1^T \widetilde{\mathbf{C}}_2 = \cos\theta_1 & \mathbf{x}_1^T \widetilde{\mathbf{C}}_3 = \cos\theta_2 t_2 & \mathbf{x}_1^T \widetilde{\mathbf{C}}_4 = \cos\theta_3 t_3
\end{array}$$

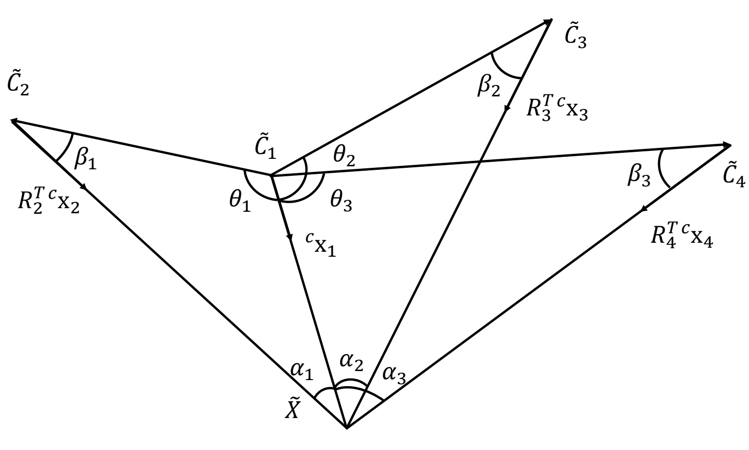


Figure 3.13. The vectors of three planes.

The matrix $M = A_{\Lambda 4}^T A_{\Lambda 4}$ can be partitioned into four blocks. The determinant of the matrix is computed by the blocks.

$$\det(M) = \det(M_a)\det\left(M_d - M_c M_a^{-1} M_b\right) \tag{3.32}$$

with

$$M = \begin{bmatrix} M_a & M_b \\ M_c & M_d \end{bmatrix}.$$

M_a is a three-by-three matrix in the upper left corner. M_d is a two-by-two matrix in the lower right corner. $\det(M_a) = \left(\mathbf{x}_2^T \mathbf{x}_2\right)\left(\mathbf{x}_3^T \mathbf{x}_3\right) = 1 \neq 0$. So the determinant condition, $\det(M) = 0$, is equivalent to $(M_d - M_c M_a^{-1} M_b) = 0$.

$$\begin{aligned}\det\left(A_{\Lambda 4}^T A_{\Lambda 4}\right) = \det\left(M_d - M_c M_a^{-1} M_b\right) &= (\sin\alpha_2 \sin\beta_1 - t_2 \sin\alpha_1 \sin\beta_2)^2 \\ &+ (\sin\alpha_3 \sin\beta_1 - t_3 \sin\alpha_1 \sin\beta_3)^2 \\ &+ (t_2 \sin\alpha_3 \sin\beta_2 - t_3 \sin\alpha_2 \sin\beta_3)^2\end{aligned}$$

The expression of the determinant of $A_{\Lambda 4}^T A_{\Lambda 4}$ has a quadratic format. The detailed steps of the derivation are in the appendix (A.3). So the determinant condition is equivalent to the three following equations.

$$\sin\alpha_2 \sin\beta_1 - t_2 \sin\alpha_1 \sin\beta_2 = 0 \tag{3.33a}$$

$$\sin\alpha_3 \sin\beta_1 - t_3 \sin\alpha_1 \sin\beta_3 = 0 \tag{3.33b}$$

$$t_2 \sin\alpha_3 \sin\beta_2 - t_3 \sin\alpha_2 \sin\beta_3 = 0 \tag{3.33c}$$

The angles can be derived from the cross product of the vectors, which gives the condition expressed by projection elements.

$$\sin\alpha_1 = \frac{\|\mathbf{x}_1 \times R_2^T\mathbf{x}_2\|}{\|\mathbf{x}_1\|\|R_2^T\mathbf{x}_2\|} \quad \sin\beta_1 = \frac{\|\widetilde{\mathbf{C}}_2 \times R_2^T\mathbf{x}_2\|}{\|\widetilde{\mathbf{C}}_2\|\|R_2^T\mathbf{x}_2\|}$$

$$\sin\alpha_2 = \frac{\|\mathbf{x}_1 \times R_3^T\mathbf{x}_3\|}{\|\mathbf{x}_1\|\|R_3^T\mathbf{x}_3\|} \quad \sin\beta_2 = \frac{\|\widetilde{\mathbf{C}}_3 \times R_3^T\mathbf{x}_3\|}{\|\widetilde{\mathbf{C}}_3\|\|R_3^T\mathbf{x}_3\|}$$

$$\sin\alpha_3 = \frac{\|\mathbf{x}_1 \times R_4^T\mathbf{x}_4\|}{\|\mathbf{x}_1\|\|R_4^T\mathbf{x}_4\|} \quad \sin\beta_3 = \frac{\|\widetilde{\mathbf{C}}_4 \times R_4^T\mathbf{x}_4\|}{\|\widetilde{\mathbf{C}}_4\|\|R_4^T\mathbf{x}_4\|}$$

The determinant condition (3.33) is equivalent to the following equations. These equations indicate that the relations do not need the prior length assignments,$\|\mathbf{x}_1\| = \|R_2^T\mathbf{x}_2\| = \|R_3^T\mathbf{x}_3\| = \|\widetilde{\mathbf{C}}_2\| = 1$.

$$3.33a \Rightarrow \|\mathbf{x}_1 \times R_2^T\mathbf{x}_2\|\|\widetilde{\mathbf{C}}_3 \times R_3^T\mathbf{x}_3\| = \|\widetilde{\mathbf{C}}_2 \times R_2^T\mathbf{x}_2\|\|\mathbf{x}_1 \times R_3^T\mathbf{x}_3\| \tag{3.34a}$$

$$3.33b \Rightarrow \|\mathbf{x}_1 \times R_2^T\mathbf{x}_2\|\|\widetilde{\mathbf{C}}_4 \times R_4^T\mathbf{x}_4\| = \|\widetilde{\mathbf{C}}_2 \times R_2^T\mathbf{x}_2\|\|\mathbf{x}_1 \times R_4^T\mathbf{x}_4\| \tag{3.34b}$$

$$3.33c \Rightarrow \|\mathbf{x}_1 \times R_3^T\mathbf{x}_3\|\|\widetilde{\mathbf{C}}_4 \times R_4^T\mathbf{x}_4\| = \|\widetilde{\mathbf{C}}_3 \times R_3^T\mathbf{x}_3\|\|\mathbf{x}_1 \times R_4^T\mathbf{x}_4\| \tag{3.34c}$$

Dividing the two sides of the two formulas,(3.34a) and (3.34b), yields the third formula (3.34c).

$$\frac{\|\mathbf{x}_1 \times R_2^T\mathbf{x}_2\|\|\widetilde{\mathbf{C}}_3 \times R_3^T\mathbf{x}_3\|}{\|\mathbf{x}_1 \times R_2^T\mathbf{x}_2\|\|\widetilde{\mathbf{C}}_4 \times R_4^T\mathbf{x}_4\|} = \frac{\|\widetilde{\mathbf{C}}_2 \times R_2^T\mathbf{x}_2\|\|\mathbf{x}_1 \times R_3^T\mathbf{x}_3\|}{\|\widetilde{\mathbf{C}}_2 \times R_2^T\mathbf{x}_2\|\|\mathbf{x}_1 \times R_4^T\mathbf{x}_4\|}$$

$$\Rightarrow$$

$$\|\mathbf{x}_1 \times R_3^T\mathbf{x}_3\|\|\widetilde{\mathbf{C}}_4 \times R_4^T\mathbf{x}_4\| = \|\widetilde{\mathbf{C}}_3 \times R_3^T\mathbf{x}_3\|\|\mathbf{x}_1 \times R_4^T\mathbf{x}_4\|$$

So each two of the three equations make the remaining one hold. The first two equations give two three-frame geometry models for the four-frame case. This derivation provides no quadrilinear equations. So the bilinear and the trilinear equations guarantee the four-frame cases. The relation of the two models and the searching for quadrilinear equations are in the following section.

3.4.2 Analysis By Spatial Geometry Approach

In this section the analysis is provided in the same way as in the previous section.

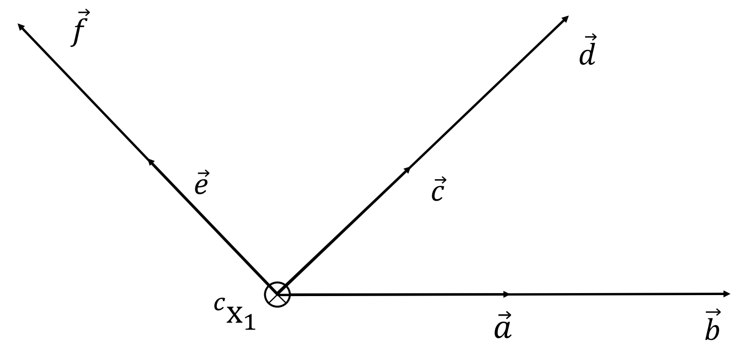


Figure 3.14. The six normal vectors of three frame planes.

Compared to the three-frame geometry, there are six normal vectors in the equations (3.34a) and (3.34b) shown in the figure (3.14). All of the six normal vectors are perpendicular to the vector $\widetilde{\mathbf{C}}_1\widetilde{\mathbf{X}} \parallel \mathbf{x}_1$. The two new ones are noted as $\mathbf{e}$ and $\mathbf{f}$. They are the normal vectors of the third $\widetilde{\mathbf{C}}_1\widetilde{\mathbf{C}}_4\widetilde{\mathbf{X}}$ plane. Then equation (3.34a) and (3.34b) are $\|\mathbf{a}\|\|\mathbf{d}\| = \|\mathbf{b}\|\|\mathbf{c}\|$ and $\|\mathbf{a}\|\|\mathbf{f}\| = \|\mathbf{b}\|\|\mathbf{e}\|$ in compact form. The combination of the two proportion relations is given here.

$$\frac{\|\mathbf{a}\|}{\|\mathbf{b}\|} = \frac{\|\mathbf{c}\|}{\|\mathbf{d}\|} = \frac{\|\mathbf{e}\|}{\|\mathbf{f}\|} = \lambda$$

The unified relation of the proportion and parallelism is provided in the equations below.

$$\begin{aligned} \mathbf{a} &= \lambda\mathbf{b} & \mathbf{x}_1^T\mathbf{b} &= 0 \\ \mathbf{c} &= \lambda\mathbf{d} & \mathbf{x}_1^T\mathbf{d} &= 0 \\ \mathbf{e} &= \lambda\mathbf{f} & \mathbf{x}_1^T\mathbf{f} &= 0 \end{aligned} \tag{3.35}$$

Each of (3.34a) and (3.34b) gives two sets of trilinear equations including inner products and cross products. One set has three algebraically independent equations. But there are only five independent equations in the four-frame geometry. The analysis here is to answer the question of whether the potential quadrilinear equations are consistent with that. The two inner product equations are $\mathbf{a}^T\mathbf{d} = \mathbf{b}^T\mathbf{c}$ and $\mathbf{a}^T\mathbf{f} = \mathbf{b}^T\mathbf{e}$, which make $(\mathbf{a}^T\mathbf{d})(\mathbf{b}^T\mathbf{e}) = (\mathbf{a}^T\mathbf{f})(\mathbf{b}^T\mathbf{c})$. The two cross product equations are $\mathbf{d}\times\mathbf{a} = \mathbf{c}\times\mathbf{b}$ and $\mathbf{f}\times\mathbf{a} = \mathbf{e}\times\mathbf{b}$, which make $(\mathbf{d}\times\mathbf{a})^T(\mathbf{e}\times\mathbf{b}) = (\mathbf{f}\times\mathbf{a})^T(\mathbf{c}\times\mathbf{b})$.

$$
\begin{aligned}
(\mathbf{a}^T\mathbf{d})(\mathbf{b}^T\mathbf{e}) = (\mathbf{a}^T\mathbf{f})(\mathbf{b}^T\mathbf{c}) \quad &\Rightarrow \mathbf{a}^T(\mathbf{f}\mathbf{c}^T - \mathbf{c}\mathbf{f}^T)\mathbf{b} = 0 \\
(\mathbf{d}\times\mathbf{a})^T(\mathbf{e}\times\mathbf{b}) = (\mathbf{f}\times\mathbf{a})^T(\mathbf{c}\times\mathbf{b}) &\Rightarrow \mathbf{a}^T([\mathbf{f}]_\times[\mathbf{c}]_\times - [\mathbf{c}]_\times[\mathbf{f}]_\times)\mathbf{b} = 0 \\
\text{and}\ \ \mathbf{a}^T(\mathbf{f}\mathbf{c}^T - \mathbf{c}\mathbf{f}^T)\mathbf{b} = \mathbf{a}^T([\mathbf{f}]_\times[\mathbf{c}]_\times - [\mathbf{c}]_\times[\mathbf{f}]_\times)\mathbf{b}
\end{aligned}
$$

The above derivation shows the dependency of the two sets. The $(\mathbf{a}^T\mathbf{d})(\mathbf{b}^T\mathbf{e}) = (\mathbf{a}^T\mathbf{f})(\mathbf{b}^T\mathbf{c})$ and $(\mathbf{d}\times\mathbf{a})^T(\mathbf{e}\times\mathbf{b}) = (\mathbf{f}\times\mathbf{a})^T(\mathbf{c}\times\mathbf{b})$ are equivalent. No quadrilinear equation is found up to now. This is revisited in the next analysis approach.

Given three bilinear equations $\mathcal{B}_{12}$, $\mathcal{B}_{13}$, and $\mathcal{B}_{14}$, there are two unknown relative distances $\|\widetilde{\mathbf{C}}_3\|/\|\widetilde{\mathbf{C}}_2\| = t_2$ and $\|\widetilde{\mathbf{C}}_4\|/\|\widetilde{\mathbf{C}}_2\| = t_3$. The t_3 can be replaced by $\|\widetilde{\mathbf{C}}_4\|/\|\widetilde{\mathbf{C}}_3\|$, because $t_3 = (\|\widetilde{\mathbf{C}}_4\|/\|\widetilde{\mathbf{C}}_3\|)t_2$. So two trilinear equations, $\mathcal{T}_{123}$ and $\mathcal{T}_{134}$ are required. The total necessary equations are three bilinear equations and two trilinear equations.

Table 3.3. The TFM for four-frame geometry

Equation	Geometry
$\mathcal{B}_{12}$	$\mathbf{x}_1 \perp \mathbf{a} \parallel \mathbf{b}$
$\mathcal{B}_{13}$	$\mathbf{x}_1 \perp \mathbf{c} \parallel \mathbf{d}$
$\mathcal{B}_{14}$	$\mathbf{x}_1 \perp \mathbf{e} \parallel \mathbf{f}$
$\mathcal{T}_{123}$	$\|\mathbf{a}\|\|\mathbf{d}\| = \|\mathbf{b}\|\|\mathbf{c}\|$
$\mathcal{T}_{134}$	$\|\mathbf{c}\|\|\mathbf{f}\| = \|\mathbf{d}\|\|\mathbf{e}\|$

3.4.3 Analysis By Column Vector Approach

The four camera matrices are $P_1 = I[I|\mathbf{0}]$, $P_{\rm i} = R_{\rm i}[I| - \widetilde{\mathbf{C}}_{\rm i}]$. One object point gives eight equations for these four frames. So the number of redundant equations is five. Using the simplified projection (3.5) and (3.4), the eight equations are shown in the stacked matrix format.

$$\begin{bmatrix} 1 & 0 & m_1 & 0 \\ 0 & 1 & p_1 & 0 \\ 1 & 0 & m_2 & n_2 \\ 0 & 1 & p_2 & q_2 \\ 1 & 0 & m_3 & n_3 \\ 0 & 1 & p_3 & q_3 \\ 1 & 0 & m_4 & n_4 \\ 0 & 1 & p_4 & q_4 \end{bmatrix} \mathbf{X} = \mathbf{0} \tag{3.37}$$

The above equation is noted as $A_{s4}\mathbf{X} = \mathbf{0}$. Gaussian elimination gives the equation below as the shown in the prior section.

$$\begin{bmatrix} 1 & 0 & m_1 & 0 \\ 0 & 1 & p_1 & 0 \\ \mathbf{0} & \mathbf{0} & \mathbf{U}_4 & \mathbf{V}_4 \end{bmatrix} \mathbf{X} = \mathbf{0} \tag{3.38}$$

$$\text{where } \begin{bmatrix} \mathbf{U}_4 & \mathbf{V}_4 \end{bmatrix} = \begin{bmatrix} m_2 - m_1 & n_2 \\ p_2 - p_1 & q_2 \\ m_3 - m_1 & n_3 \\ p_3 - p_1 & q_3 \\ m_4 - m_1 & n_4 \\ p_4 - p_1 & q_4 \end{bmatrix} \text{ and } \mathbf{U}_4 \parallel \mathbf{V}_4$$

There are two parallel vectors, $\mathbf{U}_4$ and $\mathbf{V}_4$, in the lower right corner of the matrix in (3.38). The rank of the stacked matrix is less than four making the $\mathbf{U}_4 \parallel \mathbf{V}_4$. The two rows marked as $row_{2n-1,2n}$ give one bilinear equation, such as $\mathcal{B}_{12}$, $\mathcal{B}_{13}$ and $\mathcal{B}_{14}$ referring to (3.10). The four rows marked as $row_{2n-1,2n}$ and $row_{2m-1,2m}$, give the trilinear equations, such as $\mathcal{T}_{123}$, $\mathcal{T}_{124}$ and $\mathcal{T}_{134}$ referring to (3.29). The proportion relations can give the $\mathcal{B}$ and $\mathcal{T}$ without the $\mathcal{Q}$. (i.e. without an explicit quadrilinear equation)

Selecting one row from each of $row_{1,2}$, $row_{3,4}$, and $row_{5,6}$ gives two three-dimensional vectors noted as the $\mathbf{u}$ and $\mathbf{v}$. For instance, selecting the row one, three, and five noted as $row_{1,3,5}$ gives the two vectors below.

$$\begin{bmatrix} \mathbf{u} & \mathbf{v} \end{bmatrix} = \begin{bmatrix} m_2 - m_1 & n_2 \\ m_3 - m_1 & n_3 \\ m_4 - m_1 & n_4 \end{bmatrix} \text{ and } \mathbf{u} \parallel \mathbf{v}$$

The cross product of $\mathbf{v}$ and $\mathbf{u}$ is $[\mathbf{v}]_\times \mathbf{u} = \mathbf{0}$.

$$\begin{bmatrix} 0 & -n_4 & n_3 \\ n_4 & 0 & -n_2 \\ -n_3 & n_2 & 0 \end{bmatrix} \begin{bmatrix} m_2 - m_1 \\ m_3 - m_1 \\ m_4 - m_1 \end{bmatrix} = \begin{bmatrix} -n_4(m_3 - m_1) + n_3(m_4 - m_1) \\ n_4(m_2 - m_1) - n_2(m_4 - m_1) \\ n_3(m_2 - m_1) + n_2(m_3 - m_1) \end{bmatrix} = \mathbf{0} \Leftrightarrow \begin{bmatrix} \mathcal{T}_{134} \\ \mathcal{T}_{124} \\ \mathcal{T}_{123} \end{bmatrix}$$

The above relation indicates the dependency among three equations $\mathcal{T}_{123}$, $\mathcal{T}_{124}$, and $\mathcal{T}_{134}$, with two of them being independent.

A linear transformation of $\mathbf{u}$, $\mathbf{v}$ provides the $\mathbf{u}'$, $\mathbf{v}'$.

$$\begin{bmatrix} \mathbf{u}' & \mathbf{v}' \end{bmatrix} = \begin{bmatrix} m_2 - m_1 & n_2 \\ m_3 - m_2 & n_3 - n_2 \\ m_4 - m_3 & n_4 - n_3 \end{bmatrix} \text{ and } \mathbf{u}' \parallel \mathbf{v}'$$

The cross product of $\mathbf{v}'$ and $\mathbf{u}'$ is

$$\begin{bmatrix} 0 & -(n_4 - n_3) & n_3 - n_2 \\ n_4 - n_3 & 0 & -n_2 \\ -(n_3 - n_2) & n_2 & 0 \end{bmatrix} \begin{bmatrix} m_2 - m_1 \\ m_3 - m_2 \\ m_4 - m_3 \end{bmatrix} = \mathbf{0} \Leftrightarrow \begin{bmatrix} \mathcal{T}_{134} \\ \mathcal{Q}_{1234} \\ \mathcal{T}_{123} \end{bmatrix}.$$

There is a quadrilinear equation in the above expression. Considering a linear transformation relation, $[M\mathbf{v}]_\times M\mathbf{u} = det(M)M^{-T}([\mathbf{v}]_\times \mathbf{u})$, that equation is the linear transformation of trilinear equations.

Each linear transformation of $[\mathbf{v}]_\times \mathbf{u}$ has two linearly independent equations. There are eight combinations of $row_{\mathrm{i,j},k}$. So there are sixteen equations involving four-frame parameters. It illustrates the dependency relationships between the $\mathcal{T}$ and $\mathcal{Q}$. But the closed form expressions are obscure. Trigg(1995)[71] and Heyden(1995)[72] (1997)[73] conjecture that there are transformations among $\mathcal{B}$, $\mathcal{T}$ and $\mathcal{Q}$. Faugeras and Mourrain(1995)[75] show that the $\mathcal{Q}$ equations are algebraically dependent on the other two types by using one quadrilinear equation as example. It seems that the $\mathcal{Q}$ equations are linearly dependent on the independent $\mathcal{T}$ equations. Although the $\mathcal{Q}$ equations are dependent ones, they may have numerical advantages for estimating four frames simultaneously. Hartley(1998)[76] indicates it and provides some algorithms for the four-frame geometry.

3.5 Multi-Frame Geometry

This section talks about multi-frame geometry, including n frames, where $n > 4$. There is a lot of redundancy in multi-frame Geometry. Faugeras and Mourrain(1995)[75] claim that there are only three types, $\mathcal{B}$,$\mathcal{T}$, and $\mathcal{Q}$. They indicate that given two bilinear equations, there is only one algebraically independent trilinear equation among three frames. Many researchers give their methods to represent multi-frame geometry using these kinds of equations. Using each type of equation, Heyden(2000)[38] gives one such method. Heyden and Åström (1996)[77] (1997)[78] also elaborate that the trilinear equation is essential. Shashua(1996)[47] and Avidan(1996)[44] claim that $(n-2)$ trifocal tensors are a minimal requirement for n frames. In the previous section, some models are reviewed for three-frame geometry. Indelman(2012)[117] extends his model to the multi-frame cases.

This section extends the previous derivations and analyses to a multi-frame geometry situation. The previous section concludes that the bilinear and the trilinear equations are sufficient for four-frame geometry. So this section extends the three-frame model to simplify the multi-frame geometry.

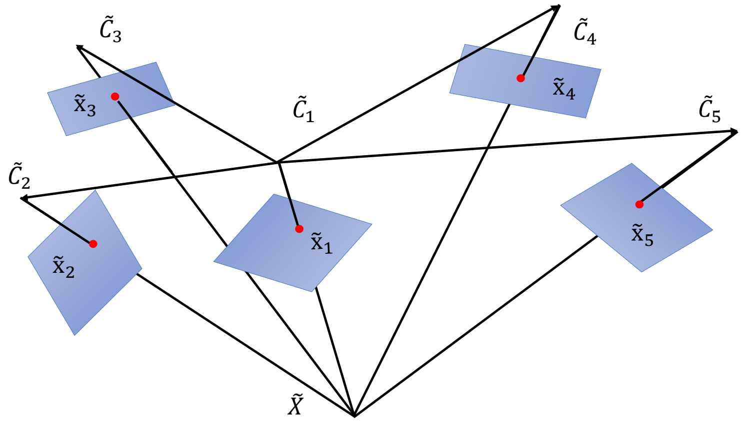


Figure 3.15. The multi-frame geometry.

Given n frames, there are $(n-1)$ image pairs $(1, \mathrm{i})$, where $\mathrm{i} \in [2, n]$ and $2(n-1)$ normal vectors $\mathbf{n1}_{\mathrm{i}}$ and $\mathbf{n2}_{\mathrm{i}}$. The $\mathbf{n1}_{\mathrm{i}}$ and $\mathbf{n2}_{\mathrm{i}}$ represent the first and the second normal vectors of the image pair $(1, \mathrm{i})$, where $\mathbf{n1}_{\mathrm{i}} \parallel \mathbf{n2}_{\mathrm{i}}$. The first three pairs of normal vectors keep the previous notation. The following analysis describes the geometry using these normal vectors.

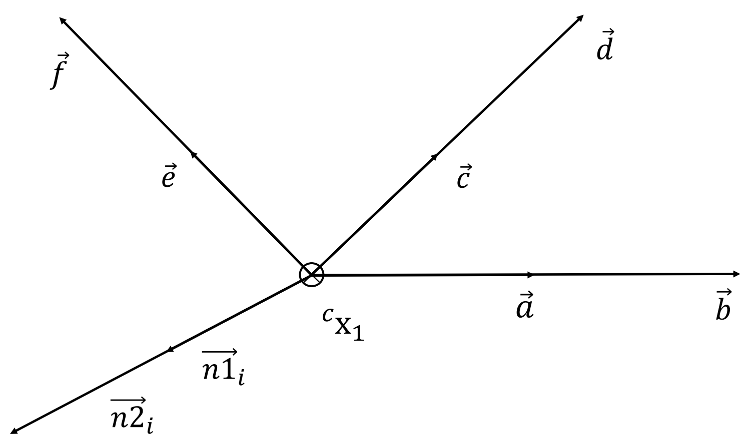


Figure 3.16. The eight normal vectors of five frame planes.

The matrix determinant approach starts from the situation where n rays intersect in one common object space point. This gives the n frame intersection equation.

$$A_{\Lambda n}\mathbf{\Lambda}_n = \mathbf{0} \tag{3.39}$$

An expression showing the elements of $A_{\Lambda n}$ is omitted. The determinant of the $A_{\Lambda n}^T A_{\Lambda n}$ is zero, yielding the equation below.

$$A_{\Lambda n}^T A_{\Lambda n} \mathbf{\Lambda}_n = \mathbf{0}$$

The homogeneous coordinate vector $\mathbf{\Lambda}_n$ is in the null space of this matrix. The zero determinant condition of $A_{\Lambda n}^T A_{\Lambda n}$ gives the equations below.

$$\frac{\|\mathbf{a}\|}{\|\mathbf{b}\|} = \frac{\|\mathbf{c}\|}{\|\mathbf{d}\|} = \frac{\|\mathbf{e}\|}{\|\mathbf{f}\|} = \cdots = \frac{\|\mathbf{n1}_{\mathrm{i}}\|}{\|\mathbf{n2}_{\mathrm{i}}\|} = \cdots = \frac{\|\mathbf{n1}_n\|}{\|\mathbf{n2}_n\|} = \lambda \tag{3.40}$$

This equation equates the $(n-2)$ individual ones as below. This is a conclusion from the four-frame study.

$$\frac{\|\mathbf{a}\|}{\|\mathbf{b}\|} = \frac{\|\mathbf{n1}_{\mathrm{i}}\|}{\|\mathbf{n2}_{\mathrm{i}}\|} = \lambda \tag{3.41a}$$

$$\frac{\|\mathbf{n1}_{\mathrm{i}-1}\|}{\|\mathbf{n2}_{\mathrm{i}-1}\|} = \frac{\|\mathbf{n1}_{\mathrm{i}}\|}{\|\mathbf{n2}_{\mathrm{i}}\|} = \lambda \tag{3.41b}$$

The unified relations of the proportion and parallelism are provided below in the $(n-1)$ equations.

$$\begin{aligned}
\mathbf{a} &= \lambda \mathbf{b} \qquad \mathbf{x}_1^T \mathbf{b} = 0 \\
\cdots &= \cdots \\
\mathbf{n1}_{\mathrm{i}} &= \lambda \mathbf{n2}_{\mathrm{i}} \qquad \mathbf{x}_1^T \mathbf{n2}_{\mathrm{i}} = 0 \\
\cdots &= \cdots \\
\mathbf{n1}_n &= \lambda \mathbf{n2}_n \qquad \mathbf{x}_1^T \mathbf{n2}_n = 0
\end{aligned} \tag{3.42}$$

For each of $(n-1)$ parallel relations in (3.42), there is an equation $\mathcal{B}_{1\mathrm{i}}$. For each of the $(n-2)$ proportion relations in (3.41b), there is an equation $\mathcal{T}_{1,\mathrm{i}-1,\mathrm{i}}$. For the n frames, the total number is $(2n-3)$, which is equal to the number of redundant equations. The extended TFM for multi-frame geometry is given below.

Table 3.4. The TFM for multi-frame geometry

Equation	Geometry	Number
$\mathcal{B}_{1\mathrm{i}}$	$\mathbf{x}_1 \perp \mathbf{n1}_\mathrm{i} \parallel \mathbf{n2}_\mathrm{i}$	n-1
$\mathcal{T}_{1,\mathrm{i}-1,\mathrm{i}}$	$\lVert\mathbf{n1}_{\mathrm{i}-1}\rVert\lVert\mathbf{n2}_\mathrm{i}\rVert = \lVert\mathbf{n2}_{\mathrm{i}-1}\rVert\lVert\mathbf{n1}_\mathrm{i}\rVert$	n-2

It is abbreviated in the equation below.

$$TFM(1,\mathrm{i},N) = \begin{cases} \mathcal{B}_{1\mathrm{i}} \\ \mathcal{T}_{1,\mathrm{i}-1,\mathrm{i}} \end{cases} \tag{3.43}$$

This method is visualized below by figure(3.17). In Indelman's[117] model, a bilinear equation $\mathcal{B}_{\mathrm{i}-1,\mathrm{i}}$ is given to two adjacent frames and a trilinear equations $\mathcal{T}_{\mathrm{i}-2,\mathrm{i}-1,\mathrm{i}}$ is given to three adjacent frames. Considering the expressions of these normal vectors, the equations $\mathcal{B}_{1\mathrm{i}}$ are recommended in the new model.

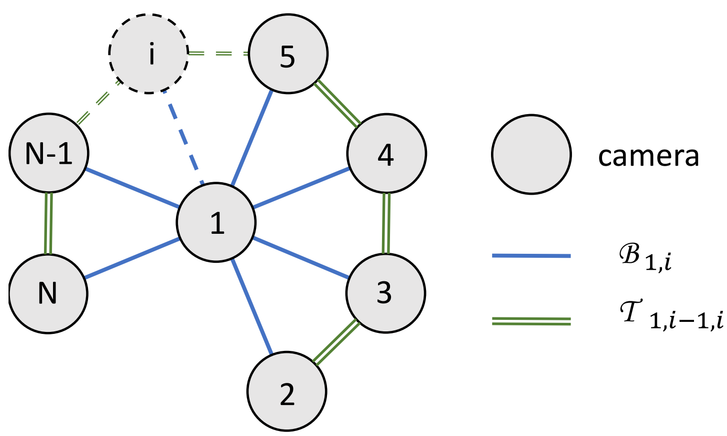


Figure 3.17. The visualization of the TFM function.

The column vector approach also supports the above conclusion and connects some of the previous studies. The n camera matrices are $P_1 = I[I|\mathbf{0}]$ and $P_\mathrm{i} = R_\mathrm{i}[I| - \widetilde{\mathbf{C}}_\mathrm{i}]$, where

$\mathrm{i} \in [2, n]$. Using the simplified projection (3.5) and (3.4), these $2n$ equation are below in the stacked matrix format.

$$\begin{bmatrix} 1 & 0 & m_1 & 0 \\ 0 & 1 & p_1 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & m_N & n_N \\ 0 & 1 & p_N & q_N \end{bmatrix} \mathbf{X} = \mathbf{0} \tag{3.44}$$

The above equation is noted as $A_{sn}\mathbf{X} = \mathbf{0}$. Gaussian elimination gives the equation below as the shown in prior section.

$$\begin{bmatrix} 1 & 0 & m_1 & 0 \\ 0 & 1 & p_1 & 0 \\ \mathbf{0} & \mathbf{0} & \mathbf{U}_n & \mathbf{V}_n \end{bmatrix} \mathbf{X} = \mathbf{0} \tag{3.45}$$

$$\text{where } \begin{bmatrix} \mathbf{U}_n & \mathbf{V}_n \end{bmatrix} = \begin{bmatrix} m_2 - m_1 & n_2 \\ p_2 - p_1 & q_2 \\ \vdots & \vdots \\ m_n - m_1 & n_n \\ p_n - p_1 & q_n \end{bmatrix} \text{ and } \mathbf{U}_n \parallel \mathbf{V}_n$$

There are two parallel vectors, $\mathbf{U}_n$ and $\mathbf{V}_n$, in the lower right corner of the matrix in (3.45). The rank of the stacked matrix is less than four making the $\mathbf{U}_n \parallel \mathbf{V}_n$. Selecting one row from each one of $row_{1,2}$, $row_{3,4}$, and $row_{2\mathrm{i}-3,2\mathrm{i}-2}$ gives two three-dimensional vectors noted as the $\mathbf{u}$ and $\mathbf{v}$, where $\mathrm{i} \in [4, n]$. For instance, selecting the row one, three, and $2\mathrm{i} - 3$ noted as $row_{1,3,2\mathrm{i}-3}$ gives two vectors below. Each selection gives a four-frame including the first, second, third and $\mathrm{i} - th$ frames.

$$\begin{bmatrix} \mathbf{u} & \mathbf{v} \end{bmatrix} = \begin{bmatrix} m_2 - m_1 & n_2 \\ m_3 - m_1 & n_3 \\ m_\mathrm{i} - m_1 & n_\mathrm{i} \end{bmatrix} \text{ and } \mathbf{u} \parallel \mathbf{v}$$

The cross product of $\mathbf{v}$ and $\mathbf{u}$ is $[\mathbf{v}]_\times \mathbf{u} = \mathbf{0}$.

$$\begin{bmatrix} 0 & -n_\mathrm{i} & n_3 \\ n_\mathrm{i} & 0 & -n_2 \\ -n_3 & n_2 & 0 \end{bmatrix} \begin{bmatrix} m_2 - m_1 \\ m_3 - m_1 \\ m_\mathrm{i} - m_1 \end{bmatrix} = \begin{bmatrix} -n_\mathrm{i}(m_3 - m_1) + n_3(m_\mathrm{i} - m_1) \\ n_\mathrm{i}(m_2 - m_1) - n_2(m_\mathrm{i} - m_1) \\ n_3(m_2 - m_1) + n_2(m_3 - m_1) \end{bmatrix} = \mathbf{0} \Leftrightarrow \begin{bmatrix} \mathcal{T}_{13\mathrm{i}} \\ \mathcal{T}_{12\mathrm{i}} \\ \mathcal{T}_{123} \end{bmatrix}$$

The above relation indicates the dependency among three equations $\mathcal{T}_{123}$, $\mathcal{T}_{12\mathrm{i}}$, and $\mathcal{T}_{13\mathrm{i}}$, with two independent ones. It was mentioned in the four-frame geometry section. The $\mathcal{T}_{123}$ and $(n-3)$ $\mathcal{T}_{12\mathrm{i}}$ are sufficient for $\mathbf{U}_n \parallel \mathbf{V}_n$, where $\mathrm{i} \in [4, n]$. Or the $(n-2)$ $\mathcal{T}_{12\mathrm{i}}$, where $\mathrm{i} \in [3, n]$, is the minimal requirement for $\mathbf{U}_n \parallel \mathbf{V}_n$. It is consistent with the theory of Shashua(1996)[47] and Avidan(1996)[44], in which $(n-2)$ trifocal tensors are sufficient. Considering the statement of Faugeras and Mourrain(1995)[75], given $(n-1)$ bilinear equations, these $(n-2)$ trifocal tensors give $(n-2)$ algebraically independent trilinear equations. This derivation summarizes the conclusions of previous scholars and supports the argument here.

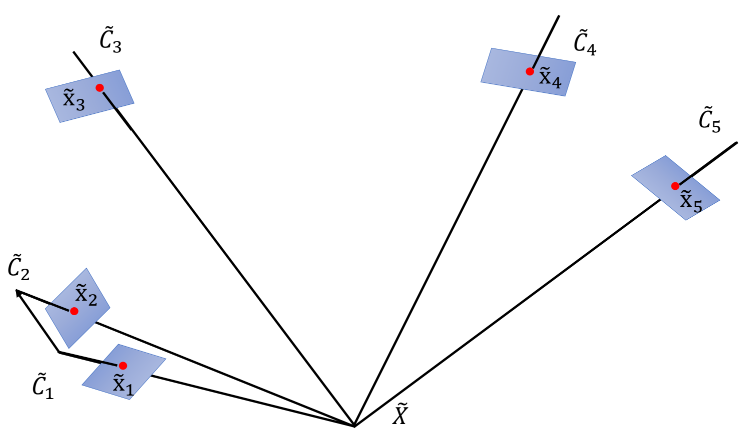


Figure 3.18. The unstable cases.

In the above derivation of $(n-2)$ $\mathcal{T}_{1,2,\mathrm{i}}$, the first two cameras are fixed. In the extended TFM, the $(n-2)$ $\mathcal{T}_{1,\mathrm{i}-1,\mathrm{i}}$ are used. The equivalent of two sets are is proved by equation (3.41). The $(n-2)\mathcal{T}_{1,\mathrm{i}-1,\mathrm{i}}$ are used because if the $\|\widetilde{\mathbf{C}}_2 - \widetilde{\mathbf{C}}_1\|$ is too small, it will affect the whole stability. It is illustrated in the above figure.

3.6 Applications

This chapter provides three applications of the TFM to resolve the issues in the three-dimensional reconstruction task. They are the image coordinate prediction, relative distance estimation, and usage in the bundle adjustment (BA).

The goal of the first two applications is to prepare a reliably initial state for the adjustment. The most important one is using the TFM as the condition equations in BA. The details of how to exploit it in the adjustment mechanics are introduced here.

3.6.1 Image Coordinate Prediction

Image coordinate prediction can be realized through the TFM. Especially for a large view camera, the TFM gives guidance about point selection.

Among the 9 N_{ij} equations, there are 6 which have 27 terms. The other 3 equations have 18 terms each. These 3 equations are derived from the base matrices $N_{\mathrm{ii}}, \mathrm{i} = 1, 2, 3$. In the format of the trilinear equation(3.21), N_{ii} makes the H_N^{i} into a 3 by 3 zero matrix. These 3 equations indicate the ratios of the three components of the vector $\mathbf{x}_1$. These equations then give a quick way for estimation of the image point on image one from the second and third images.

$$
\begin{aligned}
y_1/z_1 &= -\mathbf{x}_2^T R_2 H_{11}^2 R_3^T \mathbf{x}_3 / \mathbf{x}_2^T R_2 H_{11}^3 R_3^T \mathbf{x}_3 \\
x_1/z_1 &= -\mathbf{x}_2^T R_2 H_{22}^1 R_3^T \mathbf{x}_3 / \mathbf{x}_2^T R_2 H_{22}^3 R_3^T \mathbf{x}_3 \\
x_1/y_1 &= -\mathbf{x}_2^T R_2 H_{33}^1 R_3^T \mathbf{x}_3 / \mathbf{x}_2^T R_2 H_{33}^2 R_3^T \mathbf{x}_3
\end{aligned}
$$

The TFM could be used for the prediction of $\mathbf{x}_1$, $\mathbf{x}_2$, and $\mathbf{x}_3$ when the other two are known.

For the prediction of $\mathbf{x}_1$, $\mathcal{B}_{12}$ and $\mathcal{B}_{13}$ will be dependent when $\mathbf{a} \parallel \mathbf{c}$. In order to overcome this case, $\mathcal{T}_{123}$ is selected as $\mathbf{a}^T\mathbf{d} = \mathbf{b}^T\mathbf{c}$. The three frame model gives three equations for the prediction of $\mathbf{x}_1$.

For the prediction of $\mathbf{x}_2$ or $\mathbf{x}_3$, $\mathbf{x}_1$ is given, so $\mathbf{a}^T [\mathbf{e}_{\mathrm{i}}]_\times \mathbf{b} = \mathbf{c}^T [\mathbf{e}_{\mathrm{i}}]_\times \mathbf{d}$ can be chosen corresponding to the nonzero component of $\mathbf{x}_1$, if $\|\lambda_{cos}\| < \frac{\sqrt{2}}{2}$. The three frame model gives two equations for the prediction of $\mathbf{x}_2$ and $\mathbf{x}_3$. The λ_{cos} will be computed in alternative ways, in consideration of the unknown being $\mathbf{x}_2$ or $\mathbf{x}_3$. There are two normal vectors $\mathbf{n}_1$ and $\mathbf{n}_2$, which are parallel to $\mathbf{a}$ and $\mathbf{c}$ respectively.

$$\begin{array}{lll} \mathbf{a} = \mathbf{x}_1 \times R_2^T \mathbf{x}_2 & \quad \mathbf{n}_1 = \mathbf{x}_1 \times \widetilde{\mathbf{C}}_2 & \quad \mathbf{a} \parallel \mathbf{n}_1 \\ \mathbf{c} = \mathbf{x}_1 \times R_3^T \mathbf{x}_3 & \quad \mathbf{n}_2 = \mathbf{x}_1 \times \widetilde{\mathbf{C}}_3 & \quad \mathbf{c} \parallel \mathbf{n}_2 \end{array}$$

The $\langle \mathbf{a}, \mathbf{c} \rangle$ will be replaced by $\langle \mathbf{n}_1, \mathbf{c} \rangle$ or $\langle \mathbf{a}, \mathbf{n}_2 \rangle$ corresponding to the prediction of $\mathbf{x}_2$ or $\mathbf{x}_3$.

Table 3.5. Image point prediction

Given	Unknown	Equations	Angle
$\mathbf{x}_1, \mathbf{x}_2$	$\mathbf{x}_3$	$\mathcal{B}_{13}, \mathcal{T}_{123}$	$\langle \mathbf{a}, \mathbf{n}_2 \rangle$
$\mathbf{x}_1, \mathbf{x}_3$	$\mathbf{x}_2$	$\mathcal{B}_{12}, \mathcal{T}_{123}$	$\langle \mathbf{n}_1, \mathbf{c} \rangle$
$\mathbf{x}_2, \mathbf{x}_3$	$\mathbf{x}_1$	$\mathcal{B}_{12}, \mathcal{B}_{13}, \mathcal{T}_{123}$	

Both $\mathcal{B}$ and $\mathcal{T}$ can be written in the form of the inner product of the two 3 dimensional vectors.

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \mathbf{x}_2^T R_2 \begin{bmatrix} H^1 \\ H^2 \\ H^3 \end{bmatrix} R_3^T \mathbf{x}_3 \qquad \begin{bmatrix} u & v & w \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} = \mathbf{0}$$

For the prediction of $\mathbf{x}_1$, the three equations can be written in the matrix form $A_1 \mathbf{x}_1 = \mathbf{0}$. The vector $\mathbf{x}_1$ is a null-space vector of the 3 by 3 matrices A_1. For the prediction of $\mathbf{x}_2$ or $\mathbf{x}_3$, $\mathcal{B}$ and $\mathcal{T}$ contribute two parameter vectors, whose cross product is the prediction.

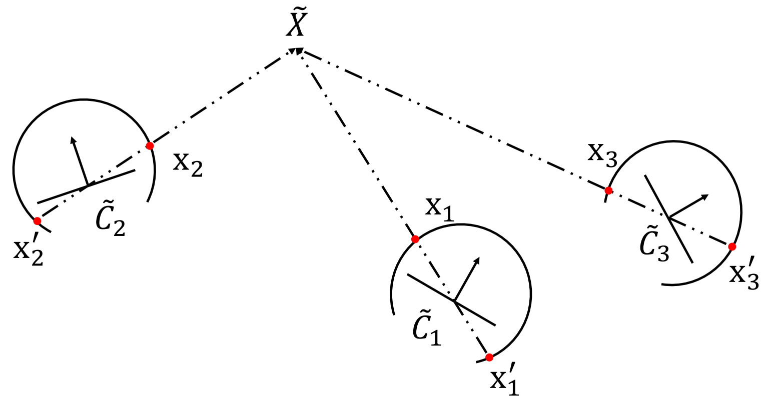


Figure 3.19. Very large field of view camera.

In the figure(3.19) the circular arcs, arrows and solid line represent view angles, Z directions and XOY plane. The $\mathbf{x}$ and $\mathbf{x}'$ represent the image points on the same light ray.

But there still is ambiguity for the sign of the predicted coordinate, which cannot be resolved by epipolar geometry or trifocal tensor based prediction. Because both of The vectors $\pm\mathbf{x}$ satisfy the linear conditions. In equation(3.16),the sign of $\lambda' s$ is dependent on the corresponding predicted image coordinates. The constraints $\lambda' > 0$ resolve it by forcing the object point to be located in the front of the image points. For a very large field of view camera, whose view angle is larger than 180 degrees, this is very important.

3.6.2 Relative Distance Estimation

There is a relative distance issue among three-frame positions. The relative distance is the invariant of the similarity transform. For the 3 frames, the relative scale $\|\tilde{\mathbf{C}}_3\|/\|\tilde{\mathbf{C}}_2\| = t$ is ambiguous which can be resolved from $\mathcal{B}$. The two groups of elements R_2, $\bar{\mathbf{C}}_2$ and R_3, $\bar{\mathbf{C}}_3$ can be computed from two $\mathcal{B}$ individually, with the constraints $\|\bar{\mathbf{C}}_2\| = 1$ and $\|\bar{\mathbf{C}}_3\| = 1$. The $\mathcal{T}$ gives the method of relative scale estimation, by supplying the linear relations among all the projection elements and all the observations. Without a known scale t, $\tilde{\mathbf{C}}_3 = t\bar{\mathbf{C}}_3$, the object point cannot be reconstructed in a unique position, which is shown in the figure.

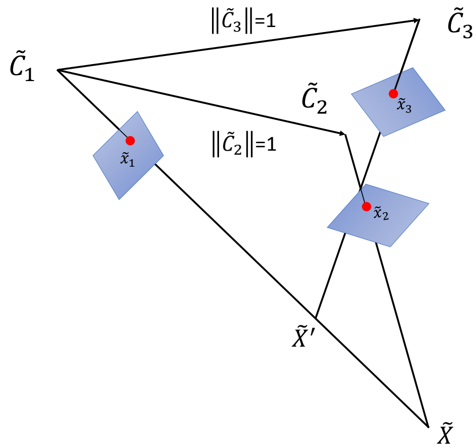


Figure 3.20. Scale ambiguity

Both $\mathcal{B}$ and $\mathcal{T}$ can be written in the form of an inner product as follows.

$$\begin{bmatrix} \mathbf{a}_{c2}^t & \mathbf{a}_{c3}^t \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{C}}_2 \\ \tilde{\mathbf{C}}_3 \end{bmatrix} = \mathbf{0}$$

$\mathbf{a}_{c2}$ and $\mathbf{a}_{c3}$ are two 3 by 1 parameter vectors. For $\mathcal{B}_{12}$ the vector $\mathbf{a}_{c3} = \mathbf{0}$ for example. In order to enforce consistency, an adjustment vector is added to $t\bar{\mathbf{C}}_3$ then $\tilde{\mathbf{C}}_3 = t\bar{\mathbf{C}}_3 + \delta\mathbf{C}_3$ and $\tilde{\mathbf{C}}_2 = \bar{\mathbf{C}}_2$.

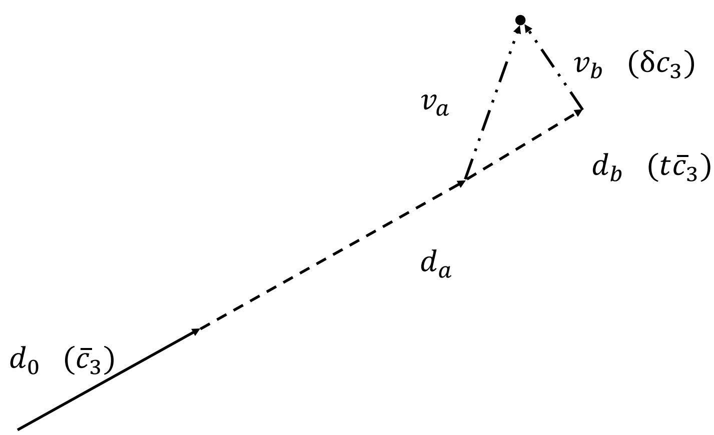


Figure 3.21. Ambiguity of scale and adjustment vector

The above graph shows, for the unit direction vector $\mathbf{d}_0$ any scale will make vector $\mathbf{d}$ with an adjustment vector $\mathbf{v}$. For this reason, we enforce $\bar{\mathbf{C}}_3^t \delta\mathbf{c}_3 = 0$. The following equation will be obtained.

$$\begin{bmatrix} A_{c3} & A_{c3}\bar{\mathbf{c}}_3 \\ \bar{\mathbf{C}}_3^T & 0 \end{bmatrix} \begin{bmatrix} \delta\mathbf{C}_3 \\ t \end{bmatrix} = \begin{bmatrix} -A_{c2}\bar{\mathbf{C}}_2 \\ 0 \end{bmatrix} \tag{3.46}$$

A_{c2} and A_{c3} are two matrices containing the vectors $\mathbf{a}_{c2}^t$ and $\mathbf{a}_{c3}^t$ as the row vectors. In this TFM, $\mathcal{B}_{13}$ and $\mathcal{T}_{123}$ contribute two equations for each image point triplet. In order to resolve the equation(3.46), 2 image point triplets are needed in this method.

By using the result of the two-frame geometry, recovering the relative orientation of three frames is much easier than using the six-point algorithm. Plus, it avoids using the object point $\mathbf{X}$, which is unreliable when the point is very far from the cameras or when the two cameras are separated by a small displacement, which will reduce the occlusion area.

Here, we propose a two-step algorithm for three frame relative orientation. First, compute the relative orientation for two pairs $\mathcal{B}_{12}$ and $\mathcal{B}_{13}$ individually for the three frames. Second, resolve the scale ambiguity using the first step results with $\mathcal{T}_{123}$.

Algorithm 1 Two Step Relative Orientation for Three Frames

Input: The matched point pairs;
Output: The three frame relative orientation elements and matched image points;
1: Relative Orientation for pairs ij using $\mathcal{B}_{\mathrm{ij}}$, including linear estimation and nonlinear optimization;
2: Eliminate the mismatches by using $\mathcal{B}_{\mathrm{ij}}$ and match the common points for three frames;
3: Resolve scale problem for three frames,including linear estimation using the proposed three frame model and nonlinear optimization such as block adjustment;
4: Eliminate the mismatches by using $\mathcal{T}_{123}$;

Given more than two frames, the $\|\widetilde{\mathbf{C}}_2\|$ will be fixed as 1. Given more than three frames, the n frames can be grouped into a sequence of three frame models ij$k$ triplets. The relative scale $t_{\mathrm{ij}} = \|\widetilde{\mathbf{C}}_{\mathrm{i}}\|/\|\widetilde{\mathbf{C}}_{\mathrm{j}}\|$ can be estimated one by one. The global length of $\|\widetilde{\mathbf{C}}_{\mathrm{i}}\|$ is $t_{\mathrm{ij}}\|\widetilde{\mathbf{C}}_{\mathrm{j}}\|$.

3.6.3 New Bundle Adjustment Methods

The numerical implementation of bundle block adjustments can lead to large systems of equations, which may possibly contain nearly parallel rays for multi-image observations of a given object point, leading to instabilities. Locating and deleting the nearly parallel rays can be tedious and slow. The proposed strategy (a) obviates the need to do this editing step, and (b) has the side benefit of reducing the size of the system of the equations which need to be solved. The strategy will accomplish this by not carrying object point parameters for any points where unnecessary.

The objective is to exploit both strong and weak geometry by mixed utilization of the indirect observations model (Gauss-Markoff-model) and the general model (Gauss-Helmert-model). The space points are divided into two classes. A first class object point is any point that has space information, such as a control point or any point with other object space constraints. This kind of point requires not only the image conditions but also the additional external constraints. A second object class point requires the image conditions only.

The Indirect observations model using the collinearity equations for camera geometry can accommodate external information into the adjustment. Using the bilinear and trilinear equations, the general model can avoid the space point parameters, which are not necessary, and may introduce instabilities.

In this section, two BA algorithms are developed, which employ the TFM. The TFM is used to compose condition equations totally or partially replacing the collinearity equations. The first one uses the TFM only as default. Considering the weak geometry situations, this method has the numerical advantage by avoiding the unstable estimations of the object structures. The second one uses both the TFM and the collinearity equations. In this method, the collinearity equation is used for the class one object points, and TFM is used for the second class object points.

Faig(1975)[91] has developed two camera calibration algorithms. One uses the bilinear equation only. The other one uses both the bilinear equation and the collinearity equations.

For the different environments and purposes, the BA methods have different requirements and priorities. These kinds of BA algorithms still deserve attention and study.

Because this research assumes a calibrated camera, the current unknowns in the two methods are only the positions and attitudes.

Table 3.6. BA methods

Method	Condition	Constraint	Model
Traditional BA	Collinearity	Yes	Indirect Observation
Method One	$\mathcal{B}$ and $\mathcal{T}$	No	General
Method Two	All of above	Yes	Constrained General

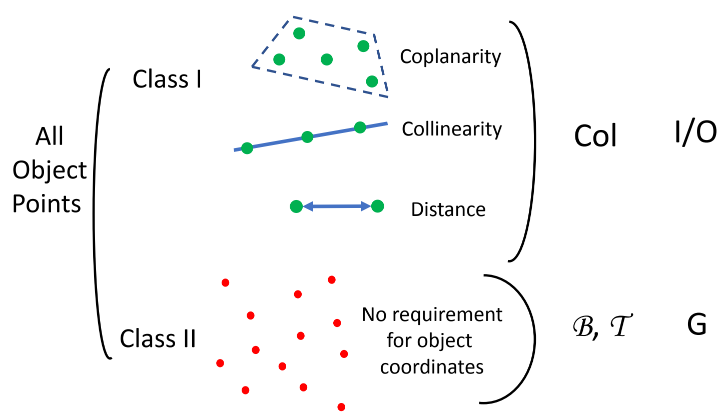


Figure 3.22. The visualization of the two classes of object points.

This figure indicates the two different choices for different kinds of object points.

A, B_c and B_p are the Jacobian matrices for image points, camera parameters and object point parameters, respectively. Δ_c and Δ_p are the unknowns for camera parameters and space point parameters. v is the vector of the residuals.

The indirect observations model is shown below.

$$I_1 v + B_{c1}\Delta_c + B_{p1}\Delta_p = \mathbf{f}_1. \tag{3.47}$$

The indirect observations model is widely used in the BA. It is appropriate for class one object points. The model can be used if the observations can be expressed as functions of the unknowns, such as (3.2). This model requires the space point coordinates, as well as good initial approximations.

Table 3.7. Traditional BA

Traditional BA	Name	Expression
Condition	Collinearity	3.2
Model	Indirect observations model	3.47

The first new method uses the TFM as condition equations and the general adjustment model. The general model is appropriate for object points of class two. This model can be used if the observations have only internal condition equations. This model does not require the object point coordinates. It avoids the space point estimation. As mentioned earlier, estimation of these parameters may introduce instabilities, depending on geometry, and will significantly enlarge the system of equations to be solved.

The general model is displayed below.

$$A_2 v + B_{c2}\Delta_c = \mathbf{f}_2. \tag{3.48}$$

Table 3.8. The new BA method one

Method One	Name	Expression
Condition	$\mathcal{B}$ and $\mathcal{T}$	3.43
Model	General model	3.48

The second new adjustment model is the constrained general adjustment model. If the first kind of point is observed, external constraints among the object space parameters can be added to the above method. The external constraints could be the known coordinates or any prior geometric relationship.

External constraints are:

$$D_p \Delta_p + D_s \Delta_s = \mathbf{h}.$$

Δ_s is the parameter vector of spatial features or spatial constraints, and Δ_p is the vector of the object points.

When both models are used in the same adjustment, the combined form will be:

$$\begin{aligned} A_3 v + B_{c3} \Delta_c + B_{p3} \Delta_p &= \mathbf{f}_3, \\ D_p \Delta_p + D_s \Delta_s &= \mathbf{h}. \end{aligned} \tag{3.49}$$

And the first equation in above expression is

$$\begin{bmatrix} I_1 & 0 \\ 0 & A_2 \end{bmatrix} v + \begin{bmatrix} B_{c1} & B_{p1} \\ B_{c2} & 0 \end{bmatrix} \begin{bmatrix} \Delta_c \\ \Delta_p \end{bmatrix} = \begin{bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \end{bmatrix} \tag{3.50}$$

Table 3.9. The new BA method two

Method One	Name	Expression
Condition	All	
Model	General model	3.49

Considering the singular cases of the equations (3.7) and (3.22), the $\widetilde{\mathbf{C}}_\mathrm{i} = \widetilde{\mathbf{C}}_\mathrm{j}$ makes the TFM trivial. Under the numerically unstable situation when $\|\widetilde{\mathbf{C}}_\mathrm{i} - \widetilde{\mathbf{C}}_\mathrm{j}\|$ is very small, the results could converge to $\widetilde{\mathbf{C}}_\mathrm{i} = \widetilde{\mathbf{C}}_\mathrm{j}$.

Rodríguez(2011)[115] suggests using the normalized bilinear equation below.

$$\mathcal{B}_\mathrm{ij} / \|\widetilde{\mathbf{C}}_\mathrm{i} - \widetilde{\mathbf{C}}_\mathrm{j}\| = 0 \tag{3.51}$$

The normalized trilinear equation is given below. The derivatives of it are in the appendix(A.4.4). This also addresses the case of the short baselines.

$$\mathcal{T}_{\mathrm{ij}k}/\|\widetilde{\mathbf{C}}_{\mathrm{j}} - \widetilde{\mathbf{C}}_{k}\| = 0 \tag{3.52}$$

And the normalized TFM for BA is given below.

$$TFM(1, \mathrm{i}, n) = \begin{cases} \mathcal{B}_{1\mathrm{i}}/\|\widetilde{\mathbf{C}}_{\mathrm{i}} - \widetilde{\mathbf{C}}_{1}\| \\ \mathcal{T}_{1,\mathrm{i}-1,\mathrm{i}}/\|\widetilde{\mathbf{C}}_{\mathrm{i}} - \widetilde{\mathbf{C}}_{\mathrm{i}-1}\| \end{cases} \tag{3.53}$$

The $\mathcal{B}_{1\mathrm{i}}/\|\widetilde{\mathbf{C}}_{\mathrm{i}} - \widetilde{\mathbf{C}}_{1}\|$ makes sure $\widetilde{\mathbf{C}}_{\mathrm{i}} \neq \widetilde{\mathbf{C}}_{1}$. And the $\mathcal{T}_{1,\mathrm{i}-1,\mathrm{i}}/\|\widetilde{\mathbf{C}}_{\mathrm{i}} - \widetilde{\mathbf{C}}_{\mathrm{i}-1}\|$ makes sure $\widetilde{\mathbf{C}}_{\mathrm{i}} \neq \widetilde{\mathbf{C}}_{\mathrm{i}-1}$.

4. EXPERIMENTS With The DEVELOPED MODELS And ALGORITHMS

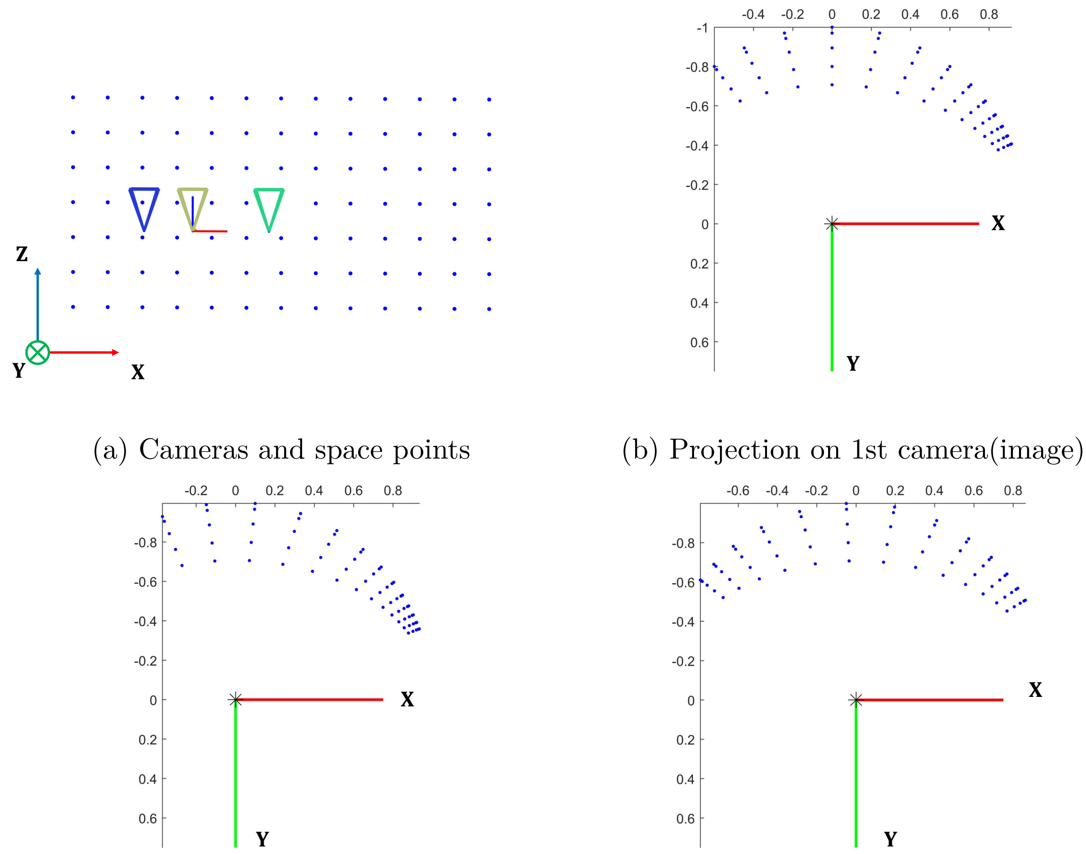
Some numerical experiments are presented here to demonstrate and verify the new methods.

The first two experiments demonstrate the two applications: image transfer and relative distance estimation.

The third set of experiments compare the conventional BA and the two new ways with regard to success rate, accuracy, and precision. Before the description of the BA experiments, the issues of weak spatial geometry and spatial constraints are also introduced. The algorithms and methods are tested on simulated data as well as on one real data set.

4.1 Image Transfer on Wide Field of View Camera

For the prediction application, the following experiment is done using synthetic camera positions and space points. The projections and predictions are computed for the space points. The projections are computed by projecting the space points via the camera model. The predictions of the third camera are computed by using the three frame method described here both with and without the constraint of $\lambda's > 0$. The two groups of images are compared.



(a) Cameras and space points

(b) Projection on 1st camera(image)

(c) Projection on 2nd camera(image)

(d) Projection on 3rd camera(image)

Figure 4.1. Comparison of the projections using wide field of view camera.

In the plots above, the three triangles represent three cameras. The red and green segments represent the x and y axis of camera coordinate system respectively. The z axis is perpendicular with the paper on the image plots. The black asterisk is camera perspective center.

In the first plot (a) above, the space points are on a horizontal plane which is above the three cameras (cameras looking horizontally) and parallel to the XOZ plane of the first camera. In the other plots, the image points are normalized with $\|\mathbf{x}\| = 1$, in each camera coordinate system. The purpose is to show the ambiguities inherent in conventional wide field of view camera models(that is field of view > 180 degrees), which are resolved by proper invocation of the $\lambda' > 0$ constraint with the new model.

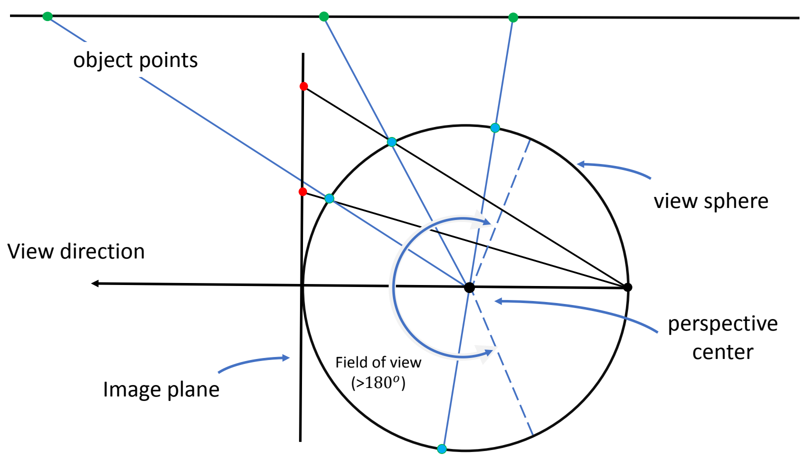
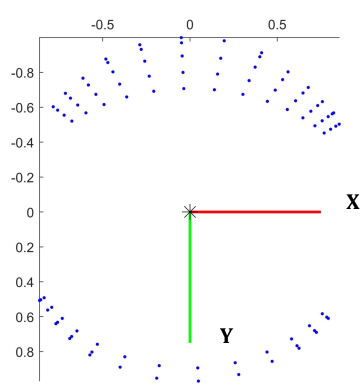
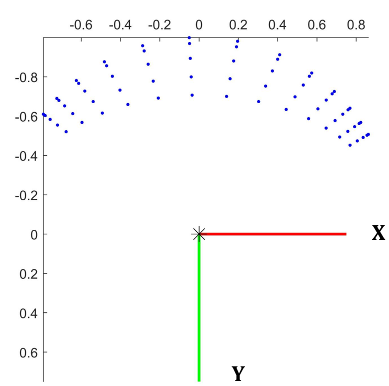


Figure 4.2. Fisheye camera geometry.

In the figure(4.2), the view direction and the field of view are shown. The green and red points represent object points and image points. That is, the points are first projected onto a sphere of radius 1, then projected onto the image plane as in figure(4.2). This is to simulate the image geometry of a "fisheye" lens.



(a) Prediction of 3rd camera(image) without $\lambda' s > 0$



(b) Prediction of 3rd camera(image) with $\lambda' s > 0$

Figure 4.3. Comparison of the predictions using wide field of view camera.

Without the constraint of $\lambda' s > 0$, the predictions can be $\pm\mathbf{x}$, which is indicated in figure (4.3a). Some predicted points appear behind the camera perspective center. For very large field of view cameras, the equations (3.16) are used to resolve this problem.

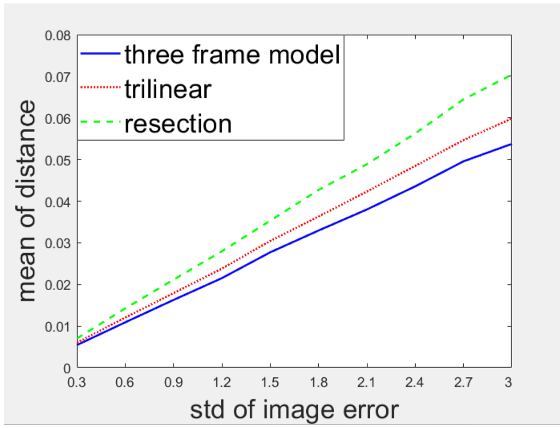
4.2 Relative Distance Estimation

For the relative length estimation, three methods are compared in this experiment. A synthetic small view angle calibrated camera is used. Gaussian noise with mean zero and standard deviation(std) σ pixels are added to the synthetic image points with an assumption that the focal length is 3000 pixels. The three camera axes are all parallel. The three camera positions are $\widetilde{\mathbf{C}}_1 = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$, $\widetilde{\mathbf{C}}_2 = \begin{bmatrix} -\sqrt{2}/2 & \sqrt{2}/2 & 0 \end{bmatrix}^T$, and $\widetilde{\mathbf{C}}_3 = \begin{bmatrix} \sqrt{2} & \sqrt{2} & \sqrt{2} \end{bmatrix}^T$. For the main purpose, which is to resolve the scale factor $t_{32} = \|\widetilde{\mathbf{C}}_3\|/\|\widetilde{\mathbf{C}}_2\|$, the experiment assumes that the other elements, such as $\bar{\mathbf{C}}_2$ and $\bar{\mathbf{C}}_3$ (normalized camera stations), are known.

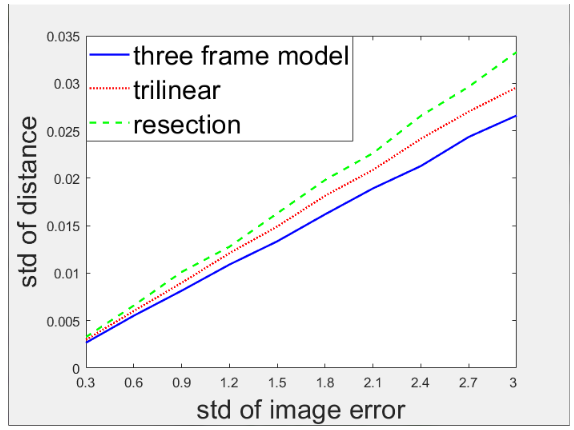
The experiment is done under two scenarios. In the first one, the space points are close to these frames. The distances between the first camera and space points are around 30 units. In the second one, the space points are far from these frames. The distances are around 300 units. The experiment is repeated 10 times with σ increasing in equal increments from to 0.3 to 3 for each scenario.

The three methods are the three-frame model, the trilinear equation only, and the space resection method. The first method is introduced in the previous section. The trilinear equation method uses the three trilinear equations, (3.20), corresponding to N_{11}, N_{22}, and $N_{21} + N_{12}$. The space resection method reconstructs the space points from the first two frames then estimates the relative length by space resection. The unknowns of equation (3.46) are resolved by the three methods, respectively.

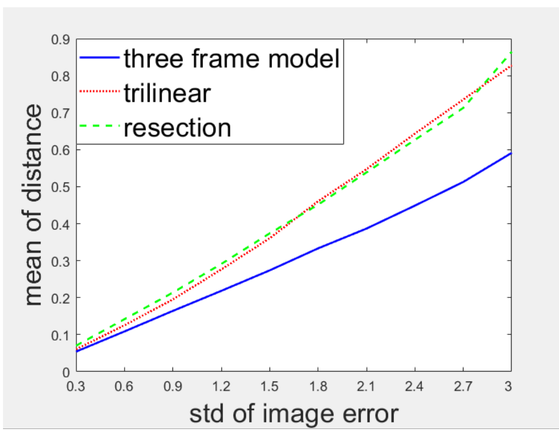
The graphs below show the mean and standard deviation of distances between the true position and its estimate. The solid, point and dash lines represent the three frame model, the trilinear and resection methods.



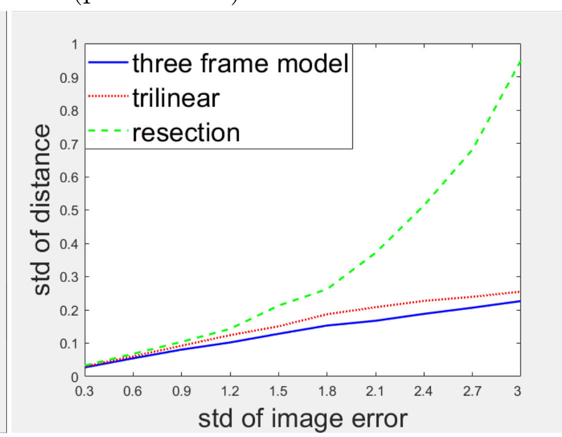
(a) plot of means of first scenario(points close)



(b) plot of standard deviations of first scenario(points close)



(c) plot of means of second scenario(points far away)



(d) plot of standard deviations of second scenario (points far away)

Figure 4.4. Comparison of the three methods under two scenarios(points close and far away).

Under the first scenario(nearby points), the three methods have commensurable performance. With image errors increasing, the error magnitude of the proposed three frame model method is still smaller than the other two methods. Under the second scenario(far away points), the means of the error of the trilinear and resection methods are commensurable and higher than the proposed three frame model method. The standard deviations of three frame model and trilinear methods are smaller than the resection method.

This experiment shows that for both of the ”close space points” and ”far space points” scenarios, the methods without the space points reconstruction have the better performance.

Under the two situations, the proposed three frame model method has advantages over the others.

4.3 Small Intersection Angle

Before the BA experiments, this section shows the issue of weak spatial configurations. The weak space configurations here are the ones that make unstable object point estimation. This issue often occurs when the intersection angle is small. Then small errors of observations will cause significant changes in the estimates of space points.

Some cameras, exposure frequency, platform motion, and object point structures will cause this issue, yielding large camera to point distance, relative to the base distance. A short baseline means a short camera-to-camera distance. And a large depth makes the large point-to-camera distance. The ratio of the depth to the length of the baseline is called the depth-base ratio.

$$r_{db} = \frac{Depth}{Base} \tag{4.1}$$

The larger the ratio, the worse the geometric conditions for intersection stability.

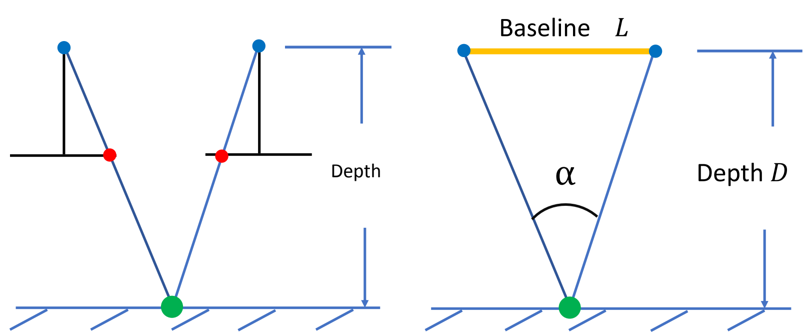


Figure 4.5. Schematic diagram of intersection angle and space intersection from two images.

In the figure above, the red points are image observations, and the green point is an object point. The yellow segment in the right diagram is the baseline between two cameras.

Camera movements(along a flight line, for example) make the intersection angle different. If the camera moves the same distance in different directions, the intersection angles are different. For an object point in front of the camera, the camera moving perpendicular to the optical axis results in a larger intersection angle. The camera moving along the optical axis leads to a small intersection angle.

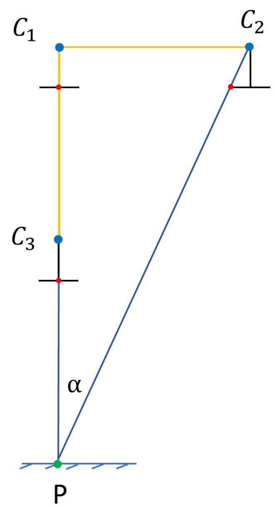
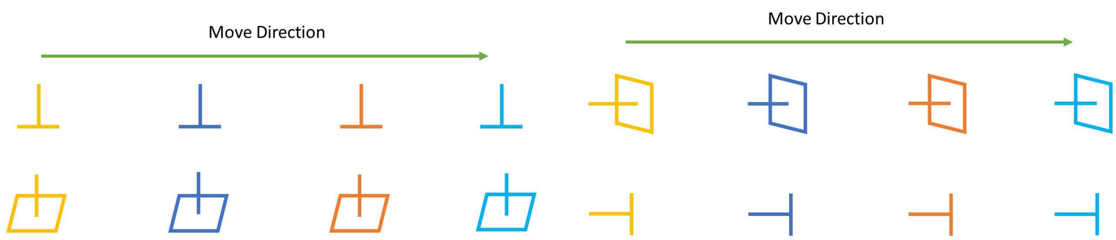


Figure 4.6. The camera movements make the intersection angle different.

Segment $\mathbf{C_1C_2}$ represents the movement perpendicular to the optical axis. Segment $\mathbf{C_1C_3}$ represents the movement along the optical axis. Two baselines have the same length. The two types are the horizontal one and the forward one.



(a) Two views of the horizontal movement. (b) Two views of the forward movement.

Figure 4.7. Two types of camera movements

The two different movements make different space camera-point structures. With the horizontal type, the camera moves along a direction that is perpendicular to the optical axis. On the contrary, with the forward type, the camera moves along the optical axis only. In traditional photogrammetric tasks, the first mode is the normal case in aerial photogrammetry which has always dominated network design, such as overlap and fight height. The second situation is common when using a portable camera or a vehicle-mounted video recorder. In practice, the real camera movement is more complicated. The more flexibility and diversity of camera movements cause depth-base variability. Processing algorithms should be able to seamlessly accommodate such variability.

The following plots indicate the strength benefits of the two movements.

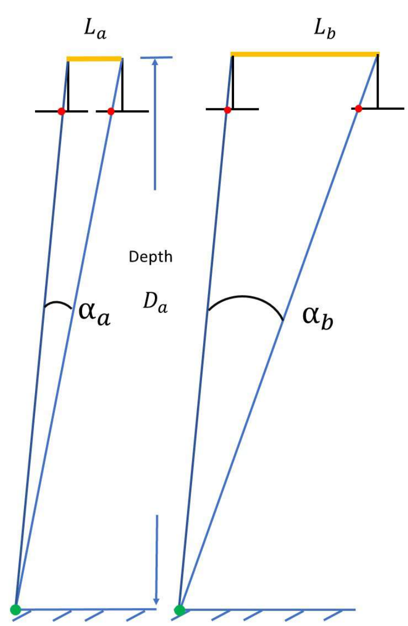


Figure 4.8. Small intersection angle of the horizontal movement.

The orange segmentation presents the distance between the two cameras. In this horizontal case, the short camera baselines and the large point-to-camera distances cause small intersection angles. A longer baseline results in a larger intersection angle from horizontal movement. Compared to this case, the small intersection angle issue occurs much more in the forward movement(along the depth axis). Even a long camera baseline cannot avoid the issue.

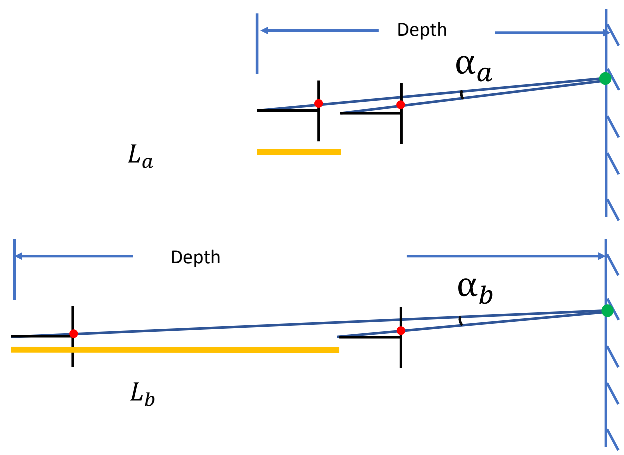


Figure 4.9. Small intersection angle of the forward movement.

In the forward path, the overlapped area always appears in front of the cameras. The closer the object points are to the optical axis, the smaller the intersection angle. When the object point is located in the center of the view, increasing the length of the baseline cannot increase the intersection angle dramatically, which is shown in the figure(4.9).

In photogrammetric tasks, point-camera relative positioning is often carefully designed to avoid weak geometry. When using a portable handheld camera, such weak geometry cannot be easily avoided. The so-called inverse depth technique is developed for far object points. In some optimization techniques, the object points are classified into two classes, which are close object points and far object points. An experiment to show the benefits of the proposed three-frame model versus conventional modeling in the case of weak intersection geometry is described in section (4.6).

4.3.1 Experiment of Incorrect Estimates

This experiment tests the incorrect estimation of one object point caused by the small intersection angle issue. The incorrect estimate here means the estimated object point appears behind any cameras which observe this object point. The estimation method uses the camera projection function, together with a cross product.

$$
\begin{aligned}
[\mathbf{x}]_\times R[I| - \widetilde{\mathbf{C}}]\mathbf{X} &= 0 \\
[\mathbf{x}]_\times R[\widetilde{\mathbf{X}} - \widetilde{\mathbf{C}}] &= 0 \\
[\mathbf{x}]_\times R\widetilde{\mathbf{X}} &= [\mathbf{x}]_\times R\widetilde{\mathbf{C}}
\end{aligned}
\tag{4.2}
$$

Each expression has two independent equations. Given n frames, there is a matrix having $2n$ independent equations. If some R_i and $\widetilde{\mathbf{C}}_\text{i}$ are known, the $\widetilde{\mathbf{X}}$ could be computed. Two frames are the least requirements. The estimated $\widetilde{\mathbf{X}}$ is the initial value of the intersection point for the BA algorithm. It is tested repeatedly under different conditions.

The experiment has five variables. They are the movement type, the number of frames, the depth-base ratio r_{db}, the noise of the camera attitudes and positions. In the all trials, the zero-mean Gaussian distributed noise is added to the image observations. The standard deviation of measurement errors is one pixel. For the small intersection angle issue, the influence of the camera angle errors on the object point estimation is far greater than the influence due to camera position errors. The experiment tests the camera attitude errors mainly.

The first variable is movement type. The two types are demonstrated in the figure(4.7). The coordinate system of first camera is established as the reference coordinate system. All the cameras are parallel to each other. The distance between each pair of adjacent cameras is one unit distance.

The second variable is the number of the cameras. The number ranges from two to four.

The third one is the the depth-base ratio r_{db}. The ratio have seven levels, $r_{db} = [15, 30, 45, 60, 75, 90, 105]$.

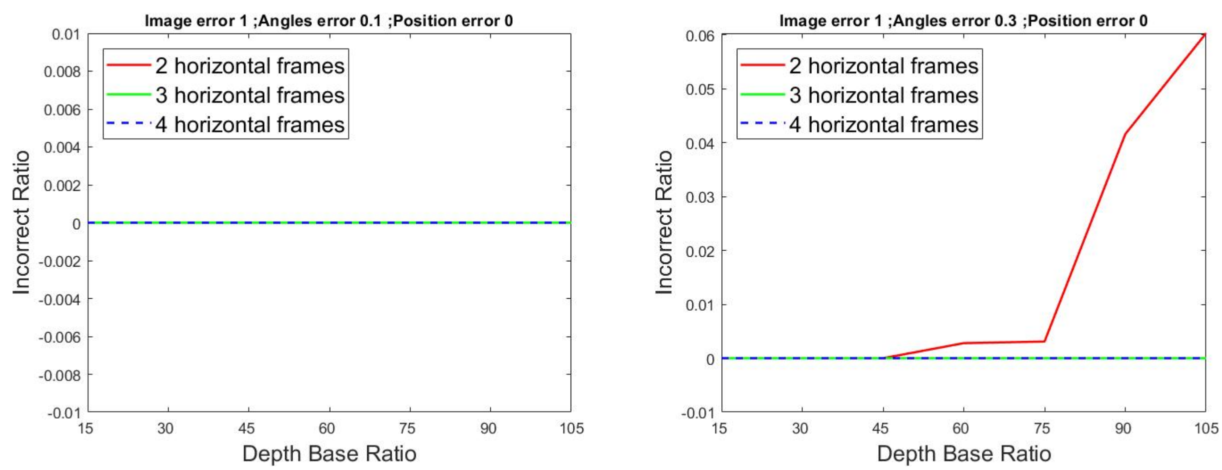
The fourth variable is the errors of the camera attitudes. The zero-mean Gaussian noise has two levels. The standard deviations of the two levels are 0.1 degree and 0.3 degree.

The fifth variable is the errors of the camera positions. The standard deviation of these errors is 0.03 unit distance.

At each depth-base ratio or error level, the test is repeated ten thousand times. Each time only one space point is tested. When the estimated object point appears behind any camera, this estimate is considered incorrect. The incorrect Estimate ratio r_{ie} is the proportion of the number of incorrect estimates relative to the total number.

$$r_{\text{ie}} = \frac{\text{the number of incorrect estimation}}{\text{the total number of estimations}} \tag{4.3}$$

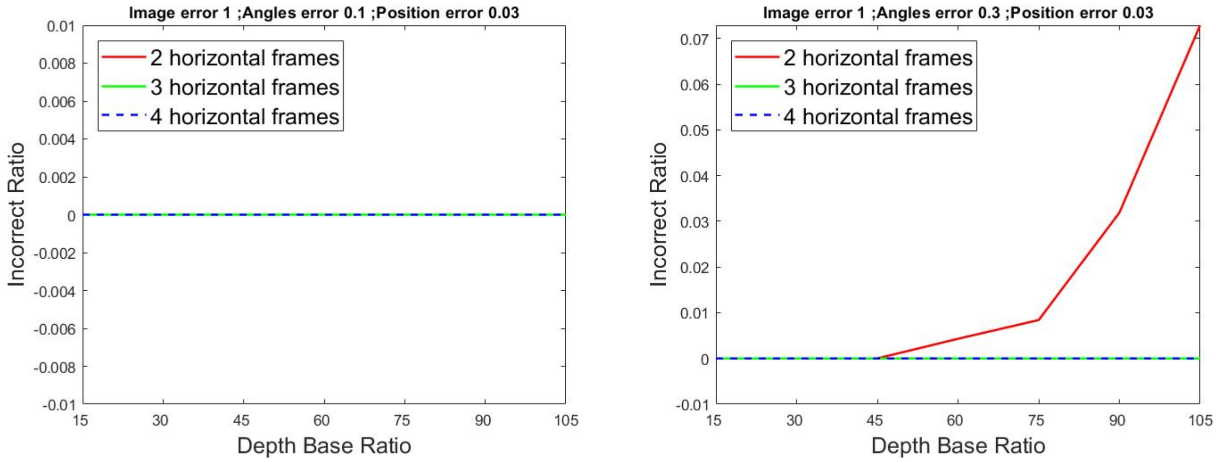
The experiment results are plotted here.



(a) camera rotation error level one. (b) camera rotation error level two.

Figure 4.10. Result of horizontal movement

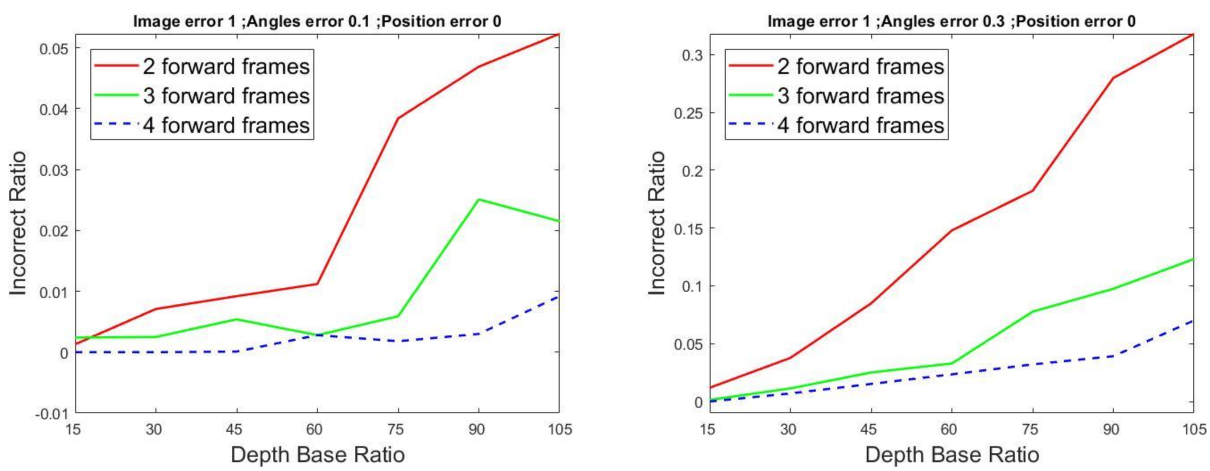
The two graphs (Figure 4.10) indicate the results of horizontal movement under two attitude error levels. In these tests, the true camera positions are kept. The left graph shows that no incorrect estimate appears under attitude error level one. The right graph indicates that the larger attitude error makes the incorrect estimates increase when the frame number is two. The ratio of incorrect estimates increases rapidly as the depth increases. When the frame number increases, there is no incorrect estimate.



(a) camera rotation error level one. (b) camera rotation error level two.

Figure 4.11. Result of horizontal movement with position errors

These two graphs (Figure 4.11) show the results of adding the position errors into the corresponding previous tests. Comparing the two previous results, small position errors change the estimate only a little. So the influence of the camera angle errors is far greater than the influence brought by camera position errors.



(a) camera rotation error level one. (b) camera rotation error level two.

Figure 4.12. Result of forward movement

The two graphs (Figure 4.12) indicate the tests of the forward movement(along the depth direction) under two attitude error levels. In the two tests, the true camera positions are kept.

The left one has a small attitude error. This graph group indicates that more redundancy makes the incorrect ratio lower. The larger attitude error makes the ratio higher significantly.

Comparing the graphs between two movement types shows that the ratio can change dramatically depending on spatial configurations. Not only object point distribution but also the distribution of camera locations are important. With the same distance, the same observation, and orientation error level, the forward trajectory has a higher incorrect estimate ratio than the horizontal one. The experiment indicates that with practical camera movements, such as a handheld camera orbit, small intersection angle issues may be a significant problem. Considering the diversity and variability of camera movements, any algorithm should be robust in the face of such variability.

4.4 Three Spatial Features, to Be Used As Constraints

There are three kinds of spatial features used in the constrained BA experiment. They are the point-to-point distance, the spatial line, and the spatial plane. The combination of the spatial plane and spatial line is also used. The unit vector and a rotation matrix are introduced to simplify the expressions.

4.4.1 Point Distance

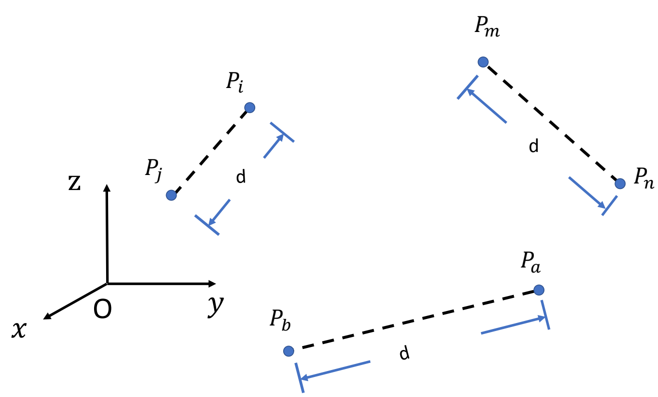


Figure 4.13. The same distance between the three pairs of points.

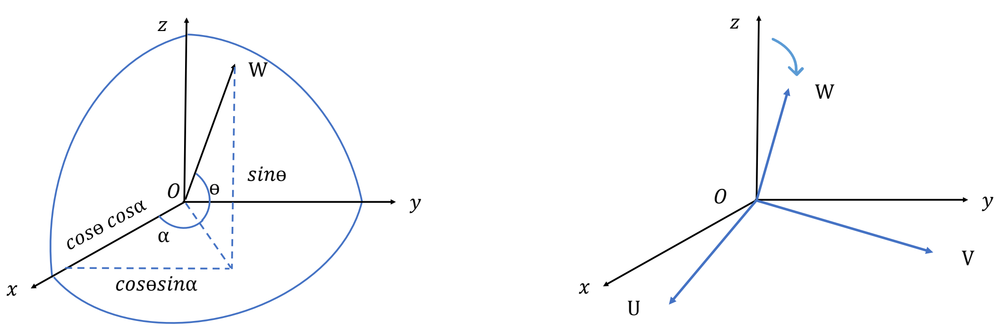
One possible point-to-point distance constraint could be that some pairs of object points, such as $P_{\mathrm{i}}P_{\mathrm{j}}$, have the same unknown distances.

$$\|P_{\mathrm{i}}P_{\mathrm{j}}\| = d \tag{4.4}$$

In the Figure (4.13), three pairs of object points have the same distance, and the constraint is expressed as $\|P_aP_b\| = \|P_{\mathrm{i}}P_{\mathrm{j}}\| = \|P_mP_n\| = d$. The distance d is the unknown of the constraint. One pair of points gives one equation, with the introduction of the unknown, d.

4.4.2 Unit Vector and Rotation

The unit vector and a rotation matrix are introduced before the other spatial features.



(a) Coordinates of a unit vector. (b) Vectors of the rotation matrix .

Figure 4.14. Unit vector and designed rotation matrix

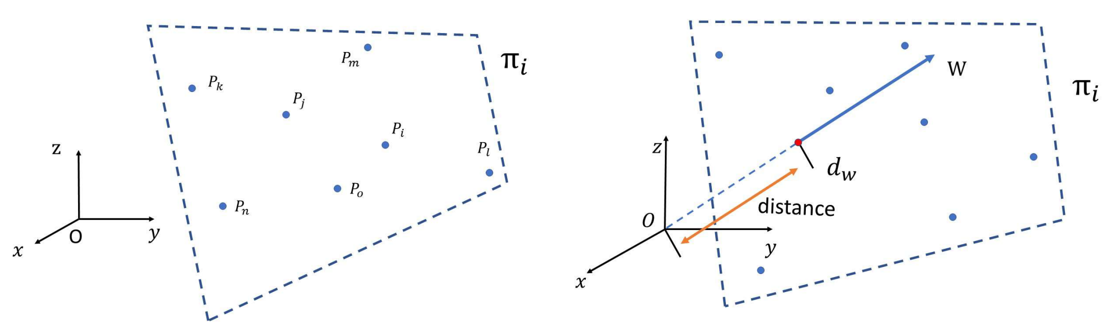
The above-left figure shows a unit vector **W**. A rotation matrix R_W rotates the three basis vectors to the three vectors **U**, **V** and **W**. It is designed below using the components of the unit vector $\mathbf{W}$. The rotation axis is the normal vector of the plane ZOW. The rotation angle is one from OZ to OW.

$$\mathbf{W} = \begin{bmatrix} w_x \\ w_y \\ w_z \end{bmatrix} = \begin{bmatrix} \cos\theta\cos\alpha \\ \cos\theta\sin\alpha \\ \sin\theta \end{bmatrix}$$

$$R_W = \begin{bmatrix} 1 - w_x^2/(1+w_z) & -w_x w_y/(1+w_z) & w_x \\ -w_x w_y/(1+w_z) & 1 - w_y^2/(1+w_z) & w_y \\ -w_x & -w_y & w_z \end{bmatrix} = \begin{bmatrix} \mathbf{U}, & \mathbf{V}, & \mathbf{W} \end{bmatrix}$$

Both the vector $\mathbf{W}$ and this designed rotation R_W have two unknowns. They simplify the expressions of the following spatial features.

4.4.3 Spatial Plane



(a) One figure of a spatial plane.

(b) Normal vector and translation of a spatial plane.

Figure 4.15. Two figures of a spatial plane.

The spatial plane constraint is appropriate for the points distributed on the same plane. The vector $\mathbf{W}$ is the normal vector of the spatial plane $\boldsymbol{\pi}$. The normal vector $\mathbf{W}$ defines a plane passing through the origin. Moving it along the vector $\mathbf{W}$ with a certain distance d_w gives the plane $\boldsymbol{\pi}$. All object points on the plane have the same projection onto the vector

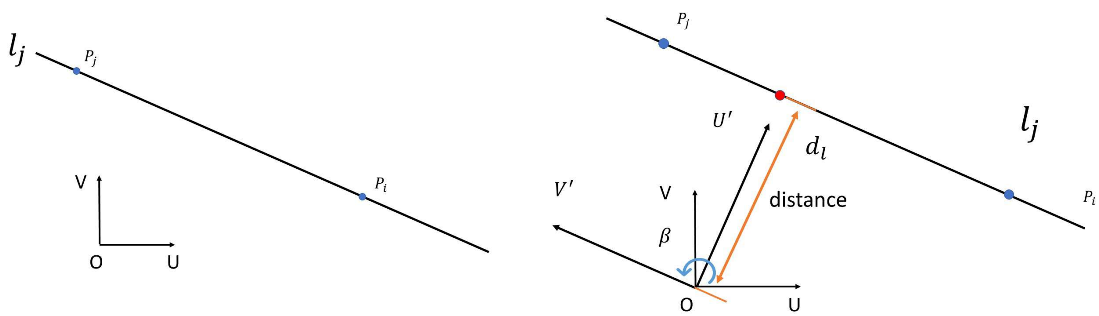
$\mathbf{W}$, which is marked by the red point on the plot above. The expression of the point $\widetilde{\mathbf{X}}$ on the spatial plane is the below one. This expression is from common knowledge of geometry.

$$\begin{aligned} \mathbf{W}^T\widetilde{\mathbf{X}} &= d_w \\ \begin{bmatrix} \mathbf{W}^T & -d_w \end{bmatrix}\mathbf{X} &= 0. \end{aligned} \tag{4.5}$$

Each spatial plane has three unknowns. One point gives one equation of the plane feature.

4.4.4 Spatial Line

The two-dimensional line and the spatial line are introduced here.

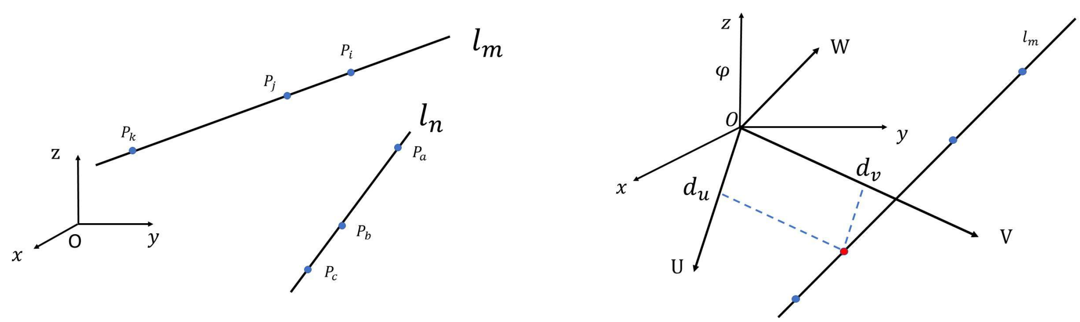


(a) One figure of two-dimensional line. (b) Direction and translation of the line.

Figure 4.16. A two dimensional line.

In the above plots, unit vectors $\mathbf{U}'$ and $\mathbf{V}'$ are the normal vector and direction vector of the two-dimensional line. The $\mathbf{V}'$ defines a line passing through the origin. Moving it along the vector $\mathbf{U}'$ with a certain distance d_l gives the line l_{j}. All points on the line have the same projection onto vector $\mathbf{U}'$, which is marked by the red point on the plot above. The angle from $\mathbf{U}$ to $\mathbf{V}'$ is β, noted as $\beta = \langle \mathbf{U}, \mathbf{V}' \rangle$. The expression of the two-dimensional point $\widetilde{\mathbf{x}} = [p_u, p_v]^T$ on the line is the one below.

$$
\begin{gathered}
\begin{bmatrix} \sin\beta & -\cos\beta \end{bmatrix} \begin{bmatrix} p_u \\ p_v \end{bmatrix} = d_l \\
\begin{bmatrix} \sin\beta & -\cos\beta & -d_l \end{bmatrix} \mathbf{x} = 0.
\end{gathered} \tag{4.6}
$$



(a) Two spatial lines are defined by their affiliated points.

(b) Direction and translation of the spatial line.

Figure 4.17. Two figures of spatial line

In the above right plot, the vector $\mathbf{W}$ is the direction vector of the spatial line. The vector $\mathbf{W}$ defines a line passing through the origin. Moving it along the $\mathbf{U}$ and $\mathbf{V}$ with certain distances d_u and d_v gives the line l_m. The expression of the object point $\widetilde{\mathbf{X}}$ on the line is the one below.

$$
\begin{gathered}
\begin{bmatrix} \mathbf{U}^T \\ \mathbf{V}^T \end{bmatrix} \widetilde{\mathbf{X}} = \begin{bmatrix} d_u \\ d_v \end{bmatrix} \\
\begin{bmatrix} \mathbf{U}^T & -d_u \\ \mathbf{V}^T & -d_v \end{bmatrix} \mathbf{X} = 0.
\end{gathered} \tag{4.7}
$$

All points on the line have the same projection on the plane OUV, which is marked by the red point on the above plot. The spatial line constraint applies to some points that are distributed on the same spatial line. Each spatial line has four unknowns. Each spatial point contributes two equations to the line parameters.

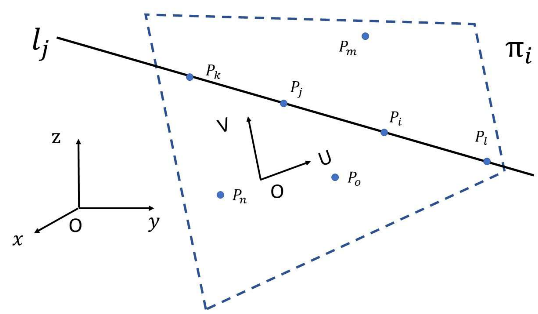


Figure 4.18. A line on the spatial plane.

The combination of a spatial line and a spatial plane is studied here. The object point on the line gives one equation (4.6) to the line and one equation (4.5) to the spatial plane. In the equation (4.6), the coordinates of the point $\widetilde{\mathbf{x}} = [p_u, p_v]^T$ in the plane's local coordinate system is $[p_u, p_v]^T = [\mathbf{U}, \mathbf{V}]^T \widetilde{\mathbf{X}}$. The combination of the two equations is the one below.

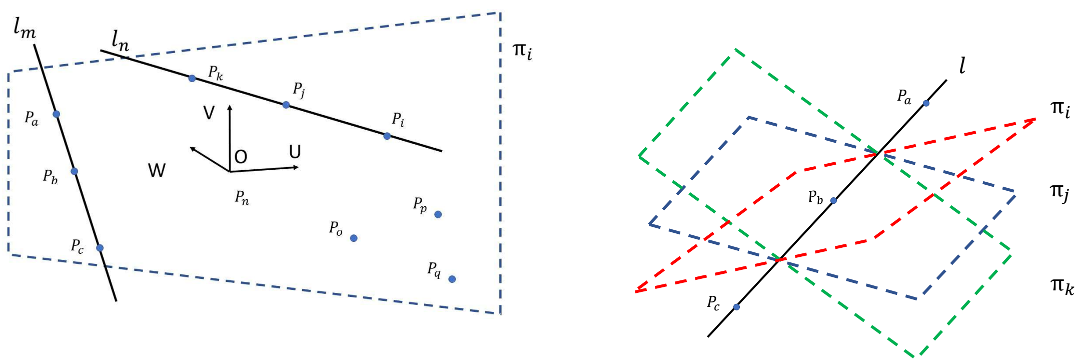
$$\begin{gathered}
\begin{bmatrix} \sin\beta & -\cos\beta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{U}^T \\ \mathbf{V}^T \\ \mathbf{W}^T \end{bmatrix} \widetilde{\mathbf{X}} = \begin{bmatrix} d_l \\ d_w \end{bmatrix} \\
\left[\begin{bmatrix} \sin\beta & -\cos\beta & 0 \\ 0 & 0 & 1 \end{bmatrix} R_w^T, - \begin{bmatrix} d_l \\ d_w \end{bmatrix} \right] \mathbf{X} = \mathbf{0}
\end{gathered} \tag{4.8}$$

Each combination has five unknowns. Three of them are belonging to the spatial plane, and the other two are belonging to the dependent line feature. The first equation in (4.8) includes the dependent line parameters. The second one has plane parameters only.

4.4.5 Equation Setup for the Compound Constraint

This section introduces a way to establish the equations for the compound constraint. The compound constraint here is for the points on more than one spatial feature. For example, the line feature on a given plane has two group parameters. Sometimes object points are located on more than one feature. In this case, the method to set up the parameters and functions is presented here.

A point on the plane, not on any line on the plane, contributes one equation (4.5) to the plane parameters. A point located on line in a plane gives the two equations in equation (4.8).



(a) A example of compound Features. (b) Intersection of multi-plane.

Figure 4.19. Two examples of compound constraint

On the above plot, the points P_o, P_p and P_q present the first kind of the point. And the points P_i, P_j and P_k present the second kind of point. When adding a line feature to the plane, one group of dependent line parameters is added.

In the right graph, if a spatial line coincides with more than one plane, this line is defined by the intersection of these planes. The points on this line give the plane equation (4.5) to each plane instead of the line equation (4.7). And the points P_a, P_b and P_c are considered as the point on the planes only. No line parameters exist in the computation.

4.5 BA Experiments Introduction

There are two experiments in the following sections. The first experiment deals with the acceptance ratio of adjustment results. It compares the conventional BA and the first new method. The performance of these two methods is evaluated by the acceptance ratio of the two-sided hypothesis test. The new method has a higher acceptance ratio.

The second experiment discusses the estimation accuracy and precision. The comparisons are among the conventional BA and two new methods. From the perspective of accuracy and precision, conventional BA and the new methods have the same performance. Spatial constraints reduce the error and variances of the estimations.

We now introduce the experiment environment. In the experiment, there are sixty frames. They are arranged with horizontal displacements as in figure (4.7). The leftmost frame is the first frame. The first camera coordinate system acts as the reference coordinate system. All the cameras are placed along the X-axis. The true position of $\mathrm{i}-th$ frame is $[\mathrm{i}-1, 0, 0]^T$. The distance between the adjacent two cameras equals 1, which is the unit of length in this experiment. The true rotation angle vector of each frame is $[0, 0, 0]^T$. During the iterations, the distance between the first frame and final frame is fixed. Then the seven coordinate system parameters are fixed.

The depth of the object points is a variable of the two experiments. The two experiments are tested under a series of depth-base ratios. The depth series is 15, 30, 45, 60, 75. This is compared to 1 unit for the between camera baseline. Object points are distributed within a narrow range about the nominal depth range. The depth is the mean of the vertical distance between the camera and object points.

Figure 4.20. Visualization of the distribution of cameras and points at depth 15.

The spatial configuration at depth 15 is demonstrated above. The small color triangles represent the cameras. The small dots represent the object points.

There are 256 object points which have no spatial constraints. Each camera observes 20 of this kind of object points. Each object point is observed by 2 to 5 frames.

Table 4.1. Summary of frames per point, maximum 5, minimum 2.

Frame Number	2	3	4	5
Point Number	16	8	8	224

The last column means that 224 object points are visible in 5 images. This distribution is kept the same for the two experiments.

There are 56 object points, which have spatial constraints. These object points are used in the second experiment only. There are four distance constraints (4.4). Each distance unknown employs three pairs of points. there are three spatial line constraints(4.7) with four points in each one. There are two spatial plane constraints(4.5). The first plane has eight points. The second one has twelve points. Among the twelve points, two two-dimensional lines(4.8) have four points each, and the other four are not assigned.

Zero-mean Gaussian noise is added to the image measurements, camera attitudes, and positions. The standard deviations of these errors are one pixel, 0.3 degrees, and 0.03 unit distance. The incorrect estimation ratio is tested under this condition in the figure(4.11) The level of error remains the same in the two experiments.

The two experiment results are evaluated by statistical techniques. The prerequisite background of least squares adjustment is introduced here. The table below includes the common notations.

Table 4.2. Notations of Least Squares Adjustment

Name	Expression
Measurement vector	$\mathbf{l}$
Measurement covariance	Σ_{ll}
Parameter vector	$\mathbf{x}$
Parameter covariance	Σ_{xx}
Wight Matrix	W
Residuals vector	$\mathbf{v}$
Redundancy	r
Prior variance	σ_0^2
Posterior variance	$\widehat{\sigma_0^2}$

The expression of posterior variance is the one below.

$$\widehat{\sigma_0^2} = \frac{\mathbf{v}^T W \mathbf{v}}{r} \quad \text{where} \quad W = \sigma_0^2 \Sigma_{ll}^{-1} \tag{4.9}$$

The adjustment computation and necessary expressions in the least squares methods are stated in Mikhail's(1976)[2] survey and adjustment book.

4.6 BA Experiment about Acceptance Ratio

The two tested methods are the conventional BA and the first new method. This experiment tests the acceptance ratio of computation results. The first new method uses the bilinear and trilinear equations and general adjustment model. In the adjustment, the $\mathbf{v}^T \Sigma_{ll}^{-1} \mathbf{v}$ has a Chi-square distribution.

$$\mathbf{v}^T \sum_{ll}^{-1} \mathbf{v} = \frac{\mathbf{v}^T W \mathbf{v}}{\sigma_0^2} = \frac{\widehat{\sigma_0^2}}{\sigma_0^2} r \sim \chi_r^2 \tag{4.10}$$

In the expression χ_r^2, r is the redundancy or, statistical degrees of freedom. This experiment performs a two-sided hypothesis test on the reference variance at a 0.05 level of significance. The hypotheses of this experiment is below.

$$\begin{aligned} H_0 &: \sigma^2 = \sigma_0^2 \\ H_1 &: \sigma^2 \neq \sigma_0^2 \end{aligned} \tag{4.11}$$

The σ_0^2 in this experiment is the variance chosen for the measurement error. If the test statistic $\frac{r\widehat{\sigma_0^2}}{\sigma_0^2}$ is between the two-sided 95 percent critical values , the experiment accepts H_0 or rejects it otherwise.

Under the same error and depth condition, the test is repeated one thousand times. The final acceptance ratio r_{ap} of the two methods will be compared.

$$r_{ap} = \frac{\text{the number of acceptance cases}}{\text{the total number of test cases}} \tag{4.12}$$

The numbers of equations, unknowns, and redundancy of the two methods are compared here.

Table 4.3. Numbers of two methods

	Conventional BA	New method
Camera unknowns	353	353
Object point unknowns	768	0
Total unknowns	1121	353
Equations	2416	1648
Redundancy	1295	1295

As mentioned before, the small intersection angle issue will cause incorrect estimates of object points. An incorrect estimate means that the estimated object point appears behind any camera. During the one thousand trials, the random errors are added to the measurements and true values of camera parameters. Each initial value of the object point is computed from the current state. If the incorrect estimate appears, the initial condition of this trial is called a bad initial condition. Otherwise, the trial has a good initial condition. The total condition includes both of them.

Table 4.4. Three initial conditions

Condition	Meaning
Good	No incorrect object point estimates
Bad	Any object point appears behind any cameras
Total	All of them

$$\text{The bad condition ratio} = \frac{\text{the number of Bad cases}}{\text{the total number of cases}} \tag{4.13}$$

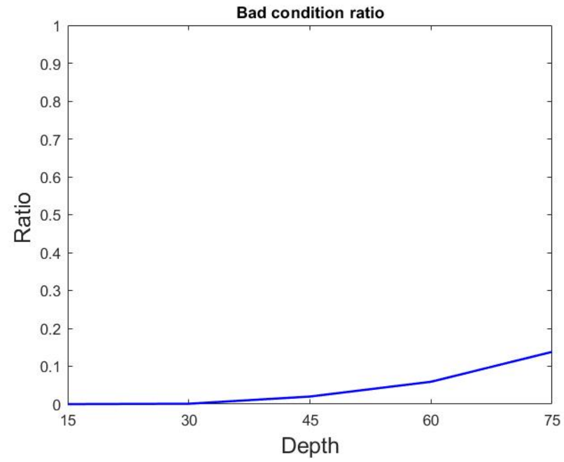
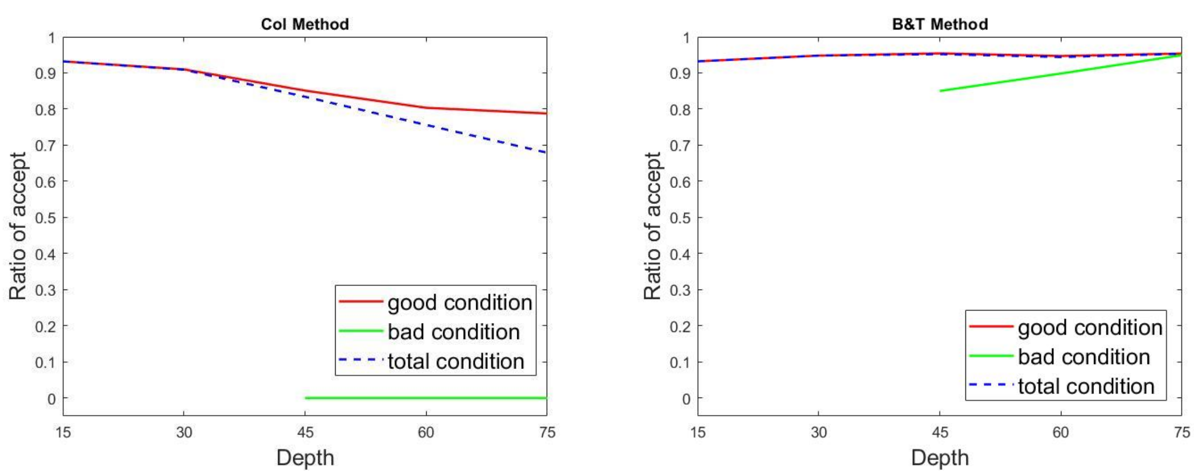


Figure 4.21. Bad condition ratio

With the growth of the distance, the frequency of the bad condition increases. The charts below show the result of the experiment. For each depth level, the total number of cases is 1,000. The numbers of bad cases are 0, 1 , 20, 59 and 138. The bad condition has not enough cases at depth 15 and 30 to give a meaningful ratio. The plots of bad condition start from depth 45.



(a) Result of conventional BA.

(b) Result of new method.

Figure 4.22. Comparison of acceptance ratio

In the legend of the chart, 'Col Method' means the conventional method. The 'BT Method' means the new method. The right plot shows that under any conditions, the ratios of new method are around 95 percent. The left plots indicates that the performance changes as the depth increases for the conventional method. The individual comparisons of the two methods under the good, bad, and total conditions are displayed below.

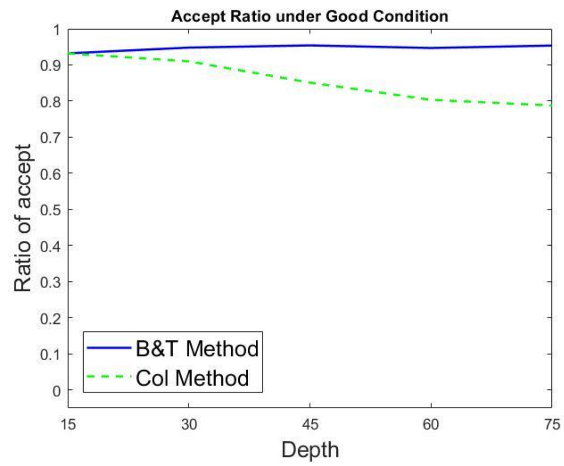


Figure 4.23. Compare the ratio under good condition

When the depth increases, under the good initial condition, the new method has a higher acceptance ratio. Its ratio has a little variety. With the growth of the distance, the acceptance ratio of the conventional method goes lower. Given reasonable estimates, the computations are still impacted by the initial values. Reasonable estimates here means that all the object points are located in front of all cameras.

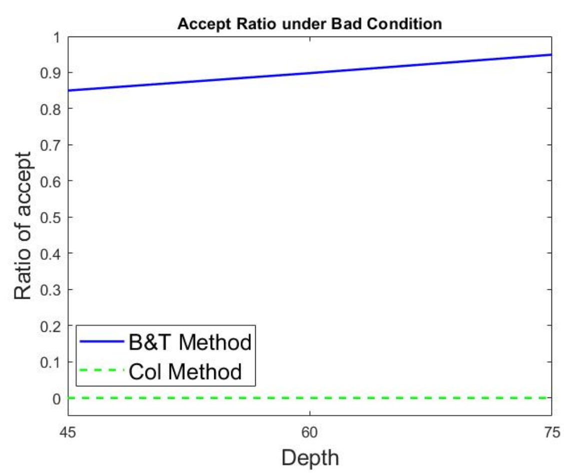


Figure 4.24. Compare the ratio under bad condition

The acceptance ratio of the bad condition states that the incorrect estimates make the conventional method fail. The initial values impact this method a lot. The method employing bilinear and trilinear equations does not require the initial estimates of object points.

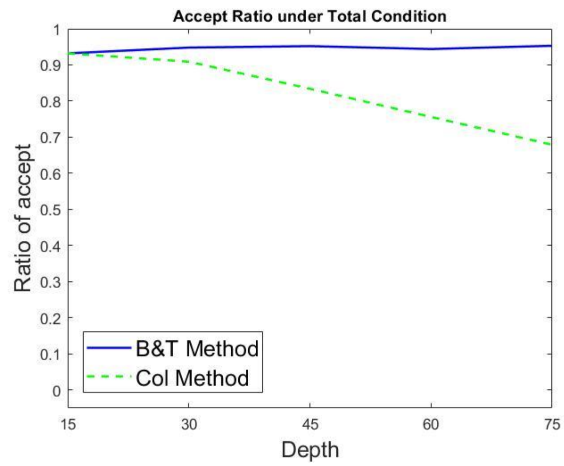


Figure 4.25. Compare the ratio under total condition

Then under the all condition, the new method has a higher acceptance ratio than the other. When the depth is small, the small intersection angle issue is not present, so the incorrect estimates are not the factor to worry about. The two methods have the same acceptance ratio. When the depth is large, the incorrect estimates caused by weak geometry become the important factors. If the experiment is tested under the forward movement type, this difference will be even more obvious.

There are two criteria for stopping the iterative computation. The first one is that the update is small enough. The second one is that computation has enough iterations. After the computation is stopped, it gives an estimated value $\frac{\widehat{\sigma_0^2}}{\sigma_0^2}r$. The number of iterations of the computations, which pass the two-sided hypothesis test, range from 6 to 13. The threshold for max iterations is 30 . So the lower acceptance ratio is a numerical problem caused by the unstable initial estimates. This experiment shows that the new method has a better performance considering overcoming weak geometry.

4.7 BA Experiment about Accuracy and Precision

This experiment tests the accuracy and precision of three methods. They are one conventional method and two new methods. The first new method uses the bilinear and trilinear equations and the general adjustment model. The second new method uses mixed collinearity, bilinear and trilinear equations and constraints. These methods are evaluated by the error and variance of camera exterior orientation parameters(EOP). In comparisons, these parameters are separated into position and rotation parameters.

There are three comparisons among these methods. The first comparison between the conventional method and first new one shows the properties of different condition equations. The second comparison between the two new methods can illuminate the role of the spatial constraint. The third comparison discusses the properties of the close and the far object points.

There are three conclusions from those comparisons. Firstly, from the perspective of accuracy and precision, the conventional method and the first new method have the same accuracy and precision. Secondly, spatial constraints reduce the error and variances of the estimations. Thirdly, both the close object points and far object points contribute to the computation significantly.

The accuracy is evaluated by the Euclidean distance between the estimated values and the real simulated values

$$d(\widehat{x}, x) = \|\widehat{x}x\|.$$

The errors of position and rotation are the Euclidean distance of the position and rotation vectors,

$$d_{pos}(\widehat{x}_{pos}, x_{pos}) = \|\widehat{x}_{pos}x_{pos}\|,$$

and

$$d_{rot}(\widehat{x}_{rot}, x_{rot}) = \|\widehat{x}_{rot}x_{rot}\|.$$

The parameter variances are estimated using the cofactor matrix and the estimate of the posterior variance.

$$\Sigma_{xx} = Q_{xx}\widehat{\sigma_0^2}.$$

In the above expression, Q_{xx} is the cofactor matrix of parameters. The variance of the position and rotation are a composite of the variances position and rotation vector elements,

$$\sigma_{pos}^2 = \sigma_x^2 + \sigma_y^2 + \sigma_z^2,$$

and

$$\sigma_{rot}^2 = \sigma_1^2 + \sigma_2^2 + \sigma_3^2.$$

The following section will compare the three methods using the d_{pos}, d_{rot}, σ_{pos}^2 and σ_{rot}^2.

The usage of the constraints increases the redundancy. The redundancy without constraint equations is

$$r = n_c - n_u,$$

where n_c and n_u are the numbers of condition equations and unknowns.

The redundancy with constraint equations is

$$r = n_c + n_s - n_u - n_q,$$

where n_s and $n_q$are the numbers of constraints and added unknowns. The added unknowns are the parameters of the spatial features.

The table below states the redundancies of three instances in this experiment. The numbers of equations, unknowns, and redundancy of the three methods are shown here.

Table 4.5. Numbers of three methods

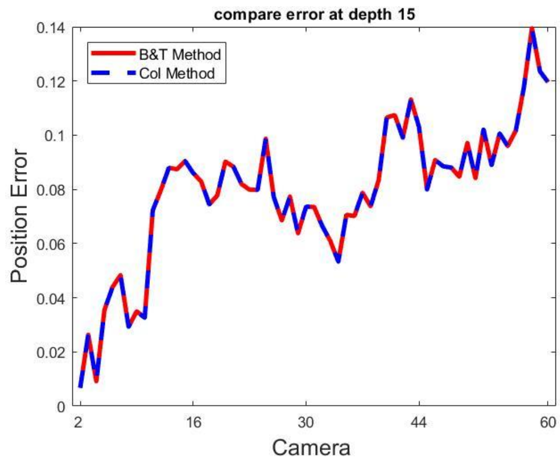
	Conventional BA	New One	New Two
Camera unknowns	353	353	353
Object point unknowns	936	0	168
Added unknowns	0	0	26
Total unknowns	1289	353	547
Condition Equations	3072	2136	2304
Constraint Equations	0	0	64
Redundancy	1783	1783	1821

The three adjustments are tested as before at the 0.05 significance level. If the three test statistics fall between the critical values, the result will be accepted. Under each depth level, these methods are repeated until all of them are accepted simultaneously 50 times. The final error values and variances are the average values of these 50 times.

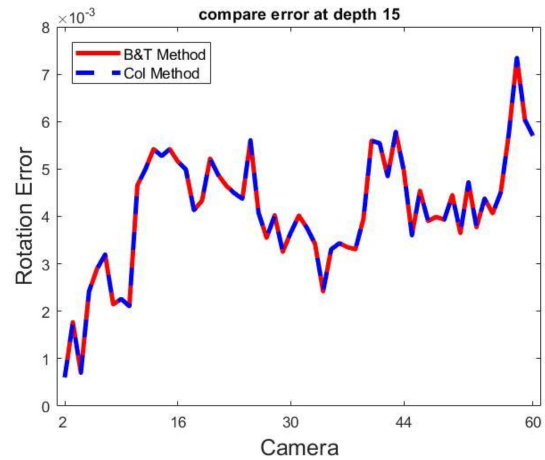
4.7.1 Comparison One

The first comparison is between the conventional method and the first new method. The differences between them are the condition equations and adjustment models. In the legend of the chart, ‘Col Method’ means the conventional method. The ’B T Method’ represents the new method.

s



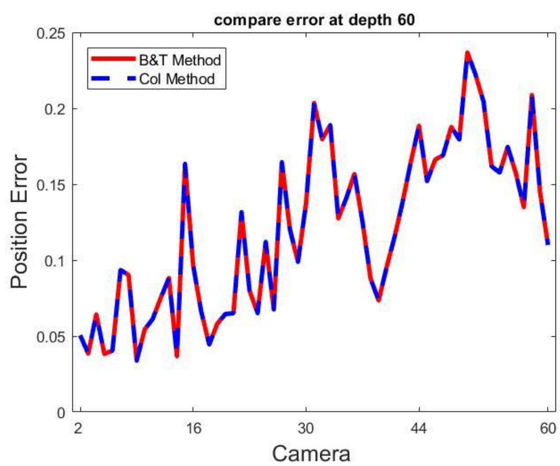
(a) d_{pos} from two methods



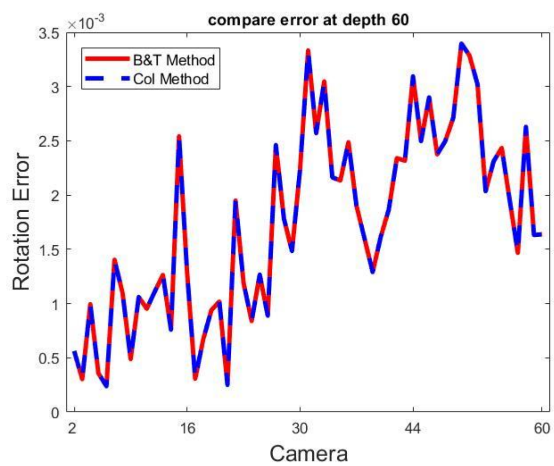
(b) d_{rot} from two methods

Figure 4.26. First Comparison: d_{pos} and d_{rot} at depth 15

In figure(4.26) are comparisons of the d_{pos} and d_{rot} at depth 15. The left one and right one are the errors of position and rotation respectively. The horizontal axis represents the camera sequence from the second camera to the last camera. The vertical axis is the error quantity.



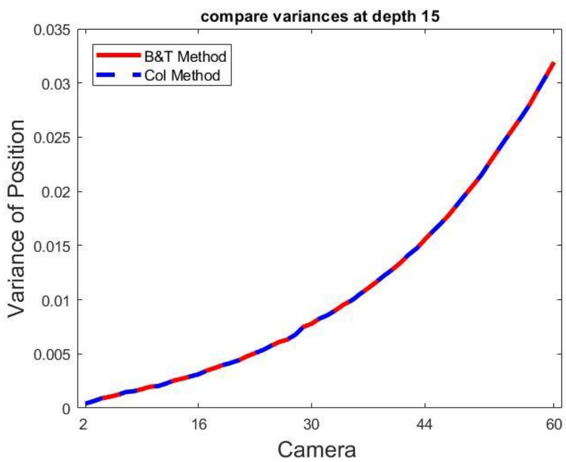
(a) d_{pos} from two methods



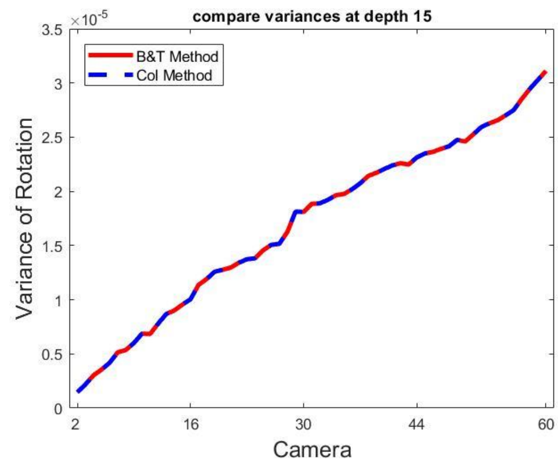
(b) d_{rot} from two methods

Figure 4.27. First Comparison: d_{pos} and d_{rot} at depth 60

In figure(4.27) are comparisons of the d_{pos} and d_{rot} at depth 60. The first camera is fixed. The errors increase as the frame sequence increases as it moves from left to right. At different depths, the error plots from the two methods coincide exactly.



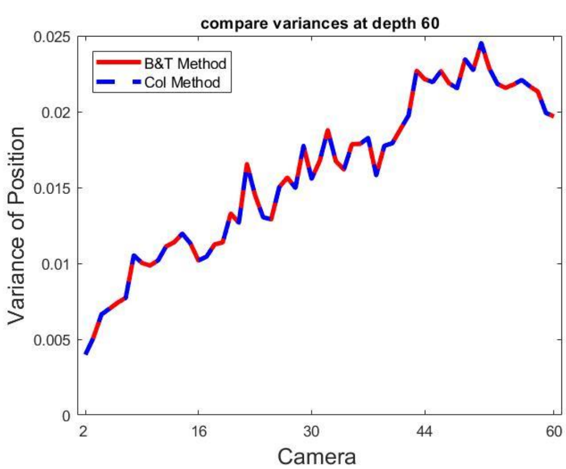
(a) σ^2_{pos} from two methods



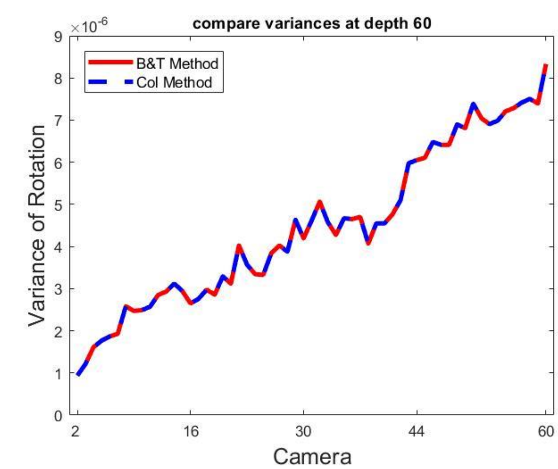
(b) σ^2_{rot} from two methods

Figure 4.28. First Comparison: σ^2_{pos} and σ^2_{rot} at depth 15

In figure(4.28) are the comparisons of σ^2_{pos} and σ^2_{rot} at depth 15.



(a) σ^2_{pos} from two methods



(b) σ^2_{rot} from two methods

Figure 4.29. First Comparison: σ^2_{pos} andσ^2_{rot} at depth 60

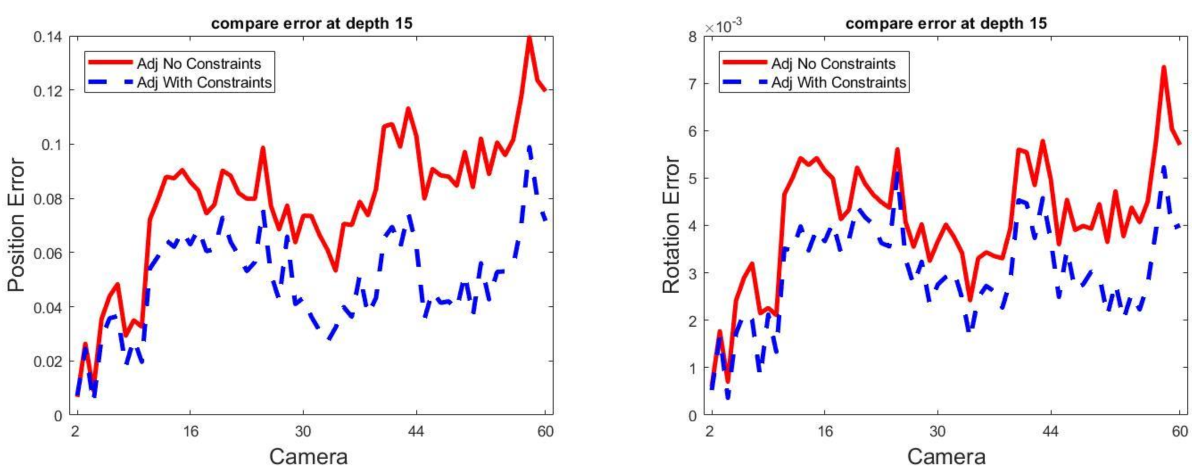
In figure(4.29) are the comparisons of σ^2_{pos} and σ^2_{rot} at depth 60, again no differences here.

The results show that the variances increase with increasing frame number as it moves away from the first camera. At different heights, both the error and the variance plots from two methods exactly coincide. There is no difference between the two methods in terms of accuracy and precision from the statistical results.

So, the conversion of the condition equations and adjustment models has no impact on the computation results. The TFM modelling achieves the same results as the collinearity modelling. This is a demonstration of the correctness of the TFM model. So the two condition equations may be used together. In the following section, the comparison is between a constrained versus unconstrained adjustment using TFM.

4.7.2 Comparison Two

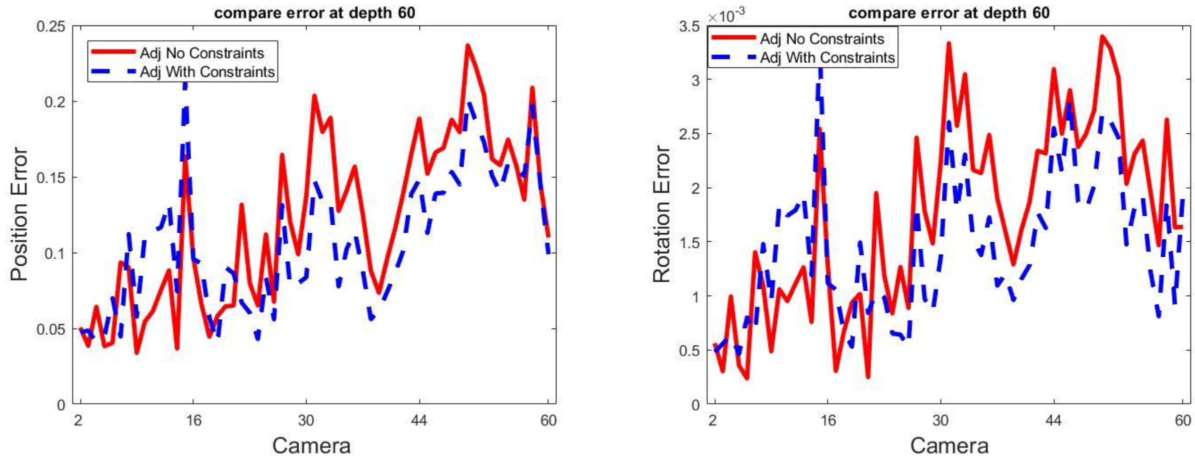
The next comparison is between the two new methods to demonstrate the advantages of spatial constraints. The difference between them is whether spatial constraints are used. In the legend of the chart, ‘Adj no constraint’ means the unconstrained method. The ’Adj with constraints’ represents the constrained, new method.



(a) d_{pos} from two methods (b) d_{rot} from two methods

Figure 4.30. Second Comparison: d_{pos} and d_{rot} at depth 15

In figure(4.30) are comparisons of the d_{pos} and d_{rot} at depth 15.



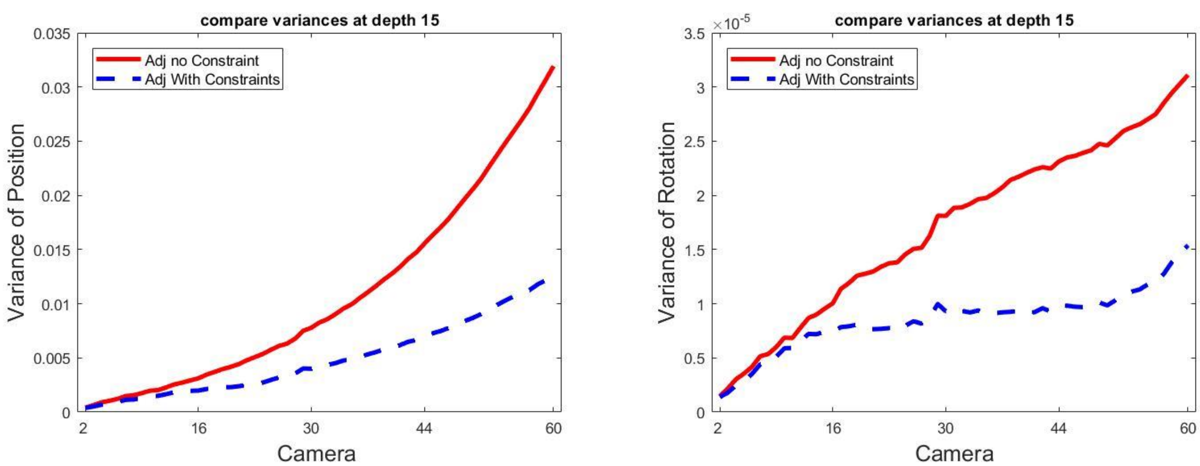
(a) d_{pos} from two methods

(b) d_{rot} from two methods

Figure 4.31. Second Comparison: d_{pos} and d_{rot} at depth 60

In figure(4.31) are comparisons of the d_{pos} and d_{rot} at depth 60.

The plots show that the constraints have a beneficial influence on the computation. At the relatively small depth, all the estimation errors are reduced by the presence of constraints in figure (4.30). At a relatively larger depth, most of the errors are reduced in figure (4.31).

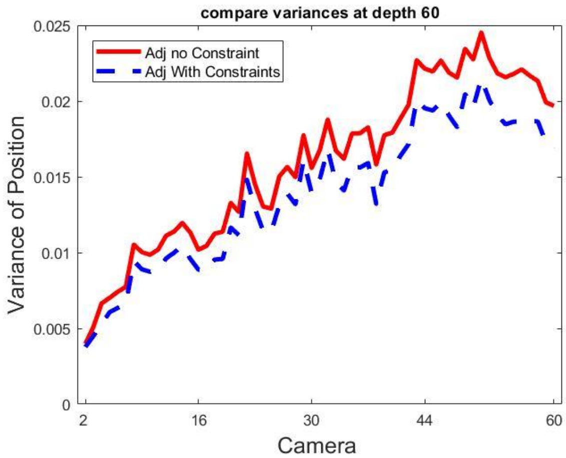


(a) σ^2_{pos} from two methods

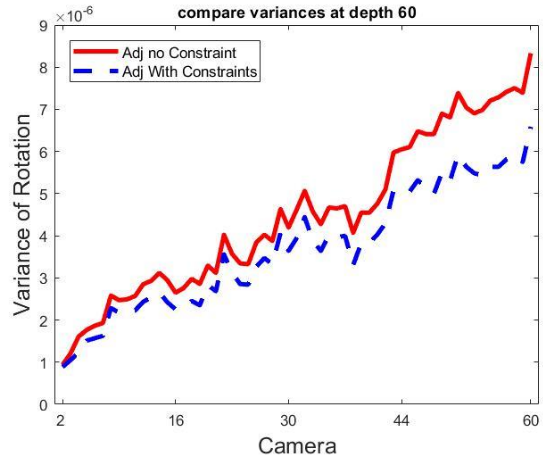
(b) σ^2_{rot} from two methods

Figure 4.32. Second Comparison: σ^2_{pos} and σ^2_{rot} at depth 15

In figure(4.32) are comparisons not of position and rotation errors, but of their uncertainties σ^2_{pos} and σ^2_{rot} at depth 15.



(a) σ^2_{pos} from two methods



(b) σ^2_{rot} from two methods

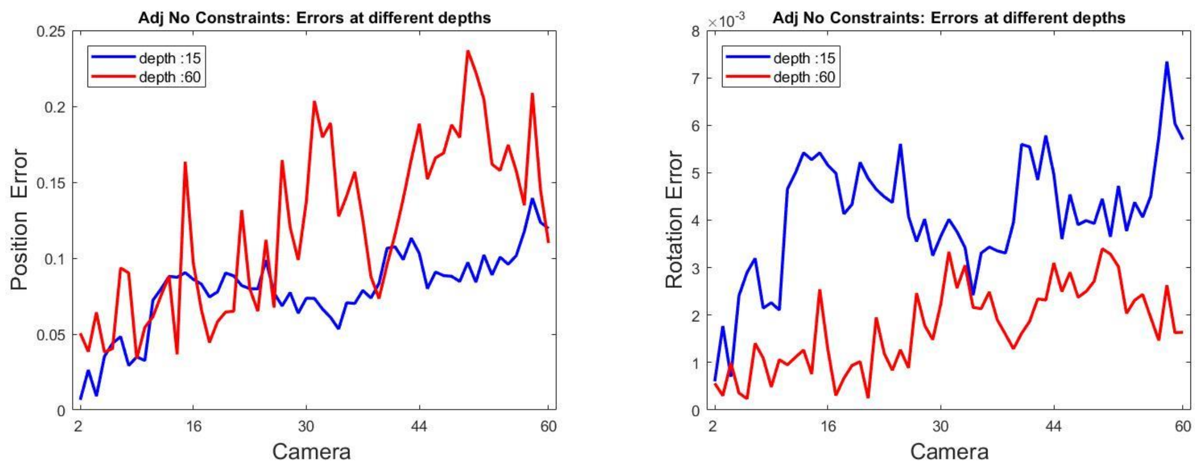
Figure 4.33. Second Comparison: σ^2_{pos} and σ^2_{rot} at depth 60

In figure(4.33) are the same comparisons of the σ^2_{pos} and σ^2_{rot} at depth 60.

At different heights, the variances with constraints are always lower than the ones without the constraints. When the depth is relatively small, the constraints have a more noticeable effect. At the relatively large depth, the influence is reduced.

4.7.3 Comparison Three

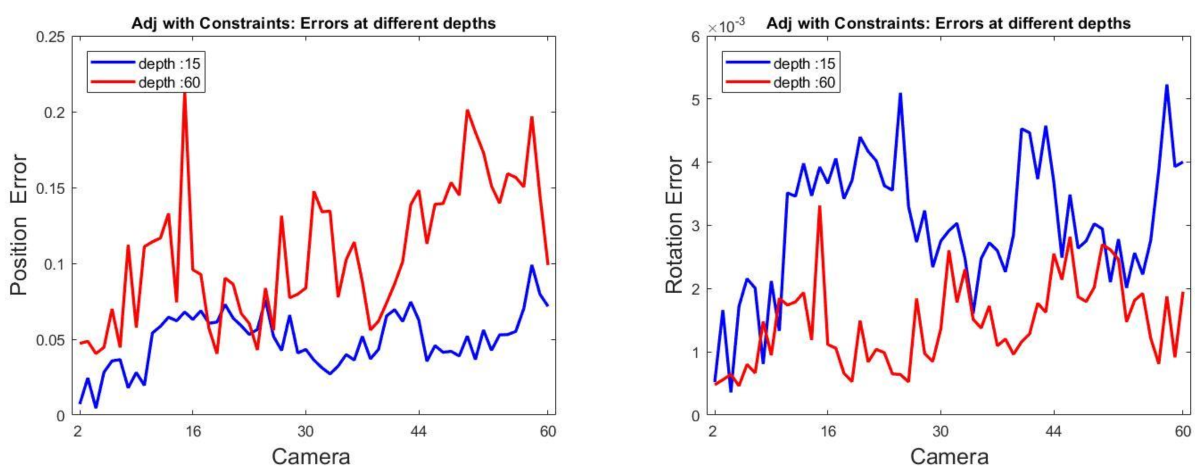
The third comparison presents the performance of the two new methods under different depths individually.



(a) d_{pos} at two depths. (b) d_{rot} at two depths.

Figure 4.34. Third Comparison: d_{pos}, d_{rot} from the unconstrained method

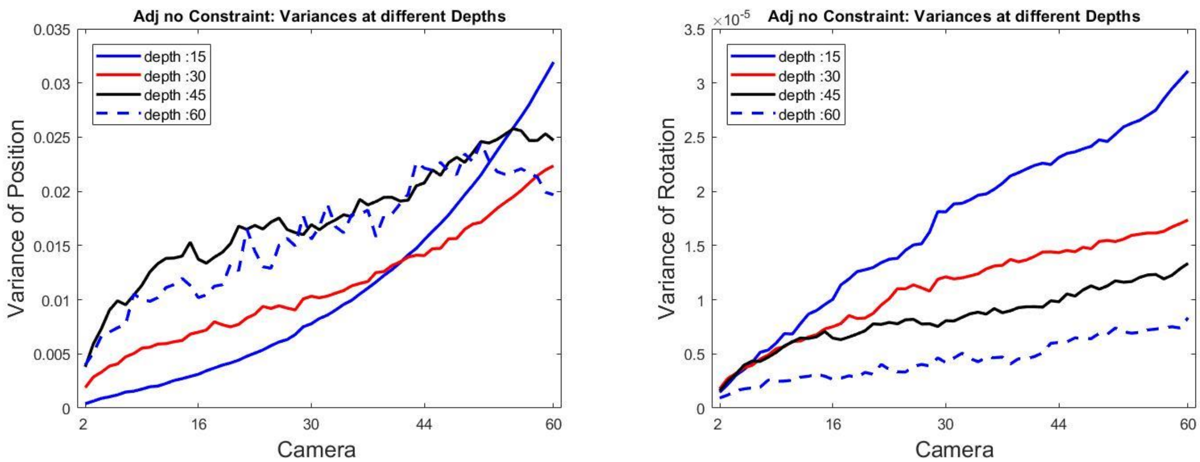
In figure(4.34) are presentations of d_{pos} and d_{rot} at different depths, always with no constraints. The two depths represent the close and far object point scenarios. The above plots show that the estimation of the position weakens as the depth increases, but the estimation of rotation is just the opposite.



(a) d_{pos} at two depths. (b) d_{rot} at two depths.

Figure 4.35. Third Comparison: d_{pos}, d_{rot} from the constrained method

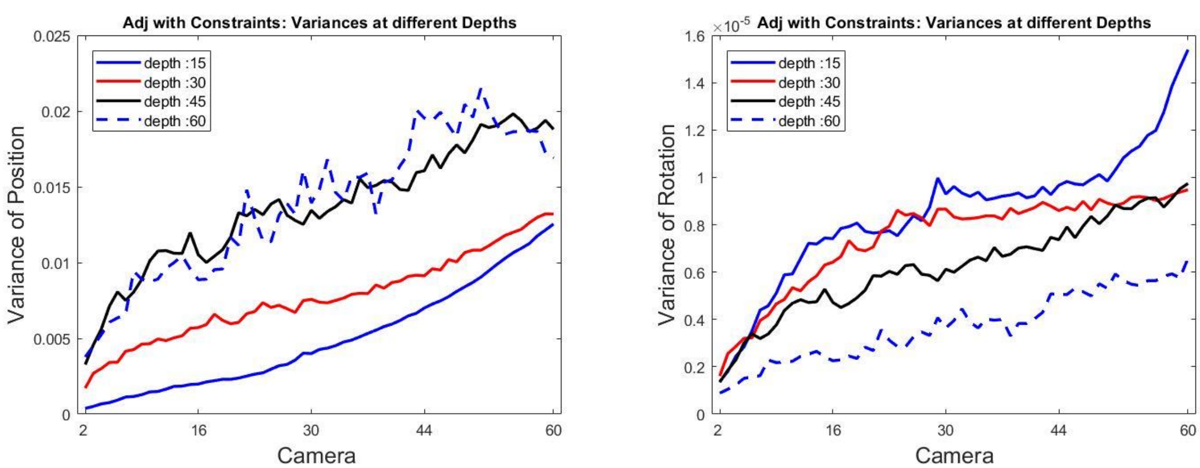
In figure(4.35) are the same comparisons of d_{pos} and d_{rot} but this time with the constrained method at depth 15 and 60. Compared to the previous plots, the conclusion is not changing after adding the constraints. The two position error graphs are very similar, and so are the two rotation error graphs.



(a) σ_{pos}^2 at four depths. (b) σ_{rot}^2 at four depths.

Figure 4.36. Third Comparison: σ_{pos}^2, σ_{rot}^2 from the unconstrained method

In figure(4.36) are comparisons of the σ_{pos}^2 and σ_{rot}^2 from the unconstrained method at four different depths.



(a) σ_{pos}^2 at four depths. (b) σ_{rot}^2 at four depths.

Figure 4.37. Third Comparison: σ_{pos}^2, σ_{rot}^2 from the constrained method

In figure(4.37) are comparisons of the σ_{pos}^2 and σ_{rot}^2 from the constrained method at four different depths.

The two plots of σ^2_{pos} show that the position variance increases with depth increase. The two plots of σ^2_{rot} show that the inverse rule applies to the rotation variance.

As the depth increases, the errors and variances of positions will increase, but the errors and variances of rotations decrease. This phenomenon of the estimation error and variances means that the close points enhance the estimation of camera positions, and the far points improve the estimation of camera rotation. The significant role of far points is also shown in Schneider's(2012)[122] experiment. The close point scenario gives a much more stable position estimation but a weaker rotation estimation than the far point scenario. So it is needed to keep both of the far object points and the close points. Elimination of the far object points in the adjustment will decrease the estimation quality.

4.8 Real Data Experiment

A real data experiment includes camera calibration and BA algorithm testing in the following section.

4.8.1 Camera Calibration

The camera calibration has two steps and uses a printed checkerboard target array. The first step is the initial linear estimation of the elements of the K matrix. Zhang's(2000)[126] algorithm is used to realize it. The second step is the nonlinear estimation of the camera interior orientation parameters and lens distortion parameters using the conventional BA algorithm. In the experiment, the conventional radial distortion and decentering distortion of Alharthy and Bethel(2002)[93] are considered. The radial distortion needs three parameters (k_1, k_2, k_3). Decentering distortion needs two parameters (p_1, p_2). The size of the sensor in the camera is 4032 pixels by 3024 pixels. There are 47 images of the template used in the camera calibration. Each image has 80 detected Harris corner points.

Note that there may be issues with calibration taking place at one focus distance and camera use taking place at different focus setting. The autofocus feature makes this difficult to control.

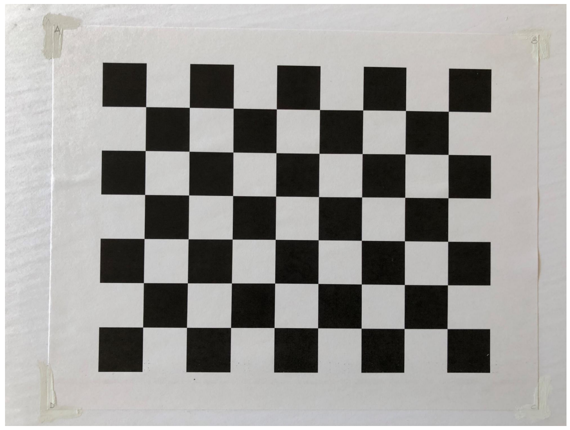


Figure 4.38. One image of Template

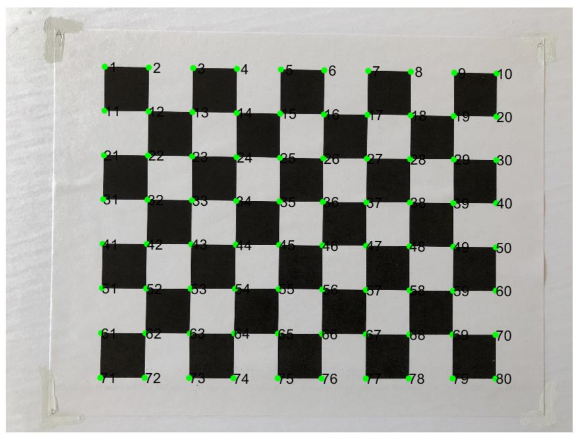


Figure 4.39. Detected corners

In the figure above, the green points are the detected corners using Harris's(1988)[127] algorithm.

The estimated K matrix is the one below.

$$\begin{bmatrix} 3349.39 & -1.35 & 2011.73 \\ 0 & 3357.60 & 1518.52 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.14}$$

The initial value of the focal length is 3353.50, which is the average of 3349.39 and 3357.60. The initial values of the $(k_1 k_2, k_3)$ and (p_1, p_2) are zero. The estimated values and standard deviations are in the below table.

Table 4.6. Camera Calibration Result

parameter	x_0	y_0	f	k_1	k_2	k_3	p_1	p_2
value	2018.39	1505.86	3342.00	-0.0609	0.146	-0.110	0.777	-0.302
σ	1.91	1.84	2.67	0.0012	0.0033	0.0027	0.34	0.32

These parameters are used to rectify the image measurements in the work below.

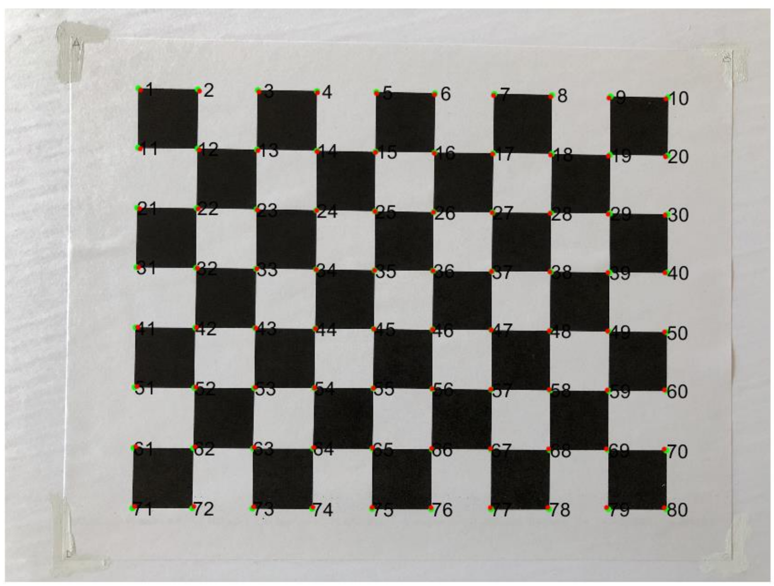


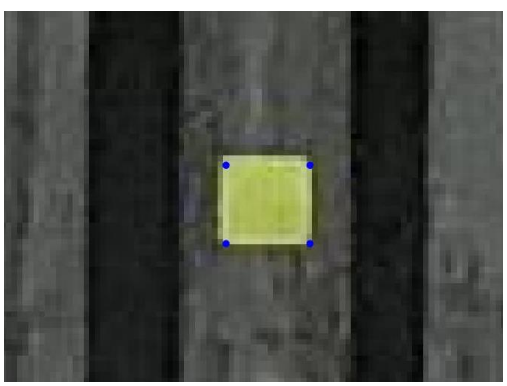
Figure 4.40. Calibrated image points

In the figure above, the green points are the detected corner points. And the red points are the rectified ones. The largest correction in this example is around 15 pixels in the figure above. The corrections become large when points are close to the boundaries.

4.8.2 BA Algorithm Testing

This experiment involves 15 images and 24 object points. The 24 object points are visible from all images. The camera moves forward while taking the photos. The first images and object points are displayed below. The observations are marked by blue points in the images. Four rectangles are used and their corner points are measured.

Figure 4.41. View of testing field



(a) One rectangle with measured corner points.

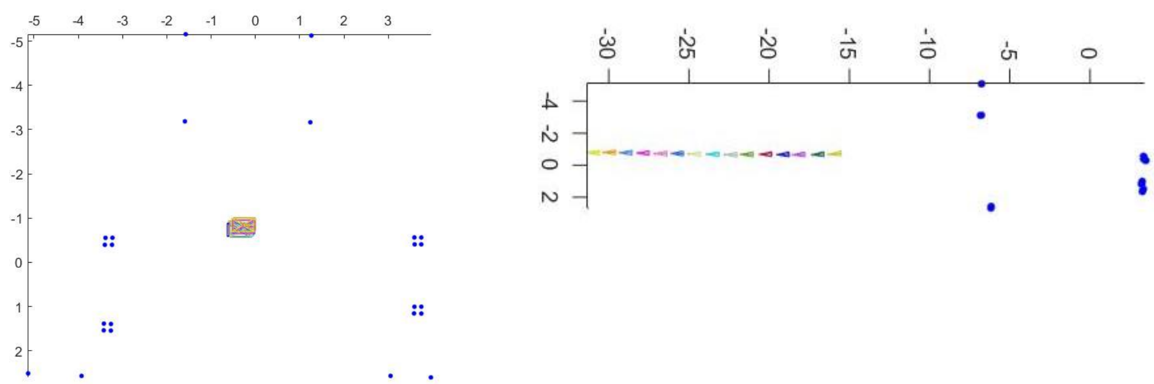
(b) Corners of a plane.

Figure 4.42. Object points distribution

All of the image points are measured manually. The image points are rectified and normalized. So only camera positions and rotations are considered in the experiment.

The coordinate system of the first camera is fixed as the reference coordinates system. Every other frame's initial position and rotation values are computed using the image-pair method with respect to the first camera. Each essential matrix E is linearly estimated using the eight-point algorithm. The relative position $\widetilde{\mathbf{C}}_{\mathrm{i}}$ and rotation R are retrieved from the essential matrix and optimized nonlinearly. The relative distance $\|\widetilde{\mathbf{C}}_{\mathrm{j}}\|/\|\widetilde{\mathbf{C}}_{\mathrm{i}}\|$is estimated using the designed method in a previous section. These three steps prepare the initial values for each camera for the following test.

The two designed BA methods(constrained and unconstrained) are tested individually. The figures below show the reconstructed objected points and the camera movement.



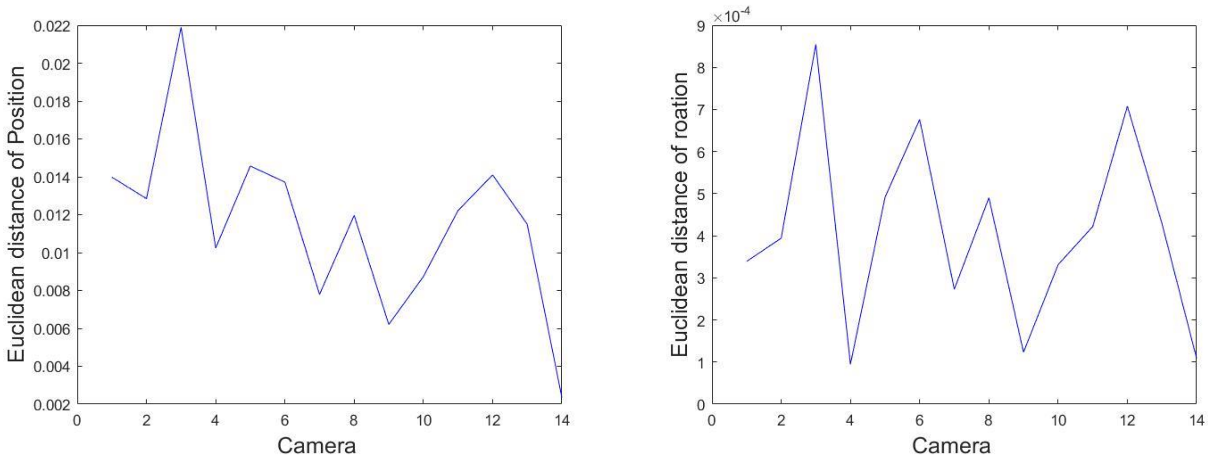
(a) View from front.

(b) View from the side.

Figure 4.43. Reconstruction and camera movement (camera movement clearly shown on the plot on the right)

The small colored triangles represent the cameras. The blue points represent the object points. The distance between the first two cameras is set as 1 unit distance. The computed distance between the rectangles and first camera is around 34. The true size of the rectangles is 7.6 cm by 7.6 cm.

The second test involves BA with spatial constraints. The distance constraint is employed for the length of the rectangle sides. And the plane constraint is engaged for the four corner points on the basketball backboard. The differences brought by the constraints are evaluated by the Euclidean distances between the two results, in figure(4.44).



(a) Distance of position. (b) Distance of rotation.

Figure 4.44. Differences brought by spatial constraints

There are no dramatic conclusions from these plotted differences, just that the new method appears to work in both constrained and unconstrained form using the new TFM methods. An instructive additional step would have been to compare estimated lengths to measured lengths to get an absolute indication of the benefit of the constraints. But this was not done.

5. SUMMARY And FUTURE WORK

This chapter gives the conclusion of the thesis and some recommended future work.

5.1 Summary

This study has three parts. The first one is the derivations and the analyses of the multi-frame geometry. The second part has the two new TFM-based BA methods. The third part is the collection experiments to illustrate the important contributions of the study.

In the first part, this thesis gives the nested derivations and analyses of multi-frame geometry. There are three different approaches given in the derivation and analysis section. These approaches are the matrix determinant, spatial geometry, and column vector approaches. The zero-determinant condition is used for the derivations of the trilinear and quadrilinear equations. But a quadrilinear equation is not found in this way. And it shows that the two-frame and three-frame conditions will guarantee the four-frame cases. The spatial geometry approach starts from the two-frame case to the multi-frame case. It explains some spatial properties of the trilinear equations, including the cross product and the inner product. It also indicates the minimal and sufficient equations in three-frame geometry. Finally, this approach promotes a new TFM and extends it from three-frame geometry to the general multi-frame cases. This TFM provides a concise set of minimal and sufficient equations, including $(n-1)$ bilinear equations and $(n-2)$ trilinear equations for n frames. The column vector approach studies and analyzes the dependence of the three types of equation. It starts from single-frame projections to the multi-frame case. For each type of equation, it gives the way to find out redundant equations. But the closed-form expressions of those equations are not displayed in the thesis. Finally it leads to revisiting the many conclusions from the other two approaches. For example, it also indicates that the quadrilinear equations are dependent on the trilinear equations.

In the application section, there are two TFM-based BA methods developed. The two methods use TFM as the condition equation fully and partially, replacing the collinearity equations. The second method uses the collinearity equations to exploit the object structures' spatial characteristics, i.e. by constraints for lines, planes, and point distances. Two other two applications, image coordinate prediction, and relative distance estimation, are designed and tested, the first one in the context of fisheye imaging.

Synthetic and real data experiments demonstrate the functionality, and validity, and some advantages of the TFM and the two TFM-based methods. The conclusions are summarized here. When an unstable estimate of the object points appears, the TFM-based BA methods have a higher acceptance ratio of the adjustment results. The TFM-based BA method achieves the same ability as the collinearity-based BA method with respect to accuracy and precision. And utilizing the spatial constraints promotes improved estimations. The experiment also shows that close object points give a more stable position estimation but a weaker rotation estimation than distant object points. So eliminating the distant object points in the adjustment will decrease the estimation quality. And the TFM, in general, handles distant points in an improved manner versus explicit object point methods.

5.2 Summary of Original Contributions of this Research

1. New derivations of the trilinear equations.

This paper gives two new derivations of the trilinear equations for the three-frame and four-frame geometry.

2. A new analysis of multi-frame geometry.

The spatial geometry analysis approach gives a new TFM for multi-frame geometry. The column vector approach shows that the quadrilinear equations are linear combinations of the trilinear equations.

3. This thesis gives two algorithms for conventional applications.

In the image coordinate prediction algorithm, the new method resolves the ambiguity in the wide-field camera. The relative distance estimation algorithm works well for close and distant object points.

4. There are two BA algorithms using bilinear and trilinear equations different from other researchers.

This thesis implements the new TFM model in the BA algorithm. Another new BA method, which combinates the TFM and collinearity equations with spatial constraints, is given in the thesis. The research discovers that the TFM-based BA algorithm has a higher acceptance ratio than the conventional method.

5.3 Future Work

There is still some necessary work should be done. This includes the theoretical model study and pratical algorithm developments.

The TFM should integrate the line feature(without object points). The point-point-line, point-line-line, and line-line-line relations are not addressed in this thesis. These relations should also be considered under the multi-frame cases to give minimal and sufficient equations. This is a future work of the model study.

In order to make the two TFM-based BA methods become realistic and comprehensive, some necessary functions should be added to the current techniques, such as self-calibration and free network adjustment. Up to now, only the camera positions and rotations are considered. The camera interior orientation parameters and lens distortion parameters should be integrated into the TFM and BA methods. The free network adjustment technique is a necessary function for close-range photogrammetry. In the current algorithms, the reference system is fixed to the first camera. This should be improved for practical applications.

The TFM has some advantages in different implementing environments. There are four particular situations which fit with the strengths of the TFM methods.

Firstly, the TFM works on the image points at infinity, whose third component equals zero. This kind of point may be common for a large view angle camera. The tiny third component makes the proportion in the collinearity equation numerically unstable. Any future TFM should work on the fisheye camera model.

The second advantage is memory saving. The unknowns of the object structure parameters are saved using the TFM, irrespective of being point or line features. And the TFM saves three condition equations for each point. In conventional computations, the size of the object feature parameters is much larger than the size of the camera parameters. When the number of frames is vast, the advantage of saving unknowns and condition equations will be obvious. After the BA computation, the object structures could be reconstructed from optimized camera parameters and observations.

Thirdly, the trilinear equation deals with the line feature easily. The line features will increase the number of the condition equations. The line-line-line, point-line-line equations could give the condition equation even for the images with no overlap!

Finally, the TFM has the potential advantage in the vision-based real-time orientation such as SLAM, as the TFM is more stable when unstable object structures appear. It seems promising to implement the TFM in such applications.

REFERENCES

[1] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Second. 2004.

[2] E. M. Mikhail, *Observations and least squares*. IEP, 1976.

[3] E. M. Mikhail, J. S. Bethel, and J. C. McGlone, *Introduction to modern photogrammetry*. 2001.

[4] B. Triggs, “Optimal estimation of matching constraints,” in *European Workshop on 3D Structure from Multiple Images of Large-Scale Environments*, Springer, 1998, pp. 63–77.

[5] B. Triggs, P. McLauchlan, R. I. Hartley, and A. Fitzgibbon, “Bundle adjustment – a modern synthesis,” in *VISION ALGORITHMS: THEORY AND PRACTICE, LNCS*, Springer Verlag, 2000, pp. 298–375.

[6] M. Cooper and P. Cross, “Statistical concepts and their application in photogrammetry and surveying,” *The Photogrammetric Record*, vol. 12, no. 71, pp. 637–663, 1988.

[7] H. C. Longuet-Higgins, “A computer algorithm for reconstructing a scene from two projections,” *Nature*, vol. 293, no. 5828, pp. 133–135, 1981.

[8] R. Y. Tsai and T. S. Huang, “Uniqueness and estimation of three-dimensional motion parameters of rigid objects with curved surfaces,” *IEEE Transactions on pattern analysis and machine intelligence*, no. 1, pp. 13–27, 1984.

[9] M. Werman and A. Shashua, “The study of 3d-from-2d using elimination,” in *Proceedings of IEEE International Conference on Computer Vision*, IEEE, 1995, pp. 473–479.

[10] Q.-T. Luong and O. D. Faugeras, “The fundamental matrix: Theory, algorithms, and stability analysis,” *International journal of computer vision*, vol. 17, no. 1, pp. 43–75, 1996.

[11] R. I. Hartley, “In defense of the eight-point algorithm,” *IEEE Transactions on pattern analysis and machine intelligence*, vol. 19, no. 6, pp. 580–593, 1997.

[12] G. Sparr, “A common framework for kinetic depth, reconstruction and motion for deformable objects,” in *ECCV (2)*, 1994, pp. 471–482.

[13] R. I. Hartley, "Projective reconstruction and invariants from multiple images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 10, pp. 1036–1041, 1994.

[14] Q.-T. Luong, R. Deriche, O. Faugeras, and T. Papadopoulo, "On determining the fundamental matrix: Analysis of different methods and experimental results," PhD thesis, Inria, 1993.

[15] R. Deriche, Z. Zhang, Q.-T. Luong, and O. Faugeras, "Robust recovery of the epipolar geometry for an uncalibrated stereo rig," in *European Conference on Computer Vision*, Springer, 1994, pp. 567–576.

[16] P. H. Torr and D. W. Murray, "The development and comparison of robust methods for estimating the fundamental matrix," *International journal of computer vision*, vol. 24, no. 3, pp. 271–300, 1997.

[17] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[18] Z. Zhang, "Determining the epipolar geometry and its uncertainty: A review," *International journal of computer vision*, vol. 27, no. 2, pp. 161–195, 1998.

[19] O. Faugeras and L. Robert, "What can two images tell us about a third one?" *International Journal of Computer Vision*, vol. 18, no. 1, pp. 5–19, 1996.

[20] A. J. McNair, "Triplets: A basic unit for analytical aerotriangulation," *Photogrammetria*, vol. 19, pp. 367–380, 1962.

[21] M. Keller, "A practical three-photo orientation solution to the analytic aerotriangulation problem," *Photogrammetria*, vol. 22, no. 4, pp. 117–125, 1967.

[22] A. W. Fitzgibbon and A. Zisserman, "Automatic camera recovery for closed or open image sequences," in *European conference on computer vision*, Springer, 1998, pp. 311–326.

[23] Y. Liu and T. S. Huang, "Estimation of rigid body motion using straight line correspondences," *Computer Vision, Graphics, and Image Processing*, vol. 43, no. 1, pp. 37–52, 1988.

[24] E. M. Mikhail, "Use of triplets in analytical aerotriangulation," *Photogr. Eng*, vol. 28, pp. 625–632, 1962.

[25] J. M. Anderson and A. McNair, "Analytic aerotriangulation: Triplets and sub-blocks," *Photogrammetria*, vol. 21, no. 6, pp. 197–216, 1966.

[26] R. I. Hartley, "Euclidean reconstruction from uncalibrated views," in *Joint European-US workshop on applications of invariance in computer vision*, Springer, 1993, pp. 235–256.

[27] M. E. Spetsakis and Y. Aloimonos, "Closed form solution to the structure from motion problem from line correspondences.," in *AAAI*, 1987, pp. 738–743.

[28] M. E. Spetsakis and J. Y. Aloimonos, "Structure from motion using line correspondences," *International Journal of Computer Vision*, vol. 4, no. 3, pp. 171–183, 1990.

[29] Y. Liu and T. S. Huang, "A linear algorithm for motion estimation using straight line correspondences," *Computer Vision, Graphics, and Image Processing*, vol. 44, no. 1, pp. 35–57, 1988.

[30] J. Weng, T. S. Huang, and N. Ahuja, "Motion and structure from line correspondences; closed-form solution, uniqueness, and optimization," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 3, pp. 318–336, 1992.

[31] A. Shashua, "Trilinearity in visual recognition by alignment," in *European Conference on Computer Vision*, Springer, 1994, pp. 479–484.

[32] A. Shashua, "Algebraic functions for recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 8, pp. 779–789, 1995.

[33] R. I. Hartley, "Camera calibration using line correspondences," in *Proc. DARPA Image Understanding Workshop*, 1993, pp. 361–366.

[34] R. I. Hartley, "Lines and points in three views and the trifocal tensor," *International Journal of Computer Vision*, vol. 22, no. 2, pp. 125–140, 1997.

[35] M. E. Spetsakis, "A linear algorithm for point and line-based structure from motion," *CVGIP: Image Understanding*, vol. 56, no. 2, pp. 230–241, 1992.

[36] O. Faugeras and T. Papadopoulo, "A nonlinear method for estimating the projective geometry of 3 views," in *Sixth International Conference on Computer Vision (IEEE Cat. No. 98CH36271)*, IEEE, 1998, pp. 477–484.

[37] A. Heyden, "A common framework for multiple view tensors," in *European Conference on Computer Vision*, Springer, 1998, pp. 3–19.

[38] A. Heyden, "Tensorial properties of multiple view constraints," *Mathematical Methods in the Applied Sciences*, vol. 23, no. 2, pp. 169–202, 2000.

[39] A. Shashua, "Trilinear tensor: The fundamental construct of multiple-view geometry and its applications," in *International Workshop on Algebraic Frames for the Perception-Action Cycle*, Springer, 1997, pp. 190–206.

[40] P. R. Mendonça and R. Cipolla, "Analysis and computation of an affine trifocal tensor.," in *BMVC*, 1998, pp. 1–9.

[41] R. Hartley Richard Iand Vidal, "The multibody trifocal tensor: Motion segmentation from 3 perspective views," in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, IEEE, vol. 1, 2004, pp. I–I.

[42] S. Avidan and A. Shashua, "Unifying two-view and three-view geometry," in *In ARPA, Image Understanding Workshop*, 1997.

[43] S. Avidan and A. Shashua, "Tensor embedding of the fundamental matrix," in *European Workshop on 3D Structure from Multiple Images of Large-Scale Environments*, Springer, 1998, pp. 47–62.

[44] S. Avidan and A. Shashua, "Tensorial transfer: On the representation of n> 3 views of a 3d scene," in *Proceedings of the ARPA Image Understanding Workshop*, Citeseer, 1996.

[45] A. Shashua and M. Werman, "Trilinearity of three perspective views and its associated tensor," in *1995 Fifth International Conference on Computer Vision Proceedings.,*, IEEE, 1995, pp. 920–925.

[46] A. Shashua and M. Werman, "Fundamental tensor: On the geometry of three perspective views," in *ICCV'95*, Citeseer, 1995.

[47] A. Shashua and S. Avidan, "The rank 4 constraint in multiple (3) view geometry," in *European Conference on Computer Vision*, Springer, 1996, pp. 196–206.

[48] S. Avidan and A. Shashua, "Novel view synthesis in tensor space," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE, 1997, pp. 1034–1040.

[49] O. Faugeras and T. Papadopoulo, "Grassman–cayley algebra for modelling systems of cameras and the algebraic equations of the manifold of trifocal tensors," *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, vol. 356, no. 1740, pp. 1123–1152, 1998.

[50] T. Papadopoulo and O. Faugeras, “A new characterization of the trifocal tensor,” in *European Conference on Computer Vision*, Springer, 1998, pp. 109–123.

[51] N. Canterakis, “A minimal set of constraints for the trifocal tensor,” in *European Conference on Computer Vision*, Springer, 2000, pp. 84–99.

[52] C. Ressl, “Geometry, constraints and computation of the trifocal tensor,” PhD thesis, 2003.

[53] A. Alzati and A. Tortora, “A geometric approach to the trifocal tensor,” *Journal of Mathematical Imaging and Vision*, vol. 38, no. 3, pp. 159–170, 2010.

[54] S. B. Heinrich and W. E. Snyder, “Internal constraints of the trifocal tensor,” *arXiv preprint arXiv:1103.6052*, 2011.

[55] R. I. Hartley, “Projective reconstruction from line correspondences,” in *CVPR*, 1994, pp. 903–907.

[56] R. I. Hartley, “A linear method for reconstruction from lines and points,” in *Computer Vision, 1995. Proceedings., Fifth International Conference on*, IEEE, 1995, pp. 882–887.

[57] A. Heyden, “Reconstruction from three images of six point objects,” in *Proc. Symposium on Image Analysis, SSAB, Halmstad, Sweden*, Citeseer, 1994, pp. 49–52.

[58] L. Quan, “Invariants of six points and projective reconstruction from three uncalibrated images,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 1, pp. 34–46, 1995.

[59] R. I. Hartley and N. Y. Dano, “Reconstruction from six-point sequences,” in *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No. PR00662)*, IEEE, vol. 2, 2000, pp. 480–486.

[60] A. Heyden, “Reconstruction and prediction from three images of uncalibrated cameras,” in *Theory And Applications Of Image Analysis II: Selected Papers from the 9th Scandinavian Conference on Image Analysis*, World Scientific, 1995, pp. 113–126.

[61] P. H. Torr and A. Zisserman, “Robust parameterization and computation of the trifocal tensor,” *Image and vision Computing*, vol. 15, no. 8, pp. 591–605, 1997.

[62] R. I. Hartley, “Minimizing algebraic error,” *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, vol. 356, no. 1740, pp. 1175–1192, 1998.

[63] Y. Kuang, M. Oskarsson, and K. Åström, "Revisiting trifocal tensor estimation using lines," in *2014 22nd International Conference on Pattern Recognition*, IEEE, 2014, pp. 2419–2423.

[64] F. Schaffalitzky, A. Zisserman, R. I. Hartley, and P. H. Torr, "A six point solution for structure and motion," in *European Conference on Computer Vision*, Springer, 2000, pp. 632–648.

[65] S. Heinrich and W. Snyder, "Robust estimation of the trifocal tensor: A comparative performance evaluation," *Preprint, https://shorturl. at/abhqu*, vol. 3, 2011.

[66] H. Mayer, "Estimation of and view synthesis with the trifocal tensor," *International Archives of Photogrammetry Remote Sensing And Spatial Inormation Sciences*, vol. 34, no. 3/A, pp. 211–217, 2002.

[67] A. Shashua, "Multiple-view geometry and photometry," in *Asian Conference on Computer Vision*, Springer, 1995, pp. 393–404.

[68] G. P. Stein and A. Shashua, "Model-based brightness constraints: On direct estimation of structure and motion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 9, pp. 992–1015, 2000.

[69] Q.-T. Luong and T. Viéville, "Canonical representations for the geometries of multiple projective views," *Computer vision and image understanding*, vol. 64, no. 2, pp. 193–229, 1996.

[70] B. Triggs, "Matching constraints and the joint image," in *Proceedings of IEEE international conference on computer vision*, IEEE, 1995, pp. 338–343.

[71] B. Triggs, *The Geometry of Projective Reconstruction I: Matching Constraints and the Joint Image*, Circulated in 1995. Accepted subject to revision to IJCV in 1995, but never completed, 1995. [Online]. Available: https://hal.inria.fr/inria-00548382.

[72] A. Heyden, "Reconstruction from multiple images using kinetic depths," *International Journal of Computer Vision*, 1995.

[73] A. Heyden, "Reconstruction from image sequences by means of relative depths," *International journal of computer vision*, vol. 24, no. 2, pp. 155–161, 1997.

[74] R. I. Hartley, "Multilinear relationships between coordinates of corresponding image points and lines," in *Proc. International Workshop on Computer Vision and Applied Geometry*, 1995.

[75] O. Faugeras and B. Mourrain, “On the geometry and algebra of the point and line correspondences between n images,” in *Proceedings of IEEE International Conference on Computer Vision*, IEEE, 1995, pp. 951–956.

[76] R. I. Hartley, “Computation of the quadrifocal tensor,” in *European Conference on Computer Vision*, Springer, 1998, pp. 20–35.

[77] A. Heyden and K. Åström, “Algebraic varieties in multiple view geometry,” in *European Conference on Computer Vision*, Springer, 1996, pp. 669–682.

[78] A. Heyden and K. Åström, “Algebraic properties of multilinear constraints,” 13, vol. 20, Wiley Online Library, 1997, pp. 1135–1162.

[79] S. Carlsson, “The double algebra: An effective tool for computing invariants in computer vision,” in *Joint European-US Workshop on Applications of Invariance in Computer Vision*, Springer, 1993, pp. 145–164.

[80] S. Carlsson, “Duality of reconstruction and positioning from projective views,” in *Representation of Visual Scenes, 1995.(In Conjuction with ICCV'95), Proceedings IEEE Workshop on*, IEEE, 1995, pp. 85–92.

[81] D. Weinshall, M. Werman, and A. Shashua, “Shape tensors for efficient and learnable indexing,” in *Proceedings IEEE Workshop on Representation of Visual Scenes (In Conjunction with ICCV'95)*, IEEE, 1995, pp. 58–65.

[82] D. Weinshall, M. Werman, and A. Shashua, “Duality of multi-point and multi-frame geometry: Fundamental shape matrices and tensors,” in *European Conference on Computer Vision*, Springer, 1996, pp. 217–227.

[83] S. Carlsson and D. Weinshall, “Dual computation of projective shape and camera positions from multiple images,” *International Journal of Computer Vision*, vol. 27, no. 3, pp. 227–241, 1998.

[84] F. Doyle, “The historical development of analytical photogrammetry,” *Photogrammetric Engineering*, vol. 30, no. 2, pp. 259–265, 1964.

[85] D. C. Brown, “A solution to the general problem of multiple station analytical stereo triangulation,” *RCA Data reduction Technical Report*, vol. 43, 1958.

[86] D. C. Brown, “The bundle method–progress and prospects,” *International Archives of Photogrammetry*, vol. 21, no. 3, pp. 1–33, 1976.

[87] S. Granshaw, “Bundle adjustment methods in engineering photogrammetry,” *The Photogrammetric Record*, vol. 10, no. 56, pp. 181–207, 1980.

[88] C. S. Fraser *et al.*, "Network design considerations for non-topographic photogrammetry," *Photogrammetric Engineering and Remote Sensing*, vol. 50, no. 8, pp. 1115–1126, 1984.

[89] S. Mason, "Expert system-based design of close-range photogrammetric networks," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 50, no. 5, pp. 13–24, 1995.

[90] D. C. Brown, "Close-range camera calibration," *Photogramm. Eng*, vol. 37, no. 8, pp. 855–866, 1971.

[91] W. Faig, "Calibration of close-range photogrammetric systems: Mathematical formulation.," 1975.

[92] F. Ackermann, "Block adjustment with additional parameters," *Photogrammetria*, vol. 36, no. 6, pp. 217–227, 1981.

[93] A. Alharthy and J. Bethel, "Laboratory self-calibration of a multi-band sensor," *INTERNATIONAL ARCHIVES OF PHOTOGRAMMETRY REMOTE SENSING AND SPATIAL INFORMATION SCIENCES*, vol. 34, no. 3/A, pp. 23–28, 2002.

[94] A. Grün, "The accuracv potential of the modern bundle lock adjustment i in aerial photogrammetry," *Photogrammetric engineering and remote sensing*, vol. 48, no. 1, pp. 45–54, 1982.

[95] W. Förstner, "The reliability of block triangulation," *Photogrammetric Engineering & Remote Sensing*, vol. 51, no. 8, pp. 1137–1149, 1985.

[96] S. El-Hakim, "A step-by-step strategy for gross-error detection," *Photogrammetric engineering and remote sensing*, vol. 50, no. 6, pp. 713–718, 1984.

[97] A. Dermanis, "The photogrammetric inner constraints," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 49, no. 1, pp. 25–39, 1994.

[98] P. F. McLauchlan, "Gauge independence in optimization algorithms for 3d vision," in *International Workshop on Vision Algorithms*, Springer, 1999, pp. 183–199.

[99] P. F. McLauchlan, "Gauge invariance in projective 3d reconstruction," in *Proceedings IEEE Workshop on Multi-View Modeling and Analysis of Visual Scenes (MVIEW'99)*, IEEE, 1999, pp. 37–44.

[100] D. D. Morris, K. Kanatani, and T. Kanade, "Uncertainty modeling for optimal structure from motion," in *International Workshop on Vision Algorithms*, Springer, 1999, pp. 200–217.

[101] E. M. Mikhail and H. RJ, “Recursive methods in photo-grammetric data reduction,” in *PHOTOGRAMMETRIC ENGINEERING*, AMER SOC PHOTOGRAMMETRY 5410 GROSVENOR LANE SUITE 210, BETHESDA, MD 20814-2160, vol. 38, 1972, p. 595.

[102] J. Blais, “Linear least-squares computations using givens transformations,” *The Canadian Surveyor*, vol. 37, no. 4, pp. 225–233, 1983.

[103] K. R. Holm, “Test of algorithms for sequential adjustment in online phototriangulation,” *Photogrammetria*, vol. 43, no. 3-4, pp. 143–156, 1989.

[104] P. F. McLauchlan and D. W. Murray, “A unifying framework for structure and motion recovery from image sequences,” in *Proceedings of IEEE International Conference on Computer Vision*, IEEE, 1995, pp. 314–320.

[105] P. F. McLauchlan, “A batch/recursive algorithm for 3d scene reconstruction,” in *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No. PR00662)*, IEEE, vol. 2, 2000, pp. 738–743.

[106] S. Agarwal, Y. Furukawa, N. Snavely, I. Simon, B. Curless, S. M. Seitz, and R. Szeliski, “Building rome in a day,” *Communications of the ACM*, vol. 54, no. 10, pp. 105–112, 2011.

[107] A. Elassal, “Algorithm for the general analytical solution,” *Photogrammetric Engineering*, vol. 35, no. 12, pp. 1268–1277, 1969.

[108] M. I. Lourakis and A. A. Argyros, “Sba: A software package for generic sparse bundle adjustment,” *ACM Transactions on Mathematical Software (TOMS)*, vol. 36, no. 1, pp. 1–30, 2009.

[109] M. Byröd and K. Åström, “Conjugate gradient bundle adjustment,” in *European Conference on Computer Vision*, Springer, 2010, pp. 114–127.

[110] S. Agarwal, N. Snavely, S. M. Seitz, and R. Szeliski, “Bundle adjustment in the large,” in *European conference on computer vision*, Springer, 2010, pp. 29–42.

[111] C. Wu, S. Agarwal, B. Curless, and S. M. Seitz, “Multicore bundle adjustment,” in *CVPR 2011*, IEEE, 2011, pp. 3057–3064.

[112] R. Hänsch, I. Drude, and O. Hellwich, “Modern methods of bundle adjustment on the gpu.,” *ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences*, vol. 3, no. 3, 2016.

[113] C. Ressl, “An introduction to the relative orientation using the trifocal tensor,” *International Archives of Photogrammetry and Remote Sensing*, vol. 33, no. B3/2; PART 3, pp. 769–776, 2000.

[114] B. Liu, M. Yu, D. Maier, and R. Männer, “Accelerated bundle adjustment in multiple-view reconstruction,” in *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*, Springer, 2003, pp. 1195–1201.

[115] A. L. Rodríguez, P. E. López-de-Teruel, and A. Ruiz, “Reduced epipolar cost for accelerated incremental sfm,” in *CVPR 2011*, IEEE, 2011, pp. 3097–3104.

[116] R. Steffen, J.-M. Frahm, and W. Förstner, “Relative bundle adjustment based on trifocal constraints,” in *European Conference on Computer Vision*, Springer, 2010, pp. 282–295.

[117] V. Indelman, “Bundle adjustment without iterative structure estimation and its application to navigation,” in *Proceedings of IEEE/ION PLANS 2012*, 2012, pp. 748–756.

[118] J. Schneider, C. Stachniss, and W. Förstner, “On the quality and efficiency of approximate solutions to bundle adjustment with epipolar and trifocal constraints,” *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 4, p. 81, 2017.

[119] C. J. Taylor and D. J. Kriegman, “Structure and motion from line segments in multiple images,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 11, pp. 1021–1032, 1995.

[120] J. M. M. Montiel, J. Civera, and A. J. Davison, “Unified inverse depth parametrization for monocular slam,” Robotics: Science and Systems, 2006.

[121] J. Civera, A. J. Davison, and J. M. Montiel, “Inverse depth parametrization for monocular slam,” *IEEE transactions on robotics*, vol. 24, no. 5, pp. 932–945, 2008.

[122] J. Schneider, F. Schindler, T. Läbe, and W. Förstner, “Bundle adjustment for multi-camera systems with points at infinity,” *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 3, pp. 75–80, 2012.

[123] H. Theiss, E. Mikhail, I. Aly, J. Bethel, and C. Lee, “Photogrammetric invariance,” in *Proceedings of ISPRS 19th Congress, Amsterdam, The Netherlands*, 2000.

[124] A. Macdonald, *Linear and geometric algebra*. Alan Macdonald, 2010.

[125] V. Indelman, “Navigation performance enhancement using online mosaicking,” PhD thesis, the Technion - Israel Institute of Technology, 2011.

[126] Z. Zhang, “A flexible new technique for camera calibration,” *IEEE Transactions on pattern analysis and machine intelligence*, vol. 22, no. 11, pp. 1330–1334, 2000.

[127] C. G. Harris, M. Stephens, *et al.*, “A combined corner and edge detector.,” in *Alvey vision conference*, Citeseer, vol. 15, 1988, pp. 10–5244.

A. APPENDICES

There are some full steps derivations in the appendix. The first one is the derivation of the simplified projection equation (3.4). The second one is the derivation of zero determinant condition of three-frame matrix. The third one is the derivation of zero determinant condition of four-frame matrix. The final section is the derivations, expressions and derivatives of the trilinear equation.

A.1 Derivation of Column Vector Approach

One object point gives two equations to one camera in the equation (3.2). There is a simplified projection relation for this issue, as shown below.

$$\begin{aligned}\mathbf{x} &\sim M[I| - \widetilde{\mathbf{C}}]\mathbf{X} \text{ where } M = KR \text{ or } R\\ M^{-1}\mathbf{x} &\sim [I| - \widetilde{\mathbf{C}}]\mathbf{X} \text{ where } \chi = M^{-1}\mathbf{x} = [u, v, w]^T\\ [\chi]_\times \chi &\sim [\chi]_\times [I| - \widetilde{\mathbf{C}}]\mathbf{X} = \mathbf{0}\end{aligned}$$

The above derivation shows $[M^{-1}\mathbf{x}]_\times [I| - \widetilde{\mathbf{C}}]\mathbf{X} = 0$. If $\chi_{4-\mathrm{i}} \neq 0, \mathrm{i} \in [1, 3]$, there are three matrices S_i with $\det(S_\mathrm{i}) \neq 0$ corresponding. This expression is abbreviated as the one below.

$$[\chi]_\times [I| - \widetilde{\mathbf{C}}]\mathbf{X} = \mathbf{0} \tag{A.1}$$

A.1.1 Format One

If $\chi_3 = w \neq 0$, replacing the third column of $[\chi]_\times$ by the vector χ gives a designed matrix S_1^{-1}.

$$S_1^{-1} = \begin{bmatrix} 0 & -w & u \\ w & 0 & v \\ -v & u & w \end{bmatrix} \text{ where } \det(S_1^{-1}) = (u^2 + v^2 + w^2)w,$$

and

$$S_1 = \begin{bmatrix} -uv & u^2 + w^2 & -vw \\ -v^2 - w^2 & uv & uw \\ uw & vw & w^2 \end{bmatrix} \cdot \frac{1}{\det(S_1^{-1})} \text{ where } S_1 [\chi]_\times \sim \begin{bmatrix} 1 & 0 & -u/w \\ 0 & 1 & -v/w \\ 0 & 0 & 0 \end{bmatrix}.$$

Multiplying the S_1 to (A.1)gives equation below.

$$S_1 [\chi]_\times [I| - \widetilde{\mathbf{C}}]\mathbf{X} = \mathbf{0}$$
$$\begin{bmatrix} 1 & 0 & -u/w & -c_1 + c_3 u/w \\ 0 & 1 & -v/w & -c_2 + c_3 v/w \\ 0 & 0 & 0 & 0 \end{bmatrix} \mathbf{X} = \mathbf{0} \text{ where } [c_1, c_2, c_3]^T = \widetilde{\mathbf{C}}$$

This expression is abbreviated as the one below.

$$\begin{bmatrix} 1 & 0 & m & n \\ 0 & 1 & p & q \end{bmatrix} \mathbf{X} = \mathbf{0} \tag{A.2}$$

A.1.2 Format Two

If $\chi_2 = v \neq 0$, replacing the second column of $[\chi]_\times$ by the vector χ gives a designed matrix S_2^{-1}.

$$S_2^{-1} = \begin{bmatrix} 0 & u & v \\ w & v & -u \\ -v & w & 0 \end{bmatrix} \text{ where } \det(S_2^{-1}) = (u^2+v^2+w^2)v,$$

and

$$S_2 = \begin{bmatrix} uw & vw & -u^2-v^2 \\ uv & v^2 & vw \\ v^2+w^2 & -vu & -uw \end{bmatrix} \cdot \frac{1}{(\det(S_2^{-1})} \text{ where } S_2\,[\chi]_\times \sim \begin{bmatrix} 1 & -u/v & 0 \\ 0 & 0 & 0 \\ 0 & -w/v & 1 \end{bmatrix}.$$

Multiplying the S_2 to (A.1)gives equation below.

$$S_2\,[\chi]_\times\,[I| - \widetilde{\mathbf{C}}]\mathbf{X} = \mathbf{0}$$
$$\begin{bmatrix} 1 & -u/v & 0 & -c_1+c_2u/v \\ 0 & 0 & 0 & 0 \\ 0 & -w/v & 1 & -c_3+c_2w/v \end{bmatrix} \mathbf{X} = \mathbf{0} \text{ where } [c_1,c_2,c_3]^T = \widetilde{\mathbf{C}}$$

This expression is abbreviated as the one below.

$$\begin{bmatrix} 1 & m & 0 & n \\ 0 & p & 1 & q \end{bmatrix} \mathbf{X} = \mathbf{0} \tag{A.3}$$

A.1.3 Format Three

If $\chi_1 = u \neq 0$, replacing the first column of $[\chi]_\times$ by the vector χ gives a designed matrix S_3^{-1}.

$$S_3^{-1} = \begin{bmatrix} u & -w & v \\ v & 0 & -u \\ w & u & 0 \end{bmatrix} \text{ where } \det(S_3^{-1}) = (u^2+v^2+w^2)u,$$

and

$$S_3 = \begin{bmatrix} uw & vw & -u^2-v^2 \\ uv & v^2 & vw \\ v^2+w^2 & -vu & -uw \end{bmatrix} \cdot \frac{1}{\det(S_3^{-1})} \text{ where } S_3\,[\chi]_\times \sim \begin{bmatrix} 0 & 0 & 0 \\ -v/u & 1 & 0 \\ -w/u & 0 & 1 \end{bmatrix}.$$

Multiplying the S_3 to (A.1)gives equation below.

$$S_3 \left[\chi\right]_\times [I| - \widetilde{\mathbf{C}}]\mathbf{X} = \mathbf{0}$$
$$\begin{bmatrix} 0 & 0 & 0 & 0 \\ -v/u & 1 & 0 & -c_2 + c_1 v/u \\ -w/u & 0 & 1 & -c_3 + c_1 w/u \end{bmatrix} \mathbf{X} = \mathbf{0} \text{ where } [c_1, c_2, c_3]^T = \widetilde{\mathbf{C}}$$

This expression is abbreviated as the one below.

$$\begin{bmatrix} m & 1 & 0 & n \\ p & 0 & 1 & q \end{bmatrix} \mathbf{X} = \mathbf{0} \tag{A.4}$$

A.1.4 Column Vectors

At here the combinations of the three types of equations are talked about. The combination of two simplified projection (A.2) gives the four equations, which are below in the stacked matrix format.

$$\begin{bmatrix} 1 & 0 & m_1 & n_1 \\ 0 & 1 & p_1 & q_1 \\ 1 & 0 & m_2 & n_2 \\ 0 & 1 & p_2 & q_2 \end{bmatrix} \mathbf{X} = \mathbf{0}$$

Gaussian elimination of the above equation gives the equation below.

$$\begin{bmatrix} 1 & 0 & m_1 & n_1 \\ 0 & 1 & p_1 & q_1 \\ 0 & 0 & m_2 - m_1 & n_2 - n_1 \\ 0 & 0 & p_2 - p_1 & q_2 - q_1 \end{bmatrix} \mathbf{X} = \mathbf{0}$$

$$\text{where } \begin{bmatrix} \mathbf{U}_2 & \mathbf{V}_2 \end{bmatrix} = \begin{bmatrix} m_2 - m_1 & n_2 \\ q_2 - q_1 & q_2 \end{bmatrix} \text{ and } \mathbf{U}_2 \parallel \mathbf{V}_2$$

The combination of simplified projection (A.2) and (A.3) gives the four equations, which are below in the stacked matrix format.

$$\begin{bmatrix} 1 & 0 & m_1 & n_1 \\ 0 & 1 & p_1 & q_1 \\ 1 & m_2 & 0 & n_2 \\ 0 & p_2 & 1 & q_2 \end{bmatrix} \mathbf{X} = \mathbf{0}$$

Gaussian elimination of the above equation gives the equation below.

$$\begin{bmatrix} 1 & 0 & m_1 & n_1 \\ 0 & 1 & p_1 & q_1 \\ 0 & 0 & -m_1 - m_2 p_1 & n_2 - n_1 - m_2 q_1 \\ 0 & 0 & 1 - p_2 p_1 & q_2 - p_2 q_1 \end{bmatrix} \mathbf{X} = \mathbf{0}$$

$$\text{where} \begin{bmatrix} \mathbf{U}_2 & \mathbf{V}_2 \end{bmatrix} = \begin{bmatrix} -m_1 - m_2 p_1 & n_2 - n_1 - m_2 q_1 \\ 1 - p_2 p_1 & q_2 - p_2 q_1 \end{bmatrix} \text{ and } \mathbf{U}_2 \parallel \mathbf{V}_2$$

The combination of simplified projection (A.2) and (A.4) gives the four equations, which are below in the stacked matrix format.

$$\begin{bmatrix} 1 & 0 & m_1 & n_1 \\ 0 & 1 & p_1 & q_1 \\ m_2 & 1 & 0 & n_2 \\ p_2 & 0 & 1 & q_2 \end{bmatrix} \mathbf{X} = \mathbf{0}$$

Gaussian elimination of the above equation gives the equation below.

$$\begin{bmatrix} 1 & 0 & m_1 & n_1 \\ 0 & 1 & p_1 & q_1 \\ 0 & 0 & -m_2 m_1 - p_1 & n_2 - m_2 n_1 - q_1 \\ 0 & 0 & 1 - p_2 m_1 & q_2 - p_2 n_1 \end{bmatrix} \mathbf{X} = \mathbf{0}$$

$$\text{where} \begin{bmatrix} \mathbf{U}_2 & \mathbf{V}_2 \end{bmatrix} = \begin{bmatrix} -m_2 m_1 - p_1 & n_2 - m_2 n_1 - q_1 \\ 1 - p_2 m_1 & q_2 - p_2 n_1 \end{bmatrix} \text{ and } \mathbf{U}_2 \parallel \mathbf{V}_2$$

Given many frames, and the below format is easy to get.

$$\begin{bmatrix} 1 & 0 & m_1 & n_1 \\ 0 & 1 & p_1 & q_1 \\ \mathbf{0} & \mathbf{0} & \mathbf{U} & \mathbf{V} \end{bmatrix} \mathbf{X} = \mathbf{0} \text{ and } \mathbf{U} \parallel \mathbf{V} \tag{A.5}$$

To brief the discussion, the expression (A.2) is used in the column vector approach analyses.

Given four sets of equations (A.2), the eight equations are below in the stacked matrix format.

$$\begin{bmatrix} 1 & 0 & m_1 & n_1 \\ 0 & 1 & p_1 & q_1 \\ 1 & 0 & m_2 & n_2 \\ 0 & 1 & p_2 & q_2 \\ 1 & 0 & m_3 & n_3 \\ 0 & 1 & p_3 & q_3 \\ 1 & 0 & m_4 & n_4 \\ 0 & 1 & p_4 & q_4 \end{bmatrix} \mathbf{X} = \mathbf{0}$$

Let the $\mathrm{i}-th$ group minus the first group. This Gaussian elimination of the above equation gives the equations below.

$$\begin{bmatrix} 1 & 0 & m_1 & n_1 \\ 0 & 1 & p_1 & q_1 \\ 0 & 0 & m_2 - m_1 & n_2 - n_1 \\ 0 & 0 & p_2 - p_1 & q_2 - q_1 \\ 0 & 0 & m_3 - m_1 & n_3 - n_1 \\ 0 & 0 & p_3 - p_1 & q_3 - q_1 \\ 0 & 0 & m_4 - m_1 & n_4 - n_1 \\ 0 & 0 & p_4 - p_1 & q_4 - q_1 \end{bmatrix} \mathbf{X} = \mathbf{0} \tag{A.6}$$

There is another format of this expression. Let the $\mathrm{i}-th$ group minus the $(\mathrm{i}-1)th$ group. This Gaussian elimination of the above equation gives the equations below.

$$\begin{bmatrix} 1 & 0 & m_1 & n_1 \\ 0 & 1 & p_1 & q_1 \\ 0 & 0 & m_2 - m_1 & n_2 - n_1 \\ 0 & 0 & p_2 - p_1 & q_2 - q_1 \\ 0 & 0 & m_3 - m_2 & n_3 - n_2 \\ 0 & 0 & p_3 - p_2 & q_3 - q_2 \\ 0 & 0 & m_4 - m_3 & n_4 - n_3 \\ 0 & 0 & p_4 - p_3 & q_4 - q_3 \end{bmatrix} \mathbf{X} = \mathbf{0}$$

This method is the linear transformation previous way (A.6). It will makes some dependent expressions talked in the column vector analyses. The method (A.6) is always used in this thesis.

A.2 Zero Determinant Condition of Three-Frame Geometry

The derivation starts from the zero determinant condition of in $A_{\Lambda 3}^T A_{\Lambda 3}$ in (3.13). There is a simplified projection relation for this issue, as shown below.

$$A_{\Lambda 3}^T A_{\Lambda 3} = \begin{bmatrix} \mathbf{x}_2^T R_2 R_2^T \mathbf{x}_2 & 0 & -\mathbf{x}_2^T R_2 \mathbf{x}_1 & \mathbf{x}_2^T R_2 \widetilde{\mathbf{C}}_2 \\ 0 & \mathbf{x}_3^T R_3 R_3^T \mathbf{x}_3 & -\mathbf{x}_3^T R_3 \mathbf{x}_1 & \mathbf{x}_3^T R_3 \widetilde{\mathbf{C}}_3 \\ -\mathbf{x}_1^T R_2^T \mathbf{x}_2 & -\mathbf{x}_1^T R_3^T \mathbf{x}_3 & 2\mathbf{x}_1^T \mathbf{x}_1 & -\mathbf{x}_1^T (\widetilde{\mathbf{C}}_2 + \widetilde{\mathbf{C}}_3) \\ \widetilde{\mathbf{C}}_2^T R_2^T \mathbf{x}_2 & \widetilde{\mathbf{C}}_3^T R_3^T \mathbf{x}_3 & -(\widetilde{\mathbf{C}}_2^T + \widetilde{\mathbf{C}}_3^T)\mathbf{x}_1 & \widetilde{\mathbf{C}}_2^T \widetilde{\mathbf{C}}_2 + \widetilde{\mathbf{C}}_3^T \widetilde{\mathbf{C}}_3 \end{bmatrix}$$

The components of the matrix $M = A_{\Lambda 3}^T A_{\Lambda 3}$ are well defined by the space vectors and their intersection angles, in the following equations. And $\|\mathbf{x}_1\| = \|R_2^T \mathbf{x}_2\| = \|R_3^T \mathbf{x}_3\| = \|\widetilde{\mathbf{C}}_2\| = 1$ and $\|\widetilde{\mathbf{C}}_3\| = t_2$.

$$\mathbf{x}_1^T R_2^T \mathbf{x}_2 = \cos\alpha_1 \qquad \mathbf{x}_1^T R_3^T \mathbf{x}_3 = \cos\alpha_2$$
$$\widetilde{\mathbf{C}}_2^T R_2^T \mathbf{x}_2 = \cos(\boldsymbol{\pi} - \beta_1) \qquad \widetilde{\mathbf{C}}_3^T R_3^T \mathbf{x}_3 = \cos(\boldsymbol{\pi} - \beta_2)\, t_2$$
$$\mathbf{x}_1^T \widetilde{\mathbf{C}}_2 = \cos\theta_1 \qquad \mathbf{x}_1^T \widetilde{\mathbf{C}}_3 = \cos\theta_2 t_2$$

The matrix $M = A_{\Lambda 4}^T A_{\Lambda 4}$ equals to the one below.

$$A_{\Lambda 3}^T A_{\Lambda 3} = \begin{bmatrix} 1 & 0 & -\cos\alpha_1 & \cos(\boldsymbol{\pi} - \beta_1) \\ 0 & 1 & -\cos\alpha_2 & \cos(\boldsymbol{\pi} - \beta_2)\, t_2 \\ -\cos\alpha_1 & -\cos\alpha_2 & 2 & -(\cos\theta_1 + \cos\theta_2 t_2) \\ \cos(\boldsymbol{\pi} - \beta_1) & \cos(\boldsymbol{\pi} - \beta_2)\, t_2 & -(\cos\theta_1 + \cos\theta_2 t_2) & 1 + t_2^2 \end{bmatrix}$$

The matrix $M = A_{\Lambda 3}^T A_{\Lambda 3}$ can be partitioned into four 2 by 2 blocks. The determinant of the matrix is computed by the blocks.

$$\det(M) = \det(M_a) \det\left(M_d - M_c M_a^{-1} M_b\right) \tag{A.7}$$

with

$$M = \begin{bmatrix} M_a & M_b \\ M_c & M_d \end{bmatrix},$$

where

$$M_a = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$
$$M_b = \begin{bmatrix} -\cos\alpha_1 & \cos(\boldsymbol{\pi} - \beta_1) \\ -\cos\alpha_2 & \cos(\boldsymbol{\pi} - \beta_2)\, t_2 \end{bmatrix}$$
$$M_c = \begin{bmatrix} -\cos\alpha_1 & -\cos\alpha_2 \\ \cos(\boldsymbol{\pi} - \beta_1) & \cos(\boldsymbol{\pi} - \beta_2)\, t_2 \end{bmatrix}$$
$$M_d = \begin{bmatrix} 2 & -(\cos\theta_1 + \cos\theta_2 t_2) \\ -(\cos\theta_1 + \cos\theta_2 t_2) & 1 + t_2^2 \end{bmatrix}.$$

And $\det(M_a) = 1$.

$$\det(M) = 0 \Leftrightarrow \left(M_d - M_c M_a^{-1} M_b\right) = (M_d - M_c M_b) = 0$$

$$
\begin{aligned}
M_d - M_c M_a^{-1} M_b &= \begin{bmatrix} 2 & -(\cos\theta_1 + \cos\theta_2 t_2) \\ -(\cos\theta_1 + \cos\theta_2 t_2) & 1 + t_2^2 \end{bmatrix} \\
&\quad - \begin{bmatrix} -\cos\alpha_1 & -\cos\alpha_2 \\ \cos(\pi - \beta_1) & \cos(\pi - \beta_2)\, t_2 \end{bmatrix} \begin{bmatrix} -\cos\alpha_1 & \cos(\pi - \beta_1) \\ -\cos\alpha_2 & \cos(\pi - \beta_2)\, t_2 \end{bmatrix} \\
&= \begin{bmatrix} 2 & -(\cos\theta_1 + \cos\theta_2 t_2) \\ -(\cos\theta_1 + \cos\theta_2 t_2) & 1 + t_2^2 \end{bmatrix} \\
&\quad - \begin{bmatrix} \cos^2\alpha_1 + \cos^2\alpha_2 & \cos\alpha_1 \cos\beta_1 + \cos\alpha_2 \cos\beta_2 t_2 \\ \cos\alpha_1 \cos\beta_1 + \cos\alpha_2 \cos\beta_2 t_2 & \cos^2\beta_1 + \cos^2\beta_2 t_2^2 \end{bmatrix} \\
&= \begin{bmatrix} 2 & \cos(\alpha_1 \cos\beta_1) + \cos(\alpha_2 \cos\beta_2) t_2 \\ \cos(\alpha_1 \cos\beta_1) + \cos(\alpha_2 \cos\beta_2) t_2 & 1 + t_2^2 \end{bmatrix} \\
&\quad - \begin{bmatrix} \cos^2\alpha_1 + \cos^2\alpha_2 & \cos\alpha_1 \cos\beta_1 + \cos\alpha_2 \cos\beta_2 t_2 \\ \cos\alpha_1 \cos\beta_1 + \cos\alpha_2 \cos\beta_2 t_2 & \cos^2\beta_1 + \cos^2\beta_2 t_2^2 \end{bmatrix} \\
&= \begin{bmatrix} 2 - \cos^2\alpha_1 - \cos^2\alpha_2 & -\sin\alpha_1 \sin\beta_1 - \sin\alpha_2 \sin\beta_2 t_2 \\ -\sin\alpha_1 \sin\beta_1 - \sin\alpha_2 \sin\beta_2 t_2 & 1 + t_2^2 - \cos^2\beta_1 - \cos^2\beta_2 t_2^2 \end{bmatrix} \\
&= \begin{bmatrix} \sin^2\alpha_1 + \sin^2\alpha_2 & -\sin\alpha_1 \sin\beta_1 - \sin\alpha_2 \sin\beta_2 t_2 \\ -\sin\alpha_1 \sin\beta_1 - \sin\alpha_2 \sin\beta_2 t_2 & \sin^2\beta_1 + \sin^2\alpha_2 {t_2}^2 \end{bmatrix}
\end{aligned}
$$

The determinant of this 2 by 2 matrix is zero. It represents the determinant condition of the full 4 by 4 matrix.

$$
\begin{aligned}
&\det\left(M_d - M_c M_a^{-1} M_b\right) \\
&= \left(\sin^2\alpha_1 + \sin^2\alpha_2\right)\left(\sin^2\beta_1 + \sin^2\beta_2 {t_2}^2\right) - (\sin\alpha_1 \sin\beta_1 + \sin\alpha_2 \sin\beta_2 t_2)^2 \\
&= \sin^2\beta_1 \sin^2\alpha_2 - 2t_2 \sin\alpha_1 \sin\beta_1 \sin\alpha_2 \sin\beta_2 + {t_2}^2 \sin^2\alpha_1 \sin^2\beta_2 \\
&= (\sin\alpha_2 \sin\beta_1 - t_2 \sin\alpha_1 \sin\beta_2)^2
\end{aligned}
$$

The expression of determinant of $A_{\Lambda 3}^T A_{\Lambda 3}$ has a quadratic format.

$$
\det\left(A_{\Lambda 3}^T A_{\Lambda 3}\right) = (\sin\alpha_2 \sin\beta_1 - t_2 \sin\alpha_1 \sin\beta_2)^2 \tag{A.8}
$$

So the determinant condition is equivalent the equation below.

$$
\det = 0 \Rightarrow \sin\alpha_2 \sin\beta_1 - t_2 \sin\alpha_1 \sin\beta_2 = 0 \tag{A.9}
$$

A.3 Zero Determinant Condition for Four-Frame Geometry

The derivation starts from the zero determinant condition of in $A_{\Lambda 4}^T A_{\Lambda 4}$ in (3.31). There is a simplified projection relation for this issue, as shown below.

$$A_{\Lambda 4}^T A_{\Lambda 4} = \begin{bmatrix} \mathbf{x}_2^T R_2 R_2^T \mathbf{x}_2 & 0 & 0 & -\mathbf{x}_2^T R_2 \mathbf{x}_1 & \mathbf{x}_2^T R_2 \widetilde{\mathbf{C}}_2 \\ 0 & \mathbf{x}_3^T R_3 R_3^T \mathbf{x}_3 & 0 & -\mathbf{x}_3^T R_3 \mathbf{x}_1 & \mathbf{x}_3^T R_3 \widetilde{\mathbf{C}}_3 \\ 0 & 0 & \mathbf{x}_4^T R_4 R_4^T \mathbf{x}_4 & -\mathbf{x}_4^T R_4 \mathbf{x}_1 & \mathbf{x}_4^T R_4 \widetilde{\mathbf{C}}_4 \\ -\mathbf{x}_1^T R_2^T \mathbf{x}_2 & -\mathbf{x}_1^T R_3^T \mathbf{x}_3 & -\mathbf{x}_1^T R_4^T \mathbf{x}_4 & 3\mathbf{x}_1^T \mathbf{x}_1 & -\mathbf{x}_1^T(\widetilde{\mathbf{C}}_2 + \widetilde{\mathbf{C}}_3 + \widetilde{\mathbf{C}}_4) \\ \widetilde{\mathbf{C}}_2^T R_2^T \mathbf{x}_2 & \widetilde{\mathbf{C}}_3^T R_3^T \mathbf{x}_3 & \widetilde{\mathbf{C}}_4^T R_4^T \mathbf{x}_4 & -(\widetilde{\mathbf{C}}_2^T + \widetilde{\mathbf{C}}_3^T + +\widetilde{\mathbf{C}}_4^T)\mathbf{x}_1 & \widetilde{\mathbf{C}}_2^T \widetilde{\mathbf{C}}_2 + \widetilde{\mathbf{C}}_3^T \widetilde{\mathbf{C}}_3 + \widetilde{\mathbf{C}}_4^T \widetilde{\mathbf{C}}_4 \end{bmatrix}$$

The components of the matrix $M = A_{\Lambda 4}^T A_{\Lambda 4}$ are well defined by the space vectors and their intersection angles, in the following equations. And $\|\mathbf{x}_1\| = \|R_2^T \mathbf{x}_2\| = \|R_3^T \mathbf{x}_3\| = \|R_4^T \mathbf{x}_4\| = \|\widetilde{\mathbf{C}}_2\| = 1$ and $\|\widetilde{\mathbf{C}}_3\| = t_2$,$\|\widetilde{\mathbf{C}}_4\| = t_3$.

$$\begin{array}{ccc} \mathbf{x}_1^T R_2^T \mathbf{x}_2 = \cos\alpha_1 & \mathbf{x}_1^T R_3^T \mathbf{x}_3 = \cos\alpha_2 & \mathbf{x}_1^T R_4^T \mathbf{x}_4 = \cos\alpha_3 \\ \widetilde{\mathbf{c}}_2^T R_2^T \mathbf{x}_2 = \cos(\boldsymbol{\pi} - \beta_1) & \widetilde{\mathbf{c}}_3^T R_3^T \mathbf{x}_3 = \cos(\boldsymbol{\pi} - \beta_2)\, t_2 & \widetilde{\mathbf{c}}_4^T R_4^T \mathbf{x}_4 = \cos(\boldsymbol{\pi} - \beta_3)\, t_3 \\ \mathbf{x}_1^T \widetilde{\mathbf{C}}_2 = \cos\theta_1 & \mathbf{x}_1^T \widetilde{\mathbf{C}}_3 = \cos\theta_2 t_2 & \mathbf{x}_1^T \widetilde{\mathbf{C}}_4 = \cos\theta_3 t_3 \end{array}$$

The matrix $M = A_{\Lambda 4}^T A_{\Lambda 4}$ equals to the one below.

$$A_{\Lambda 4}^T A_{\Lambda 4} = \begin{bmatrix} 1 & 0 & 0 & -\cos\alpha_1 & \cos(\pi - \beta_1) \\ 0 & 1 & 0 & -\cos\alpha_2 & \cos(\pi - \beta_2)\, t_2 \\ 0 & 0 & 1 & -\cos\alpha_3 & \cos(\pi - \beta_3)\, t_3 \\ -\cos\alpha_1 & -\cos\alpha_2 & -\cos\alpha_3 & 3 & -(\cos\theta_1 + \cos\theta_2 t_2 + \cos\theta_3 t_3) \\ \cos(\pi - \beta_1) & \cos(\pi - \beta_2)\, t_2 & \cos(\pi - \beta_3)\, t_3 & -(\cos\theta_1 + \cos\theta_2 t_2 + \cos\theta_3 t_3) & 1 + t_2^2 + t_3^2 \end{bmatrix}$$

The matrix $M = A_{\Lambda 3}^T A_{\Lambda 3}$ can be partitioned into four 2 by 2 blocks. The determinant of the matrix is computed by the blocks.

$$\det(M) = \det(M_a)\det\left(M_d - M_c M_a^{-1} M_b\right) \tag{A.10}$$

with

$$M = \begin{bmatrix} M_a & M_b \\ M_c & M_d \end{bmatrix},$$

where

$$M_a = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$M_b = \begin{bmatrix} -\cos\alpha_1 & \cos(\pi - \beta_1) \\ -\cos\alpha_2 & \cos(\pi - \beta_2)\,t_2 \\ -\cos\alpha_3 & \cos(\pi - \beta_3)\,t_3 \end{bmatrix}$$

$$M_c = \begin{bmatrix} -\cos\alpha_1 & -\cos\alpha_2 & -\cos\alpha_3 \\ \cos(\pi - \beta_1) & \cos(\pi - \beta_2)\,t_2 & \cos(\pi - \beta_3)\,t_3 \end{bmatrix}$$

$$M_d = \begin{bmatrix} 3 & -(\cos\theta_1 + \cos\theta_2 t_2 + \cos\theta_3 t_3) \\ -(\cos\theta_1 + \cos\theta_2 t_2 + \cos\theta_3 t_3) & 1 + t_2^2 + t_3^2 \end{bmatrix}.$$

And $\det(M_a) = 1$.

$$\det(M) = 0 \Leftrightarrow \left(M_d - M_c M_a^{-1} M_b\right) = (M_d - M_c M_b) = 0$$

$$\begin{aligned}
M_d - M_c M_a^{-1} M_b &= \begin{bmatrix} 3 & -(\cos\theta_1 + \cos\theta_2 t_2 + \cos\theta_3 t_3) \\ -(\cos\theta_1 + \cos\theta_2 t_2 + \cos\theta_3 t_3) & 1 + t_2^2 + t_3^2 \end{bmatrix} \\
&\quad - \begin{bmatrix} -\cos\alpha_1 & -\cos\alpha_2 & -\cos\alpha_3 \\ \cos(\pi - \beta_1) & \cos(\pi - \beta_2)\,t_2 & \cos(\pi - \beta_3)\,t_3 \end{bmatrix} \begin{bmatrix} -\cos\alpha_1 & \cos(\pi - \beta_1) \\ -\cos\alpha_2 & \cos(\pi - \beta_2)\,t_2 \\ -\cos\alpha_3 & \cos(\pi - \beta_3)\,t_3 \end{bmatrix} \\
&= \begin{bmatrix} 3 & -(\cos\theta_1 + \cos\theta_2 t_2 + \cos\theta_3 t_3) \\ -(\cos\theta_1 + \cos\theta_2 t_2 + \cos\theta_3 t_3) & 1 + t_2^2 + t_3^2 \end{bmatrix} \\
&\quad - \begin{bmatrix} \cos^2\alpha_1 + \cos^2\alpha_2 + \cos^2\alpha_3 & \cos\alpha_1\cos\beta_1 + \cos\alpha_2\cos\beta_2 t_2 + \cos\alpha_3\cos\beta_3 t_3 \\ \cos\alpha_1\cos\beta_1 + \cos\alpha_2\cos\beta_2 t_2 + \cos\alpha_3\cos\beta_3 t_3 & \cos^2\beta_1 + \cos^2\beta_2 t_2^2 + \cos^2\beta_3 t_3^2 \end{bmatrix} \\
&= \begin{bmatrix} 3 & \cos(\alpha_1\cos\beta_1) + \cos(\alpha_2\cos\beta_2)t_2 + \cos(\alpha_3\cos\beta_3)t_3 \\ \cos(\alpha_1\cos\beta_1) + \cos(\alpha_2\cos\beta_2)t_2 + \cos(\alpha_3\cos\beta_3)t_3 & 1 + t_2^2 + t_3^2 \end{bmatrix} \\
&\quad - \begin{bmatrix} \cos^2\alpha_1 + \cos^2\alpha_2 + \cos^2\alpha_3 & \cos\alpha_1\cos\beta_1 + \cos\alpha_2\cos\beta_2 t_2 + \cos\alpha_3\cos\beta_3 t_3 \\ \cos\alpha_1\cos\beta_1 + \cos\alpha_2\cos\beta_2 t_2 + \cos\alpha_3\cos\beta_3 t_3 & \cos^2\beta_1 + \cos^2\beta_2 t_2^2 + \cos^2\beta_3 t_3^2 \end{bmatrix} \\
&= \begin{bmatrix} 3 - \cos^2\alpha_1 - \cos^2\alpha_2 - \cos^2\alpha_3 & -\sin\alpha_1\sin\beta_1 - \sin\alpha_2\sin\beta_2 t_2 - \sin\alpha_3\sin\beta_3 t_3 \\ -\sin\alpha_1\sin\beta_1 - \sin\alpha_2\sin\beta_2 t_2 - \sin\alpha_3\sin\beta_3 t_3 & 1 + t_2^2 + t_3^2 - \cos^2\beta_1 - \cos^2\beta_2 t_2^2 - \cos^2\beta_3 t_3^2 \end{bmatrix} \\
&= \begin{bmatrix} \sin^2\alpha_1 + \sin^2\alpha_2 + \sin^2\alpha_3 & -\sin\alpha_1\sin\beta_1 - \sin\alpha_2\sin\beta_2 t_2 - \sin\alpha_3\sin\beta_3 t_3 \\ -\sin\alpha_1\sin\beta_1 - \sin\alpha_2\sin\beta_2 t_2 - \sin\alpha_3\sin\beta_3 t_3 & \sin^2\beta_1 + \sin^2\alpha_2 {t_2}^2 + \sin^2\alpha_3 {t_3}^2 \end{bmatrix}
\end{aligned}$$

The determinant of this 2 by 2 matrix is zero. It represents the determinant condition of the full 5 by 5 matrix.

$$\begin{aligned}
&\det\left(M_d - M_c M_a^{-1} M_b\right) \\
&= \left(\sin^2\alpha_1 + \sin^2\alpha_2 + \sin^2\alpha_3\right)\left(\sin^2\beta_1 + \sin^2\beta_2 {t_2}^2 + \sin^2\beta_3 {t_3}^2\right) \\
&\qquad\qquad - \left(\sin\alpha_1 \sin\beta_1 + \sin\alpha_2 \sin\beta_2 t_2 + \sin\alpha_3 \sin\beta_3 t_3\right)^2 \\
&= \sin^2\beta_1 \sin^2\alpha_2 - 2t_2 \sin\alpha_1 \sin\beta_1 \sin\alpha_2 \sin\beta_2 + {t_2}^2 \sin^2\alpha_1 \sin^2\beta_2 \\
&\qquad\qquad + \sin^2\beta_1 \sin^2\alpha_3 - 2t_3 \sin\alpha_1 \sin\beta_1 \sin\alpha_3 \sin\beta_3 + {t_3}^2 \sin^2\alpha_1 \sin^2\beta_3 \\
&\qquad\qquad + {t_2}^2 \sin^2\beta_2 \sin^2\alpha_3 - 2t_2 t_3 \sin\alpha_2 \sin\beta_2 \sin\alpha_3 \sin\beta_3 + {t_3}^2 \sin^2\alpha_2 \sin^2\beta_3 \\
&= \left(\sin\alpha_2 \sin\beta_1 - t_2 \sin\alpha_1 \sin\beta_2\right)^2 + \left(\sin\alpha_3 \sin\beta_1 - t_3 \sin\alpha_1 \sin\beta_3\right)^2 \\
&\qquad\qquad + \left(t_2 \sin\alpha_3 \sin\beta_2 - t_3 \sin\alpha_2 \sin\beta_3\right)^2
\end{aligned}$$

The expression of determinant of $A_{\Lambda 4}^T A_{\Lambda 4}$ has a quadratic format.

$$\begin{aligned}
\det\left(A_{\Lambda 4}^T A_{\Lambda 4}\right) = &\left(\sin\alpha_2 \sin\beta_1 - t_2 \sin\alpha_1 \sin\beta_2\right)^2 + \left(\sin\alpha_3 \sin\beta_1 - t_3 \sin\alpha_1 \sin\beta_3\right)^2 \\
&+ \left(t_2 \sin\alpha_3 \sin\beta_2 - t_3 \sin\alpha_2 \sin\beta_3\right)^2
\end{aligned} \tag{A.11}$$

So the determinant condition gives the equations below.

$$\begin{aligned}
\det = 0 \Rightarrow &\sin\alpha_2 \sin\beta_1 - t_2 \sin\alpha_1 \sin\beta_2 = 0 \\
&\sin\alpha_3 \sin\beta_1 - t_3 \sin\alpha_1 \sin\beta_3 = 0 \\
&t_2 \sin\alpha_3 \sin\beta_2 - t_3 \sin\alpha_2 \sin\beta_3 = 0
\end{aligned} \tag{A.12}$$

A.4 Derivations, Expressions and Derivatives of the trilinear equation

The derivations, closed form expressions and the derivatives are displayed here to let users implement them easily.

A.4.1 Derivation of the outer product

This section gives the derivation of outer product equation (3.23) . The four normal vectors are

$$\begin{aligned}
\mathbf{a} = \mathbf{x}_1 \times R_2^T \mathbf{x}_2 \quad \mathbf{b} = \widetilde{\mathbf{C}}_2 \times R_2^T \mathbf{x}_2 \\
\mathbf{c} = \mathbf{x}_1 \times R_3^T \mathbf{x}_3 \quad \mathbf{d} = \widetilde{\mathbf{C}}_3 \times R_3^T \mathbf{x}_3.
\end{aligned}$$

They can be noted in the below expressions, considering $\mathbf{u} \times \mathbf{v} = [\mathbf{u}]_\times \mathbf{v} = -[\mathbf{v}]_\times \mathbf{u}$.

$$\begin{aligned}
\mathbf{a} = -\left[R_2^T \mathbf{x}_2\right]_\times \mathbf{x}_1 \quad \mathbf{b} = -\left[R_2^T \mathbf{x}_2\right]_\times \widetilde{\mathbf{C}}_2 \\
\mathbf{c} = -\left[R_3^T \mathbf{x}_3\right]_\times \mathbf{x}_1 \quad \mathbf{d} = -\left[R_3^T \mathbf{x}_3\right]_\times \widetilde{\mathbf{C}}_3.
\end{aligned}$$

The outer product, $\mathbf{ad}^T = \mathbf{bc}^T$, is the one below,

$$\left[R_2^T\mathbf{x}_2\right]_\times \mathbf{x}_1(\left[R_3^T\mathbf{x}_3\right]_\times \widetilde{\mathbf{C}}_3)^T = \left[R_2^T\mathbf{x}_2\right]_\times \widetilde{\mathbf{C}}_2(\left[R_3^T\mathbf{x}_3\right]_\times \mathbf{x}_1)^T$$

The following steps are given here.

$$\left[R_2^T\mathbf{x}_2\right]_\times \mathbf{x}_1(\widetilde{\mathbf{C}}_3)^T(\left[R_3^T\mathbf{x}_3\right]_\times)^T = \left[R_2^T\mathbf{x}_2\right]_\times \widetilde{\mathbf{C}}_2(\mathbf{x}_1)^T(\left[R_3^T\mathbf{x}_3\right]_\times)^T$$

Moving the right part to the left gives the expression below.

$$\begin{aligned}
\left[R_2^T\mathbf{x}_2\right]_\times \mathbf{x}_1(\widetilde{\mathbf{C}}_3)^T(\left[R_3^T\mathbf{x}_3\right]_\times)^T - \left[R_2^T\mathbf{x}_2\right]_\times \widetilde{\mathbf{C}}_2(\mathbf{x}_1)^T(\left[R_3^T\mathbf{x}_3\right]_\times)^T &= [0]_{3\times3} \\
\left[R_2^T\mathbf{x}_2\right]_\times (\mathbf{x}_1\widetilde{\mathbf{C}}_3^T - \widetilde{\mathbf{C}}_2\mathbf{x}_1^T)\left[R_3^T\mathbf{x}_3\right]_\times &= [0]_{3\times3} \\
R_2^T\left[\mathbf{x}_2\right]_\times R_2(\mathbf{x}_1\widetilde{\mathbf{C}}_3^T - \widetilde{\mathbf{C}}_2\mathbf{x}_1^T)R_3^T\left[\mathbf{x}_3\right]_\times R_3 &= [0]_{3\times3}
\end{aligned}$$

The expression of outer product(3.23) is got in the final row.

$$\left[\mathbf{x}_2\right]_\times R_2(\mathbf{x}_1\widetilde{\mathbf{C}}_3^T - \widetilde{\mathbf{C}}_2\mathbf{x}_1^T)R_3^T\left[\mathbf{x}_3\right]_\times = [0]_{3\times3}$$

Then the expression of outer product(3.24) is got.

A.4.2 Derivation of dependency in the outer product

This section shows the dependency in the outer product. The four normal vectors are

$$\begin{aligned}
\mathbf{a} &= \mathbf{x}_1 \times R_2^T\mathbf{x}_2 \quad & \mathbf{b} &= \widetilde{\mathbf{C}}_2 \times R_2^T\mathbf{x}_2 \\
\mathbf{c} &= \mathbf{x}_1 \times R_3^T\mathbf{x}_3 \quad & \mathbf{d} &= \widetilde{\mathbf{C}}_3 \times R_3^T\mathbf{x}_3.
\end{aligned}$$

The $\mathbf{x}_1 = [x, y, 1]^T$ and $R_2^T\mathbf{x}_2 = [u, v, w]$ give the equation below.

$$\mathbf{a} = \left[\mathbf{x}_1\right]_\times R_2^T\mathbf{x}_2 = \begin{bmatrix} 0 & -1 & y \\ 1 & 0 & -x \\ -y & x & 0 \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} v - yw \\ -u + xw \\ uy - vx \end{bmatrix} = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} \tag{A.13}$$

The above equation shows that $a_3 = (-x)a_1 + (-y)a_2$. The expression of $\mathbf{c}$ shows that $c_3 = (-x)c_1 + (-y)c_2$. The $\mathbf{a} \parallel \mathbf{b}$ and $\mathbf{c} \parallel \mathbf{d}$ give $b_3 = (-x)b_1 + (-y)b_2$ and $d_3 = (-x)d_1 + (-y)d_2$. Then the outer product $\mathbf{ad}^T = \mathbf{bc}^T$

$$\begin{bmatrix} a_1d_1 & a_1d_2 & a_1d_3 \\ a_2d_1 & a_2d_2 & a_2d_3 \\ a_3d_1 & a_3d_2 & a_3d_3 \end{bmatrix} = \begin{bmatrix} b_1c_1 & b_1c_2 & b_1c_3 \\ b_2c_1 & b_2c_2 & b_2c_3 \\ b_3c_1 & b_3c_2 & b_3c_3 \end{bmatrix},$$

has the expression below.

$$
\begin{bmatrix} a_1d_1 & a_1d_2 & -xa_1d_1 - ya_1d_2 \\ a_2d_1 & a_2d_2 & -xa_2d_1 - ya_2d_2 \\ -xa_1d_1 - ya_2d_1 & -xa_1d_2 - ya_2d_2 & x^2a_1d_1 + xya_1d_2 + xya_2d_1 + y^2a_2d_2 \end{bmatrix}
$$

$$
=
$$

$$
\begin{bmatrix} b_1c_1 & b_1c_2 & -xb_1c_1 - yb_1c_2 \\ b_2c_1 & b_2c_2 & -xb_2c_1 - yb_2c_2 \\ -xb_1c_1 - yb_2c_1 & -xb_1c_2 - yb_2c_2 & x^2b_1c_1 + xyb_1c_2 + xyb_2c_1 + y^2b_2c_2 \end{bmatrix}.
$$

The above expression shows the dependency in the outer product. Then $a_1d_1 = b_1c_1$, $a_1d_2 = b_1c_2$, $a_2d_1 = b_2c_1$ and $a_2b_2 = c_2d_2$ can guarantee the other five equations.

A.4.3 Expressions of the trilinear equations

The trilinear equation (3.22) for the frame triplet (o, p, q) using the notations $\mathbf{v}_o = R_o^T\mathbf{x}_o$, $\mathbf{v}_p = R_p^T\mathbf{x}_p$ and $\mathbf{v}_q = R_q^T\mathbf{x}_q$. This equation is noted as $T_{o,p,q}$. And the three base vectors are $\mathbf{e}_1$, $\mathbf{e}_2$, and $\mathbf{e}_3$.

$$
\mathbf{v}_p^T\Big([\mathbf{v}_o]_\times N\left[\widetilde{\mathbf{C}}_q - \widetilde{\mathbf{C}}_o\right]_\times - \left[\widetilde{\mathbf{C}}_p - \widetilde{\mathbf{C}}_o\right]_\times N[\mathbf{v}_o]_\times\Big)\mathbf{v}_q = 0 \tag{A.14}
$$

This equation is written in the following inner product form of two three dimensional vectors.

$$
\begin{aligned}
&\left[\mathbf{v}_p^TH_N^1\mathbf{v}_q, \mathbf{v}_p^TH_N^2\mathbf{v}_q, \mathbf{v}_p^TH_N^3\mathbf{v}_q\right]\mathbf{v}_o = 0 \\
\text{and}\quad &H_N^{\text{i}} = [\mathbf{e}_{\text{i}}]_\times N\left[\widetilde{\mathbf{C}}_q - \widetilde{\mathbf{C}}_o\right]_\times - \left[\widetilde{\mathbf{C}}_p - \widetilde{\mathbf{C}}_o\right]_\times N[\mathbf{e}_{\text{i}}]_\times.
\end{aligned} \tag{A.15}
$$

There are the closed form expressions of the H_N^{i}, respect to the different N matrices. The three camera positions are noted as $\widetilde{\mathbf{C}}_o = [x_o, y_o, z_o]^T$,$\widetilde{\mathbf{C}}_p = [x_p, y_p, z_p]^T$, and $\widetilde{\mathbf{C}}_q = [x_q, y_q, z_q]^T$.

When $N = I$, the three matrices H_N^{i} are

$$
H_N^1 = \begin{bmatrix} 0 & y_o - y_p & z_o - z_p \\ y_q - y_o & x_p - x_q & 0 \\ z_p - z_o & 0 & x_p - x_q \end{bmatrix},
$$

$$
H_N^2 = \begin{bmatrix} y_p - y_q & x_q - x_o & 0 \\ x_o - x_p & 0 & z_o - z_p \\ 0 & z_q - z_o & y_p - y_q \end{bmatrix},
$$

$$
H_N^3 = \begin{bmatrix} z_p - z_q & 0 & x_q - x_o \\ 0 & z_p - z_q & y_q - y_o \\ x_o - x_p & y_o - y_p & 0 \end{bmatrix}.
$$

When $N = [\mathbf{e}_1]_\times$, the three matrices H_N^{i} are

$$H_N^1 = \begin{bmatrix} 0 & z_o - z_p & y_p - y_o \\ z_o - z_q & 0 & x_q - x_p \\ y_q - y_o & x_p - x_q & 0 \end{bmatrix},$$
$$H_N^2 = \begin{bmatrix} z_p + z_q - 2z_o & 0 & x_o - x_q \\ 0 & 0 & 0 \\ x_o - x_p & 0 & 0 \end{bmatrix},$$
$$H_N^3 = \begin{bmatrix} 2y_o - y_p - y_q & x_q - x_o & 0 \\ x_p - x_o & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

When $N = [\mathbf{e}_2]_\times$, the three matrices H_N^{i} are

$$H_N^1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 2z_o - z_p - z_q & y_q - y_o \\ 0 & y_p - y_o & 0 \end{bmatrix},$$
$$H_N^2 = \begin{bmatrix} 0 & z_q - z_o & y_p - y_q \\ z_p - z_o & 0 & x_o - x_p \\ y_q - y_p & x_o - x_q & 0 \end{bmatrix},$$
$$H_N^3 = \begin{bmatrix} 0 & y_o - y_p & 0 \\ y_o - y_q & x_p + x_q - 2x_o & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

When $N = [\mathbf{e}_3]_\times$, the three matrices H_N^{i} are

$$H_N^1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & z_o - z_p \\ 0 & z_o - z_q & y_p + y_q - 2y_o \end{bmatrix},$$
$$H_N^2 = \begin{bmatrix} 0 & 0 & z_p - z_o \\ 0 & 0 & 0 \\ z_q - z_o & 0 & 2x_o - x_p - x_q \end{bmatrix},$$
$$H_N^3 = \begin{bmatrix} 0 & z_q - z_p & y_o - y_q \\ z_p - z_q & 0 & x_q - x_o \\ y_o - y_p & x_p - x_o & 0 \end{bmatrix}.$$

A.4.4 Derivatives of the trilinear equations

The derivatives use the Lie algebra knowledge. A three-dimensional vector ϕ defines a rotation matrix $R = exp([\phi]_\times)$. For example, R_o is $exp([\phi_o]_\times)$. The two useful derivative equations are given below.

$$\frac{\partial R^T \mathbf{v}}{\partial \phi} = R^T [\mathbf{v}]_\times$$
$$\frac{\partial R \mathbf{v}}{\partial \phi} = -[R\mathbf{v}]_\times$$

In the above equations, the$\mathbf{v}$ is a three-dimensional vector. They compute the derivatives of the ϕ respect to the $R^T\mathbf{v}$ and $R\mathbf{v}$. Each derivative is a three by three matrix. After getting the increment $\Delta\phi$, the ϕ the is updated using the approximate Baker-Campbell-Hausdorff(BCH) formula.

The equation (A.15) is written in the following way, in order to brief the expressions.

$$\begin{bmatrix} a_1 & a_2 & a_3 \end{bmatrix} \mathbf{v}_o = 0$$
$$a_{\mathrm{i}} = \mathbf{v}_p^T H_N^{\mathrm{i}} \mathbf{v}_q$$
$$\mathbf{v}_a = \begin{bmatrix} a_1 & a_2 & a_3 \end{bmatrix}^T$$

The below expression of H matrix gives the short expression.

$$H = H_N^1 v_{o1} + H_N^2 v_{o2} + H_N^3 v_{o3},$$
$$\text{and } ,\mathbf{v}_o = \begin{bmatrix} v_{o1} & v_{o2} & v_{o3} \end{bmatrix}^T$$

The $T_{o,p,q}$ equals to $\mathbf{v}_p^T H \mathbf{v}_q = 0$. There are some useful vectors in the derivative computation.

$$\mathbf{rv}_a = R_o \mathbf{v}_a$$
$$\mathbf{hv}_p = H^T \mathbf{v}_p \qquad \mathbf{hv}_q = H \mathbf{v}_q$$
$$\mathbf{rhv}_p = R_q \mathbf{hv}_p \qquad \mathbf{rhv}_q = R_p \mathbf{hv}_q$$

The $\mathbf{hv}_p$ and $\mathbf{rhv}_p$ give expression below.

$$T_{o,p,q} = \mathbf{hv}_p^T \mathbf{v}_q = \mathbf{hv}_p^T R_q^T \mathbf{x}_q = \mathbf{rhv}_p^T \mathbf{x}_q = 0$$

The $\mathbf{hv}_q$ and $\mathbf{rhv}_q$ give expression below.

$$T_{o,p,q} = \mathbf{v}_p^T \mathbf{hv}_q = \mathbf{x}_p^T R_p \mathbf{hv}_q = \mathbf{x}_p^T \mathbf{rhv}_q = 0$$

The $\mathbf{v}_a$ and $\mathbf{rv}_a$ give expression below.

$$T_{o,p,q} = \mathbf{v}_a^T \mathbf{v}_o = \mathbf{v}_a^T R_o^T \mathbf{x}_o = \mathbf{x}_o^T \mathbf{rv}_a = 0$$

The derivatives are computed by the above vectors very easy. The derivatives of image measurements, camera rotations and camera positions are given below. The three image point coordinates are noted as $\mathbf{x}_o = [p_{xo}, p_{yo}, 1]^T$,$\mathbf{x}_p = [p_{xp}, p_{yp}, 1]^T$, and $\mathbf{x}_q = [p_{xq}, p_{yq}, 1]^T$.

The derivatives of the three image measurements are given here.

$$\begin{aligned}
\frac{\partial T_{o,p,q}}{\partial p_{xo}} &= \mathbf{e}_1^T \mathbf{rv}_a & \qquad \frac{\partial T_{o,p,q}}{\partial p_{yo}} &= \mathbf{e}_2^T \mathbf{rv}_a \\
\frac{\partial T_{o,p,q}}{\partial p_{xp}} &= \mathbf{e}_1^T \mathbf{rhv}_q & \qquad \frac{\partial T_{o,p,q}}{\partial p_{yp}} &= \mathbf{e}_2^T \mathbf{rhv}_q \\
\frac{\partial T_{o,p,q}}{\partial p_{xq}} &= \mathbf{e}_1^T \mathbf{rhv}_p & \qquad \frac{\partial T_{o,p,q}}{\partial p_{yq}} &= \mathbf{e}_2^T \mathbf{rhv}_p
\end{aligned}$$

The derivatives of the three camera rotations are given here.

$$\begin{aligned}
\frac{\partial T_{o,p,q}}{\partial \phi_o} &= \frac{\partial \mathbf{x}_o^T R_o \mathbf{v}_a}{\partial \phi_o} = -\mathbf{x}_o^T [R_o \mathbf{v}_a]_\times = -\mathbf{x}_o^T [\mathbf{rv}_a]_\times \\
\frac{\partial T_{o,p,q}}{\partial \phi_p} &= \frac{\partial \mathbf{x}_p^T R_p \mathbf{hv}_q}{\partial \phi_p} = -\mathbf{x}_p^T [R_p \mathbf{hv}_q]_\times = -\mathbf{x}_p^T [\mathbf{rhv}_q]_\times \\
\frac{\partial T_{o,p,q}}{\partial \phi_q} &= \frac{\partial \mathbf{x}_q^T R_q \mathbf{hv}_p}{\partial \phi_q} = -\mathbf{x}_q^T [R_q \mathbf{hv}_p]_\times = -\mathbf{x}_q^T [\mathbf{rhv}_p]_\times
\end{aligned}$$

The derivatives of the three camera positions have same format. Here gives the derivatives of first camera positions only, where $\widetilde{\mathbf{C}}_o = [x_o, y_o, z_o]^T$.

$$\begin{aligned}
\frac{\partial T_{o,p,q}}{\partial x_o} &= \begin{bmatrix} \mathbf{v}_p^T \frac{\partial H_N^1}{\partial x_o} \mathbf{v}_q & \mathbf{v}_p^T \frac{\partial H_N^2}{\partial x_o} \mathbf{v}_q & \mathbf{v}_p^T \frac{\partial H_N^3}{\partial x_o} \mathbf{v}_q \end{bmatrix} \mathbf{v}_o \\
\frac{\partial T_{o,p,q}}{\partial y_o} &= \begin{bmatrix} \mathbf{v}_p^T \frac{\partial H_N^1}{\partial y_o} \mathbf{v}_q & \mathbf{v}_p^T \frac{\partial H_N^2}{\partial y_o} \mathbf{v}_q & \mathbf{v}_p^T \frac{\partial H_N^3}{\partial y_o} \mathbf{v}_q \end{bmatrix} \mathbf{v}_o \\
\frac{\partial T_{o,p,q}}{\partial z_o} &= \begin{bmatrix} \mathbf{v}_p^T \frac{\partial H_N^1}{\partial z_o} \mathbf{v}_q & \mathbf{v}_p^T \frac{\partial H_N^2}{\partial z_o} \mathbf{v}_q & \mathbf{v}_p^T \frac{\partial H_N^3}{\partial z_o} \mathbf{v}_q \end{bmatrix} \mathbf{v}_o
\end{aligned}$$

In the normalized equation (3.52), the function is $f(x)_{o,p,q} = \frac{T_{o,p,q}}{\|\widetilde{\mathbf{C}}_p - \widetilde{\mathbf{C}}_q\|}$. If note the $d_{p,q} = \|\widetilde{\mathbf{C}}_p - \widetilde{\mathbf{C}}_q\|$, the derivative of $f(x)_{o,p,q}$ is

$$\frac{\partial f(x)_{o,p,q}}{\partial x} = \frac{1}{d_{p,q}} \frac{\partial T_{o,p,q}}{\partial x} - \frac{T_{o,p,q}}{d_{p,q}^2} \frac{\partial d_{p,q}}{\partial x}. \tag{A.16}$$

A.4.5 Changes brought about by Forward rotation

In order to work with the collinearity equation together, the forward rotation is used in camera projection matrix. For different reference system, vector ϕ in $R_o = exp([\phi_o]_\times)$ is different. Then the camera projection is $\mathbf{x} \sim R_f^T [I| - \widetilde{\mathbf{C}}]\mathbf{X}$.

In the equation (A.15), $\mathbf{v}_o = R_o \mathbf{x}_o$, $\mathbf{v}_p = R_p \mathbf{x}_p$ and $\mathbf{v}_q = R_q \mathbf{x}_q$. The other changes are $\mathbf{rv}_a = R_o^T \mathbf{v}_a$, $\mathbf{rhv}_p = R_q^T \mathbf{hv}_p$, and $\mathbf{rhv}_q = R_p^T \mathbf{hv}_q$.

It will cause the changes in the derivatives of the camera rotations. The new derivatives are given here.

$$\frac{\partial T_{o,p,q}}{\partial \phi_o} = \frac{\partial \mathbf{v}_a^T R_o \mathbf{x}_o}{\partial \phi_o} = -\mathbf{v}_a^T [R_o \mathbf{x}_o]_\times = -\mathbf{v}_a^T [\mathbf{v}_o]_\times$$

$$\frac{\partial T_{o,p,q}}{\partial \phi_p} = \frac{\partial \mathbf{h} \mathbf{v}_q^T R_p \mathbf{x}_p}{\partial \phi_p} = -\mathbf{h}\mathbf{v}_q^T [R_p \mathbf{x}_p]_\times = -\mathbf{h}\mathbf{v}_q^T [\mathbf{v}_p]_\times$$

$$\frac{\partial T_{o,p,q}}{\partial \phi_q} = \frac{\partial \mathbf{h} \mathbf{v}_p^T R_q \mathbf{x}_q}{\partial \phi_q} = -\mathbf{h}\mathbf{v}_p^T [R_q \mathbf{x}_q]_\times = -\mathbf{h}\mathbf{v}_p^T [\mathbf{v}_q]_\times$$

Other expressions of derivatives are same.

VITA

Chen Ma received her Bachelor of Engineering Degree in Photogrammetry and Remote Sensing at Information Engineering University, Zhengzhou, China. He received a non-thesis Master of Engineering degree from Lyles School of Civil Engineering at Purdue University in May 2014. He was awarded the Roland S. Corning II Memorial Fellowship for the 2014-2015 academic year. .