

# CAMERA SELF-CALIBRATION AND ANALYSIS OF SINGULAR CASES

CHENG ZHAO LIN

(B.Eng.)

A THESIS SUBMITTED  
FOR THE DEGREE OF MASTER OF ENGINEERING  
DEPARTMENT OF MECHANICAL ENGINEERING  
NATIONAL UNIVERSITY OF SINGAPORE

2003

## Acknowledgements

The work described in this thesis is a cooperation project with The French National Institute for Research in Computer Science and Control (INRIA). First, I am very grateful to my supervisors Professor Poo Aun Neow and Professor Peter C.Y. Chen for their consistent encouragement and advice during my two-year study in National University of Singapore. More thanks also go to Prof. Chen who read the whole thesis and gave many suggestions to revise it. Without their kind support, the work and the thesis would not be completed.

I would like to express my deep gratitude to Dr. Peter Sturm, who kindly arranged my visit to INRIA and enabled the cooperation. With the aid of emails, we exchanged a lot of creative ideas, which immensely filled stuffs in my work. His strict attitude to research also made me never be relaxed with some minor success. Without his help, my work would not be published.

Here, I am also appreciated for a lot of useful discussion with a lot of people in Control and Mechatronics Lab such as Duan Kai Bo, Ankur Dahnik, Tay Wee Beng, Zhang Zheng Hua and Sun Jie. They never conserved in their advice.

Last but not least, I thank my dear parents and Li Min for their constant encouragement, understanding and support. These helped me get through many harsh days.

# Table of Contents

Summary.....	vi
List of Tables.....	vii
List of Figures.....	viii
Chapter 1. Introduction.....	1
1.1 Motivation.....	1
1.2 From 2D images to 3D model.....	2
1.2.1 Image feature extraction and matching.....	2
1.2.2 Structure from motion.....	3
1.2.3 Self-calibration.....	4
1.2.4 Dense 3D model.....	5
1.3 Main contribution.....	5
1.4 Thesis outline.....	6
Chapter 2. Projective Geometry.....	8
2.1 Introduction.....	8
2.2 Duality.....	9
2.3 Projective 2D and 3D geometry.....	10
2.3.1 The 2D projective plane.....	10
2.3.2 The 3D projective space.....	10
2.3.3 The plane at infinity.....	11

2.3.4 Conics and quadrics .....	12
2.4 Conclusion.....	13
<b>Chapter 3. Two-View Geometry .....</b>	<b>14</b>
3.1 Camera model .....	14
3.1.1 Perspective projection camera model.....	14
3.1.2 Intrinsic parameters.....	16
3.1.3 Extrinsic parameters.....	16
3.1.4 Radial distortion.....	17
3.2 Epipolar geometry and the fundamental matrix .....	18
3.2.1 Epipolar geometry.....	18
3.2.2 The fundamental matrix .....	19
3.3 Recovery of camera matrix from the fundamental matrix.....	21
3.3.1 Canonical form of camera matrices of a stereo rig .....	21
3.3.2 Camera matrices obtained from $F$ .....	22
3.4 The fundamental matrix computation.....	22
3.4.1 Linear approaches for $F$ computation .....	23
3.4.2 Nonlinear approaches for $F$ computation.....	25
3.4.3 Robust estimation of the fundamental matrix .....	26
3.5 The stratification of the 3D geometry .....	29
3.5.1 The 3D projective structure.....	29
3.5.2 The 3D affine structure .....	29
3.5.3 The 3D metric structure .....	30
3.5.4 Camera self-calibration, the bond between projective reconstruction and metric reconstruction .....	32

Chapter 4. Camera self-calibration .....	34
4.1 Kruppa's equations based camera self-calibration .....	34
4.1.1 Absolute conic and image of the absolute conic .....	34
4.1.2 Kruppa's equations .....	37
4.1.3 Simplified Kruppa's equations .....	39
4.2 Review of Camera self-calibration .....	41
4.2.1 Self-calibration for stationary cameras .....	41
4.2.2 Kruppa's equations based self-calibration for two special motions .....	42
4.2.3 Self-calibration from special objects.....	43
4.3 Focal length self-calibration from two images .....	45
Chapter 5 Singular cases analyses .....	47
5.1 Critical motion sequences for camera self-calibration .....	48
5.1.1 Potential absolute conics .....	49
5.1.2 PAC on the plane at infinity.....	49
5.1.3 PAC not on the plane at infinity.....	50
5.1.4 Useful critical motion sequences in practice.....	51
5.2 Singular cases for the calibration algorithm in Section 4.3 .....	52
5.2.1 Generic singularities .....	52
5.2.2 Heuristic interpretation of generic singularities.....	53
5.2.3 Algebraic interpretation of generic singularities.....	55
5.2.4 Conclusion .....	58
Chapter 6 Experiment results.....	60
6.1 Experiment involving synthetic object .....	60

6.1.1 Synthetic object and images.....	60
6.1.2 Performance with respect to Gaussian noise level.....	61
6.1.3 Detecting different singular cases for the linear and quadratic equations.....	62
6.2 Experiment involving actual images.....	65
6.2.1 Camera setup.....	66
6.2.2 Experiment involving images taken from a special object .....	66
6.2.3 Calibration using arbitrary scenes.....	74
6.3 Conclusion.....	78
Chapter 7 Conclusion .....	80
Reference .....	81
Appendix A .....	87
Orthogonal least squares problem.....	87
Appendix B.....	88
B.1 The equivalent form of the semi-calibrated fundamental matrix.....	88
B.2 Coplanar optical axes .....	89
B.3 Non-coplanar optical axes .....	92
B.3.1 Linear equations .....	92
B.3.2 Quadratic equation.....	94

## Summary

Obtaining a 3D model for the world is one of the main goals of computer vision. The task to achieve this goal is usually divided into several modules, i.e., projective reconstruction, affine reconstruction, metric reconstruction, and Euclidean reconstruction. Camera self-calibration, which is one key step among them, links the so-called projective and metric reconstruction. However, a lot of existed self-calibration algorithms are fairly unstable and thus fail to take up this role. The main reason is that singular cases are not rigorously detected.

In this thesis, a new camera self-calibration approach based on Kruppa's equations is proposed. We assume only the focal length is unknown and constant, the Kruppa's equations are then decomposed as two linear and one quadratic equations. All of generic singular cases, which are nearly correspondent to algebraically singular cases for those equations are fully derived and analyzed. We then thoroughly carry out experiments and find that the algorithm is quite stable and easy to implement when the generic singular cases are excluded.

## List of Tables

Table 6.1: Calibration results with respect to the principal point estimation .....	68
Table 6.2: Experiment considering the stability of this algorithm.....	70
Table 6.3: Reconstruction results using calibrated focal length .....	74
Table 6.4: Results calibrated from images containing 3 cups .....	75
Table 6.5: Results calibrated from images containing a building.....	76



## List of Figures

Figure 2.1: Line-point dual figure in projective 2D geometry.....	9
Figure 3.1: The pinhole camera model .....	15
Figure 3.2: The Euclidean transformation between the world coordinate system and the camera coordinate system .....	17
Figure 3.3: Epipolar geometry .....	19
Figure 3.4: Different structures recovered on different layers of 3D geometry .....	31
Figure 4.1: Absolute conic and its image.....	37
Figure 5.1: Illustration of critical motion sequences. (a) Orbital motion. (b) Rotation about parallel axes and arbitrary translation. (c) Planar motion (d) Pure rotations (not critical for self-calibration but for the scene reconstruction). .....	52
Figure 5.2: Possible camera center positions when the PAC is not on $\Pi_\infty$ . (a) The PAC is a proper virtual circle. All the camera centers are on the line L. (b) The PAC is a proper virtual ellipse. All the camera centers are on a pair of ellipse/hyperbola. ....	54
Figure 5.3: Illustration of the equidistant case (arrows show the directions of camera's optical axes) .....	55
Figure 5.4: Configuration of non-generic singularity for the linear equations.....	58
Figure 6.1: The synthetic object.....	61
Figure 6.2: Relative error of focal length with respect to Gaussian noise level .....	62
Figure 6.3: Coordinates of two cameras .....	63
Figure 6.4: Coplanar optical axes (neither parallel nor equidistance case) .....	64
Figure 6.5: The two camera centers are near to be equidistant from the intersection of the two optical axes.....	65
Figure 6.6: The two optical axes are near parallel .....	65
Figure 6.7: Some images of the calibration grid.....	67
Figure 6.8: Effect of the principal point estimation on the focal length calibration.....	69
Figure 6.9: The middle plane .....	70
Figure 6.10: Sensitivity of focal length with respect to the angle $c$ .....	71

Figure 6.11: Images of three cups.....	76
Figure 6.12: some images of a building.....	77
Figure 6.13: The reconstructed cup. First row: general appearance of the scene, once with overlaid triangular mesh. Second row: rough top view of cups and two close-ups of the plug in the background (rightmost image shows the near coplanarity of the reconstruction). Third row: top views of two of the cups, showing that their cylindrical shape has been recovered.....	78

## Nomenclature

In order to enhance readability of the thesis, a few notations are used throughout the thesis. Generally, 3D points are represented in capital form and their images are the same letters of low case form. Vectors are column vectors embraced in square brackets. Homogeneous coordinates differ with their correspondent inhomogeneous dummies by adding a "~" on their heads.

$\times$	cross product
$\cdot$	dot product
$A^T$	transpose of the matrix $A$
$P$	the projection matrix
$\Pi$	world plane (4-vector)
$\Pi_\infty$	the plane at infinity
$l$	image line (3-vector)
$A$	camera intrinsic parameter matrix
$F$	fundamental matrix
AC	absolute conic
IAC	image of the absolute conic
DIAC	dual of IAC
$\ v\ $	Euclidean norm of the vector $v$
$\sim$	equivalent up to scale

# Chapter 1. Introduction

## 1.1 Motivation

Computer vision system attempts to mimic human being's vision. They first appear in robotics applications. The commonly accepted computational theory of vision proposes that constructing a model of the world is a prerequisite for a robot to carry out any visual task [22]. Based on such a theory, obtaining 3D model becomes one of the major goals of the computer vision community.

Increased interests on application of computer vision come from entertainment and media industry recently. One example is that a virtual object is generated and merged into a real scene. Such application heavily depends on the availability of an accurate 3D model.

Conventionally, CAD or 3D modeling system is employed to obtain a 3D model. The disadvantage of such approaches is that the costs in terms of labor and time investment often rise to a prohibitive level. Furthermore, it is also difficult to include delicate details of a scene into a virtual object.

An alternative approach is to use images. Details of an object can be copied from images to a generated virtual object. The remaining problem is that 3D information is lost by projection. The task is then to recover the lost depth to a certain extent<sup>1</sup>.

The work reported in this thesis deals with this task of depth recovery in the reconstruction of 3D models using 2D information. Due to the limited time and space, it does not cover all the details of how to obtain a 3D model. Instead, it focuses on the

---

<sup>1</sup> Details on the different kind of reconstruction (or structure recovery) will be discussed in later chapters.

so-called camera self-calibration that is a key step for constructing 3D models using 2D images.

## 1.2 From 2D images to 3D model

As we may see in the later chapters, camera self-calibration is one of important steps in automatic 3D modeling. Therefore it is a logical start that we first introduce how to reconstruct a 3D model.

Although it is natural for us to perceive 3D, it is hardly so for a computer. The fundamental problems associated with such a perception task are what can be directly obtained from images and what can help a computer to find 3D information from images. Those problems are usually categorized as image feature extraction and matching.

### 1.2.1 Image feature extraction and matching

Most of us have the experience that if we look at a homogeneous object (such as a white wall), there is no way to perceive 3D. We have to depend on some distinguished features to do so. Such distinguished features may be corners, lines, curves, surfaces, and even colors. Usually, corners or points are used since they can be easily formulated by mathematics. Harris corner detector [8] shows superior performance considering the criteria of independence of camera pose and illumination change [27]. Matching between two images is a difficult task in image processing since a little change of conditions (such as illumination or camera pose) may produce very different matches. Hence current cross-correlation approaches that are widely employed often assume that images are not very different from each other.

### 1.2.2 Structure from motion

After some image correspondences (i.e. pairwise matches) are obtained, the next step toward 3D model is to recover the scene's structure. The word "structure" we use here does not have the same meaning as what we imagine in the Euclidean world. The connotation of structure in the field of computer vision depends on the different layers of 3D geometry. This stratification of 3D geometry will be discussed in detail in the later chapters. We just give a brief introduction here. Generally, if no information other than the image correspondences is available, a projective reconstruction can be done at this stage. In fact, as we may see in Chapter 3, the structure is recovered up to a  $4 \times 4$  arbitrary projective transformation matrix. However, when the camera's intrinsic parameter matrix is known, the structure could be recovered up to an arbitrary similarity transformation. Such similarity transformation has one degree of freedom more than a Euclidean transformation, which is determined by a rotation and a translation. That one-degree of freedom is exactly the yardstick to measure the real object's dimension. At this stage, this process of structure recovery is called the metric reconstruction.

Early work on structure from motion assumes that the camera intrinsic parameter matrix is known. Based on this assumption, camera motion and the scene's structure can be recovered from two images [19], [40] or from image sequences [30], [34]. A further assumption of the affine camera model can give another robust algorithm [35].

Since the so-called fundamental matrix was obtained by Faugeras [5] and Hartley [9], uncalibrated structure from motion has been drawing extensive attention from researchers. The fundamental matrix computation is the starting point to conduct such research. Two papers [36, 43] represent the existing state-of-the-art research in this area. After the fundamental matrix is obtained, the camera matrices can be constructed with some extent of ambiguity. This will be discussed in detail in Chapter 3.

### 1.2.3 Self-calibration

Camera self-calibration is the crux that links the projective and metric reconstruction. Self-calibration means that the cameras can be calibrated just from images without any calibration pattern of known 3D information. It is very interesting since in the last subsection, we note that only the projective reconstruction can be obtained from images. However, the camera's intrinsic parameter matrix is exactly constrained by the so-called images of the absolute conic (IAC), which in fact can be obtained from images through the so-called Kruppa's equations. We will present this in detail in Chapter Four.

Faugeras and Maybank initiated the research on Kruppa's equations based camera self-calibration [6]. Hartley then conducted a singular value decomposition (SVD) based simplification of the Kruppa's equations [13]. These simplified Kruppa's equations clearly show that two images give rise to two independent equations that impose constraints on the camera's intrinsic parameter matrix. Since the camera's intrinsic parameter matrix has 5 unknown parameters, at least three images are needed (One fundamental matrix introduces two independent Kruppa's equations. Three images lead to three fundamental matrices and then six equations would be obtained if no degeneration occurs).

A lot of algorithms on camera self-calibration were proposed [45] [46] in the past ten years. However, the calibrated results seemed not so satisfactory [47]. Recently, a lot of researchers delved into the existing problems in camera self-calibration. Sturm showed that some special image sequences could result in incorrect constraints on the camera parameter matrix [31, 32]. The corresponding camera motions are then called critical motion sequences [31]. The geometric configurations corresponding to critical motion sequences are called the singular cases (or the singularities) of a calibration al-

gorithm in this thesis. In addition to the analyses of the critical motion sequence analyses, some researchers also found constraints on the camera's intrinsic parameters would yield more robust results [24] [48]. We propose that if some camera's intrinsic parameters are known first, singularities of a calibration algorithm would be discovered as a whole [49]. This part of work on singularities will be discussed in Chapter 5.

### 1.2.4 Dense 3D model

The structure recovered from the approaches discussed in the last subsection has only restricted feature points. Those points are not sufficient for robot vision and object recognition. Hence a dense 3D model needs to be recovered. However, since after structure recovery, the geometry among cameras has been found and then it is easy to match other common points in the images. Typical matching algorithms at this stage are area-based algorithms (such as [3, 15]) and space carving algorithms (such as [17] [28]). Details can be found in the work by P. Torr [37]<sup>1</sup>.

## 1.3 Main contribution

Before we move on to present the technical details, it is essential to clarify the contribution made by the author.

1. Theoretically, our work stands on two footstones. Firstly, three calibration equations are obtained (in Section 4.3, Chapter 4). One of them is quadratic and the remaining is linear. Focal length is in the closed form in these equations, and thus solution is easy to obtain. Secondly, all of singular cases associated with the equations are described geometrically and derived algebraically (in

---

<sup>1</sup> An alternative way is by optical flow. However, it estimates the camera geometry and dense matching simultaneously and thus we don't discuss it here.



Section 5.2, Chapter 5 and Appendix B). Part of the results has been published in our paper [2].

2. Experimentally, intensive tests have been conducted both on simulated and real data (in Chapter 6). This part of work together with the theoretical work is described in the report [49].

## 1.4 Thesis outline

This thesis consists of seven chapters. Chapter 2 introduces basic concepts of projective geometry that are needed in the later part of the thesis. Since a camera is a 3D to 2D projection model, only the 2D projective plane and 3D projective space are presented in Chapter 2. The concept of duality, which is essential to the Kruppa's equations, is also introduced in this chapter. Some geometric entities such as points, lines, planes, and conics are briefly discussed as well.

Two-view geometry, which is fundamental to the new self-calibration algorithm in this thesis, is then introduced in Chapter 3. We start with the camera model, and two-view geometry (or epipolar geometry) is then established. The fundamental matrix (which is the core of two-view geometry) is then fully presented. Next, the recovery of camera matrix from the fundamental matrix is discussed. Here, we also discuss computation of the fundamental matrix. This section is essential since the fundamental matrix computation determines the performance of the calibration algorithm presented in the thesis. Finally, the stratification of the 3D geometry is presented. The role of camera self-calibration gradually emerges after such stratification.

In Chapter 4, we focus on camera self-calibration. Kruppa's equations are first introduced through the invariant of the images of the absolute conic (IAC) with respect to camera motions. A brief history of camera self-calibration is introduced, and a few

relevant important algorithms are reviewed. Our focal length calibration algorithm is then presented after the introduction of Hartley's simplification of Kruppa's equations [13].

Chapter 5 starts by discussing so-called critical motions that make camera self-calibration impossible. After then, we give heuristic and algebraic analysis of singular cases for our algorithm. Both of them nearly lead to the same results.

Both simulation and experiments with actual images are presented in Chapter 6. We show that the proposed algorithm is very stable, and the results perfectly match the analysis on singular cases of the chapter 5.

Conclusion is drawn in Chapter 7. To enhance the readability of this text, some of mathematical derivations are placed in appendices.

## Chapter 2. Projective Geometry

This chapter discusses some important concepts and properties of projective geometry. First, some basic concepts of  $n$  dimensional projective space are introduced in Section 2.1. Then in Section 2.2, the concept of duality is presented. Two important instances of projective geometry (namely the 2D projective plane and the 3D projective space) are then discussed in Section 2.3. Some important geometric entities are also presented in this section. Various background information discussed in this chapter can be found in the books by Faugeras[7], Wolfgang Bohem and Hartmut Prautzsch[1], and Semple and Kneebone[29].

### 2.1 Introduction

With the introduction of Cartesian coordinates in Euclidean geometry, geometry became closely associated with algebra. Usually, a point in  $R^n$  is given by an  $n$ -vector of coordinates, i.e.,  $X = [x_1 \dots x_n]^T$ . Homogeneous coordinates, which are footstone of projective geometry, however, add one dimension up to an  $n+1$ -vector of coordinates, i.e.,  $\tilde{X} = [\tilde{x}_1 \dots \tilde{x}_{n+1}]^T$ . Given homogeneous coordinates  $\tilde{X}$ , the Euclidean coordinates are obtained by

$$x_1 = \frac{\tilde{x}_1}{\tilde{x}_{n+1}}, \quad \dots, \quad x_n = \frac{\tilde{x}_n}{\tilde{x}_{n+1}}. \quad (2.1.1)$$

Since the relationship (2.1.1) exists, two points  $X$  and  $Y$  in projective  $n$ -space are equal if their homogeneous coordinates are related by  $x_i = \lambda y_i$ , where  $\lambda$  is a nonzero scalar. However, if  $\tilde{x}_{n+1} = 0$ , then the Euclidean coordinates go to infinity accordingly. In projective geometry, such a point is called an *ideal point* or a *point at the infinity*. The important role of such a point will be discussed in Section 2.3.2.

## 2.2 Duality

We note that the  $n$  dimensional projective space  $P^n$  can be expressed in  $n+1$ -vector homogeneous coordinates. Therefore, a hyperplane in  $P^n$ , when expressed algebraically, is in the form of  $u'x=0$ . Here,  $u$  and  $x$  are both  $n+1$ -vectors and  $u$  is the hyperplane's coordinate vector. Then the coordinates of the hyperplane span another  $n$  dimensional projective space  $P^*$  that is called the *dual space* of  $P^n$ .

If the term "point" (in the previous paragraph, it is expressed in homogeneous coordinates  $x$ ) is interchanged with "hyperplane" and correspondently "collinear" with "coincident" and "intersection" with "join", etc., then there is no way to tell the difference between the projective geometry formed by a space and by its dual space. Specifically, let us consider a line  $[1 \ 2 \ 3]^T$  in projective 2D geometry. Three points in homogeneous coordinates  $p_1 = [-1 \ -1 \ -1]^T$ ,  $p_2 = [-3 \ 0 \ 1]^T$  and  $p_3 = [5 \ -1 \ -1]^T$  are on this line. However, if we treat these three vectors as the lines' coordinate vectors, then the lines finally intersect at a point  $[1 \ 2 \ 3]^T$ . Hence, points are interchanged with lines (two dimensional hyperplane) and so are collinear and coincidence. The geometry after the interchange is the same as the geometry before the interchange. Figure 2.1 shows such dual relation.

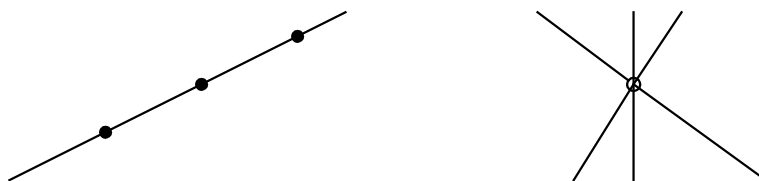


Figure 2.1: Line-point dual figure in projective 2D geometry

## 2.3 Projective 2D and 3D geometry

Projective 2D and 3D geometry are the two most important projective geometries since they correspond to 2D plane and 3D space in Euclidean geometry. In computer vision, the 2D projective plane corresponds to the geometry in 2D image plane, while projective 3D space corresponds to geometry in 3D world space.

### 2.3.1 The 2D projective plane

The 2D projective plane is the projective geometry of  $P^2$ . A point in  $P^2$  is expressed by a 3-vector  $\tilde{X} = [\tilde{x} \ \tilde{y} \ \tilde{w}]^T$ . Its Euclidean coordinates are then given by

$$x = \frac{\tilde{x}}{\tilde{w}}, \quad y = \frac{\tilde{y}}{\tilde{w}} \quad (2.3.1)$$

In  $P^2$ , a line is also represented by a 3-vector. In fact, given a line  $l$ , a point  $\tilde{X}$  is on the line if and only if the equation  $l^T \tilde{X} = 0$  holds. According to the description in the last section, here points and lines are a pair of duality.

Given two points  $\tilde{X}_1$  and  $\tilde{X}_2$ , the line  $l$  passing through these two points is written as  $l = \tilde{X}_1 \times \tilde{X}_2$  since  $\tilde{X}_1 \times \tilde{X}_2 \cdot \tilde{X}_1 = 0$  and  $\tilde{X}_1 \times \tilde{X}_2 \cdot \tilde{X}_2 = 0$ . Because of duality, two lines  $l_1$  and  $l_2$  intersect at one point  $\tilde{X} = l_1 \times l_2$ .

### 2.3.2 The 3D projective space

A point  $X$  in the projective 3D space is represented by a 4-vector  $\tilde{X} = [\tilde{x} \ \tilde{y} \ \tilde{z} \ \tilde{w}]^T$ . The corresponding Euclidean coordinates are

$$x = \frac{\tilde{x}}{\tilde{w}}, \quad y = \frac{\tilde{y}}{\tilde{w}}, \quad z = \frac{\tilde{z}}{\tilde{w}}. \quad (2.3.2)$$

Its dual buddy is a plane  $\Pi$  given by  $\Pi^T \tilde{X} = 0$ .

A line in the 3D projective space is not easy to express directly since it has four degrees of freedom. Four degrees of freedom need a homogeneous 5-vector expression. Hence, it is not easy to combine a line in 5-vector with a point and plane in 4-vector. The usual way to express a line is based on the fact that a line is the join of two points, e.g.  $l = \lambda_1 \tilde{X}_1 + \lambda_2 \tilde{X}_2$ , or dually it is the intersection of two planes, e.g.  $l = \Pi_1 \cap \Pi_2$ .

### 2.3.3 The plane at infinity

In 3D projective geometry, a point at infinity is interpreted as  $\tilde{p} = [\tilde{x} \ \tilde{y} \ \tilde{z} \ 0]^T$  in homogeneous coordinates. The plane at infinity,  $\Pi_\infty$ , consists of all the points at infinity. Hence, the homogeneous coordinates of  $\Pi_\infty$  are  $[0 \ 0 \ 0 \ 1]^T$ .

It is well known that  $\Pi_\infty$  is invariant under any affine projective transformation. An

affine transformation is in the form of  $P_{aff} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix}^T$ . The proof is given

briefly below.

**Result 2.1** The plane at infinity  $\Pi_\infty$  is invariant under the affine transformation  $P_{aff}$ .

*Proof:* A point  $X$  is in  $\Pi_\infty$  if and only if  $\Pi_\infty^T X = 0$ . Since  $\Pi_\infty^T P_{aff}^{-1} P_{aff} X = 0$ ,  $\Pi_\infty$  is transformed to  $P_{aff}^{-T} \Pi_\infty$  under an affine transformation. Therefore we have,

$$\Pi'_\infty = P_{aff}^{-T} \Pi_\infty = \begin{bmatrix} \mathbf{P}^{-T} & \mathbf{0} \\ -\mathbf{P}^{-T} \mathbf{p}^T & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \Pi_\infty, \quad (2.3.3)$$

where  $\mathbf{P}$  is the upper  $3 \times 3$  matrix of  $P_{aff}$  and  $\mathbf{p}$  is the vector of  $\mathbf{p} = [p_{14} \ p_{24} \ p_{34}]^T$ .

Since  $\Pi_\infty$  is fixed under a general affine transformation, it is the basic invariant in the affine space, which is the intermediate layer between projective space and Euclidean space. Because of this, it plays an important role in the interpretation of different kinds of reconstruction.

### 2.3.4 Conics and quadrics

**Conic** In  $P^2$ , a conic is a planar curve, which is represented by a  $3 \times 3$  symmetric matrix  $C$  up to an unknown scale factor. Points on the conic satisfy the homogeneous equation

$$S(x) = x^T C x = 0. \quad (2.3.4)$$

**Dual conic** The duality of a conic is the envelope of its tangents, which satisfy the following homogeneous equations

$$l^T C^* l = 0. \quad (2.3.5)$$

Like a conic  $C$ ,  $C^*$  is also a  $3 \times 3$  symmetric matrix up to an unknown scale factor.

**Line-conic intersection** A point on the line  $l$  can be expressed as  $x_0 + tl$ , where  $t$  is a scalar and  $x_0$  is a reference point on the line. Following the conic definition, there is

$$(x_0 + tl)^T C (x_0 + tl) = 0. \quad (2.3.6)$$

Finally the equation (2.3.6) can be expressed as

$$x_0^T C x_0 + 2tl^T C x_0 + t^2 l^T C l = 0. \quad (2.3.7)$$

Therefore, generally, a line has two intersections with a conic.

**Tangent to a conic** From the equation (2.3.7), we know that the line  $l$  is tangent to the conic  $C$  if and only if  $(l^T C x_0)^2 - (x_0^T C x_0) \cdot (l^T C l) = 0$ . If  $x_0$  is on the conic, then we have

$$l^T C x_0 = 0 \quad (2.3.8)$$

So the tangent to a conic is  $l \sim C^T l \sim C^T x_0$ , where  $\sim$  means it is equivalent up to an unknown scale factor.

**The relation between conic and dual conic** Results from the above description demonstrate that the relation between conic and dual conic is  $C^* \sim C^{-1}$  if the conic does not degenerate.

**Quadric** A quadric  $Q$  is a set of points satisfying a homogeneous quadratic equation. Therefore, a conic is a special case of quadric in  $P^2$ . Like conics, a quadric in  $P^n$  can be represented by a  $(n+1) \times (n+1)$  symmetric matrix. Hence, its dual is also a  $(n+1) \times (n+1)$  symmetric matrix. In  $P^3$ , a plane tangent to a quadric is then determined by  $\Pi = Q^T \Pi$ . Similarly, the dual of a quadric  $Q^*$  satisfies  $Q^* \sim Q^{-1}$ .

## 2.4 Conclusion

In this chapter, some basic concepts of projective geometry are introduced. These concepts provide the background for the discussion of two-view geometry.



## Chapter 3. Two-View Geometry

Two-view geometry is the basic geometry that constrains image correspondences between two images. The term "two view" in this thesis means that two images of a scene are taken by a stereo rig, (i.e., a two-camera system), or by a rigid motion of a camera. Hence there are two camera projection matrices  $P_1$  and  $P_2$  associated with these two views.

This chapter is organized as follows: In Section 3.1, the pinhole camera model is briefly introduced. In Section 3.2, epipolar geometry, (i.e., two-view geometry) is described. A special matrix called the fundamental matrix  $F$  is then introduced to explain the geometry constraint between the two views. Section 3.3 deals with the issue of reconstruction for a given  $F$ . In Section 3.4, we briefly review methods for computing the fundamental matrix. In Section 3.5, we focus on the stratification of the 3D geometry to study different kinds of reconstructions that are achievable in different strata.

### 3.1 Camera model

In this section, the perspective projection model (also called the pinhole camera model) is presented. Basic concepts associated with this model, such as camera center, principal axis and intrinsic parameter matrix, are described in detail. We then discuss the issue of radial distortion and learn how to correct it.

#### 3.1.1 Perspective projection camera model

In computer vision context, the most widely used camera model is the perspective projection model. This model assumes that all rays coming from a scene pass through one unique point of the camera, namely, the camera center  $C$ . The camera's focal length  $f$  is then defined by the distance between  $C$  and the image plane. Figure 3.1

shows one example of such a camera model. In this model, the origin of the camera coordinate system  $CXYZ$  is placed at  $C$ . The  $Z$  axis, perpendicular to the image plane  $R$  and passing through  $C$ , is called the principal axis. The plane passing through  $C$  and parallel to  $R$  is the principal plane. The image coordinate system  $xyz$  is on the image plane  $R$ . The intersection of the principal axis with the image plane is accordingly called the principal point  $c$ . The origin of the image coordinate system here is placed at  $c$ .

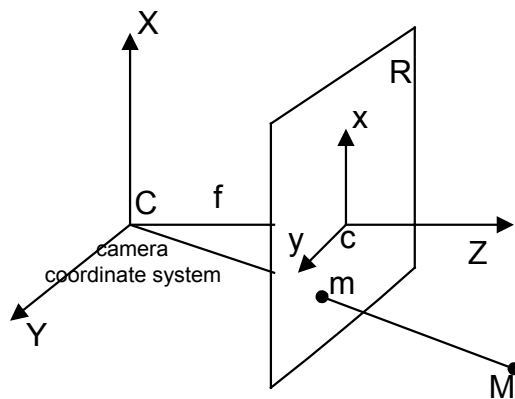


Figure 3.1: The pinhole camera model

At first, we assume that the world coordinate system is the same as the camera coordinate system. Following the simple geometry pictured in Figure 3.1, we have

$$\frac{x}{X} = \frac{y}{Y} = \frac{f}{Z} \quad (3.1.1)$$

Applying homogeneous representation, a linear projection equation can be obtained

$$\tilde{m} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = s \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \sim \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = [\mathbf{I} \ \mathbf{0}] \tilde{M}, \quad (3.1.2)$$

where  $\mathbf{I}$  is the  $3 \times 3$  identity matrix and  $\mathbf{0}$  is a null 3-vector. This is the canonical representation of a perspective projection model. Here,  $\tilde{m}$  and  $\tilde{M}$  represent the homoge-

neous coordinates of the image point  $m$  and the world point  $M$ , respectively. The symbol  $\sim$  means the equation is satisfied up to an unknown scale factor  $s$ .

### 3.1.2 Intrinsic parameters

In many cases, however, the origin of the image coordinate system is not at the principal point. Furthermore, in practice, pixels may not be exact squares and the horizontal axis may not form exact right angle with the vertical axis. To recount for such non-ideal situations, we rewrite the equation (3.1.2) as

$$\tilde{m} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \sim \begin{bmatrix} \alpha f & \beta & u_0 \\ 0 & f & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = A[\mathbf{I} \ \mathbf{0}] \tilde{M}, \quad (3.1.3)$$

with aspect ratio  $\alpha$  the relative scale in image vertical and horizontal axis, skew factor  $\beta$  the skewness of the two axes,  $f$  the focal length and  $(u_0, v_0)$  the principal point. These five parameters are independent of the camera's orientation and position. Hence they are called the intrinsic parameters of the camera and then  $A$  is called the intrinsic parameter matrix.

### 3.1.3 Extrinsic parameters

If the position and orientation of the world coordinate system is different from that of camera coordinate system, then the two coordinate systems are related by a rotation and a translation. Consider Figure 3.2, which illustrates that the rotation  $R$  and the translation  $\mathbf{t}$  bring the world coordinate system to the camera coordinate system, we have

$$\tilde{m} \sim A[R \ | \ \mathbf{t}] \tilde{M}, \quad (3.1.4)$$

where  $R$  and  $t$  represent the camera's orientation and position, respectively, and they are the so-called extrinsic parameters of the camera.

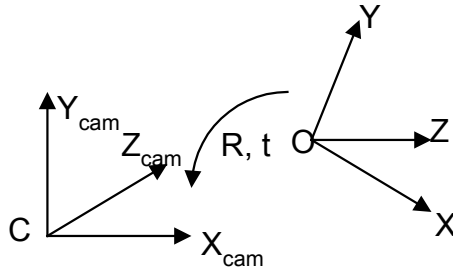


Figure 3.2: The Euclidean transformation between the world coordinate system and the camera coordinate system

The intrinsic parameter matrix and the extrinsic parameter matrix can be combined to produce the so-called the projection matrix (or camera matrix)  $P$ , i.e.,  $P = A[R | t]$ .

Therefore,

$$\tilde{m} \sim P\tilde{M}. \quad (3.1.5)$$

### 3.1.4 Radial distortion

The perspective projection model is a distortion-free camera model. Due to design and assembly imperfections, the perspective projection model does not always hold true and in reality must be replaced by a model that includes geometrical distortion.

Geometrical distortion mainly consists of three types of distortion: radial distortion, decentering distortion, and thin prism distortion [42]. Among them, radial distortion is the most significant and is considered here.

Radial distortion causes inward or outward displacement of image points from their true positions [42]. An important property of radial distortion is its strict symmetry about the principal axis. Thus the principal point is the center of radial distortion. Based on this important property, we can then easily get the form for the expression that measures the size of radial distortion.

$$\delta_{\rho^r} = k_1 \rho^3 + k_2 \rho^5 + k_3 \rho^7 + \dots, \quad (3.1.6)$$

where,  $\delta_{\rho^r}$  measures the deviation of an observed point from an ideal position,  $\rho$  is the distance between a distorted point and the principal point, and  $k_1$ ,  $k_2$  and  $k_3$  are the coefficients of radial distortion. In Cartesian coordinates, equation (3.1.6) becomes

$$\delta_{ur} = k_1 u(u^2 + v^2) + k_2 u(u^2 + v^2)^2 + O[(u, v)^5], \quad (3.1.7)$$

$$\delta_{vr} = k_1 v(u^2 + v^2) + k_2 v(u^2 + v^2)^2 + O[(u, v)^5], \quad (3.1.8)$$

where  $\delta_{ur}$  and  $\delta_{vr}$  are horizontal and vertical components of  $\delta_{\rho^r}$ , and  $u$ ,  $v$  are projections of  $\rho$  in the horizontal and vertical axes. The location of a distorted image point is then given by

$$u' = u + \delta_{ur}(u, v), \quad (3.1.9)$$

$$v' = v + \delta_{vr}(u, v). \quad (3.1.10)$$

## 3.2 Epipolar geometry and the fundamental matrix

Epipolar geometry is the internal geometry that constrains two views. It is independent of scene structure and only depends on the camera's internal parameters and relative pose.

### 3.2.1 Epipolar geometry

Consider the two-camera system in Figure 3.3.  $C$  and  $C'$  are the camera centers. The projections of the two camera centers on the left and right image planes  $e$  and  $e'$  are formally called *epipoles*. A 3D world point  $X$  then defines a plane with  $C$  and  $C'$ . Naturally, its two projections  $x$  and  $x'$  on two image planes are also on this plane. We call this plane the *epipolar plane*. In other words, one projection  $x$  of the world point  $X$  forms the epipolar plane with the baseline  $CC'$ . This plane intersects the other optical ray of  $C'X$  at  $x'$  and the other image plane at an *epipolar line*  $l'$ . Of

course,  $l'$  passes through the epipole  $e'$ . Such geometry discloses the following important facts:

1. Epipolar geometry tells us that, instead of searching for an image point's correspondence on a two-dimensional plane, we only need to look for it along a so-called epipolar line and hence one degree of freedom is eliminated.
2. All epipolar lines intersect at the common point--- epipole.
3. It is possible to recover a 3D world point, because this 3D point and one pair of correspondences form a triangulation, with the 3D point being the intersection of two optical rays. However there is no way to recover any point on the baseline since the epipolar plane is degenerate into a line then.

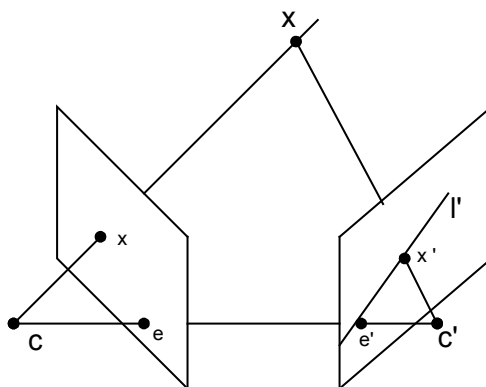


Figure 3.3: Epipolar geometry

### 3.2.2 The fundamental matrix

In Figure 3.3, the epipolar line  $l'$  can be expressed as  $l' = e' \times x' = [e']_x x'$ , where  $\times$  is the cross product, and  $[e']_x$  is the skew symmetric matrix of vector  $e'$ . From equation (3.1.5), we have  $x' = P'X$  and  $x = PX$ . The optical ray back-projected from  $x$  by

$P$  is then given by solving the equation  $x = PX$ . Then we have  $X = \lambda P^+ x + C$ <sup>1</sup>, where  $P^+$  is the pseudo-inverse of  $P$ ,  $C$  is the camera center and  $\lambda$  is a scalar. Following the last section's epipolar geometry, we find  $x'$ , the image correspondence of  $x$ , is on  $x'$ 's correspondent epipolar line  $l'$ . Therefore

$$0 = l'^T x' = l'^T P'X = l'^T P'(\lambda P^+ x + C) = \lambda ([e']_x x')^T P'P^+ x + ([e']_x x')^T P'C. \quad (3.2.1)$$

Since  $P'C = e'$ , the second part of the right side of (3.2.1) is zero. We then have

$$([e']_x x')^T P'P^+ x = x'^T [e']_x P'P^+ x = 0. \quad (3.2.2)$$

We define  $F = [e']_x P'P^+$  as the fundamental matrix. Then the equation (3.2.2) becomes

$$x'^T Fx = 0. \quad (3.2.3)$$

Suppose that the two camera matrices of a stereo rig are

$$P = A[\mathbf{I} \mid \mathbf{0}] \quad P' = A[\mathbf{R} \mid \mathbf{t}]. \quad (3.2.4)$$

Then

$$P^+ = A^{-1} \begin{bmatrix} \mathbf{I} \\ \mathbf{0} \end{bmatrix} \quad C = \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix}. \quad (3.2.5)$$

Hence

$$F = [P'C]_x P'P^+ = [A\mathbf{t}]_x ARA^{-1} = A^{-T} [\mathbf{t}]_x RA^{-1}. \quad (3.2.6)$$

Equation (3.2.6) is the explicit form of the fundamental matrix in terms of camera motion.

Note that, from equation (3.2.6), the rank of the fundamental matrix is two, since  $\text{rank}([\mathbf{t}]_x) = 2$  and both  $A$  and  $R$  are full rank.

---

<sup>1</sup>  $C$  is the null space of  $P$  and one solution of the equation  $x = PX$  is  $P^+x$ . Hence the solution is  $X = \lambda C + P^+x$ . Since  $X$  is determined up to a scale factor, the solution can be expressed as  $X = \lambda P^+x + C$  too.

### 3.3 Recovery of camera matrix from the fundamental matrix

The results from the last section tell us that if a pair of camera matrices  $P$  and  $P'$  are known, the fundamental matrix  $F$  can then be uniquely determined up to an unknown scale factor. However, the converse is not true. That is, if a fundamental matrix is given, two camera matrices cannot be fully recovered, but can still be recovered up to an unknown  $4 \times 4$  projective transformation. This is called the projective ambiguity of cameras given  $F$ .

In order to prove the above assertion, we introduce a simple form of a stereo rig.

#### 3.3.1 Canonical form of camera matrices of a stereo rig

Consider two camera matrices of a stereo rig  $P$  and  $P'$ . If  $H$  is a nonsingular  $4 \times 4$  projective transformation matrix<sup>1</sup>, then the two pairs of camera matrices, namely  $(P, P')$  and  $(PH, P'H)$ , determine the same fundamental matrix. This result is obvious since  $PX = (PH)(H^{-1}X)$  and  $P'X = (P'H)(H^{-1}X)$ . That means a world point  $H^{-1}X$  projected through two camera matrices  $(PH, P'H)$  has the same projections as  $X$  through  $(P, P')$ . As a result, these two pairs of camera matrices have the same fundamental matrix.

We can now assume that two camera matrices of a general stereo rig are in canonical form, i.e.,  $P = [I \mid \mathbf{0}]$  and  $P' = [M \mid \mathbf{m}]$ , where  $I$  is the  $3 \times 3$  identity matrix,  $\mathbf{0}$  is a null 3-vector,  $M$  is a  $3 \times 3$  matrix and  $\mathbf{m}$  is a 3-vector. In other words, we just place the world coordinate system at the position that has the unitary distance with the image plane. Three axes of the world coordinate system are of course parallel to those of the camera coordinate system.

---

<sup>1</sup> A  $4 \times 4$  projective transformation matrix is a  $4 \times 4$  matrix in projective 3D geometry.



### 3.3.2 Camera matrices obtained from $F$

If the camera matrices  $P$  and  $P'$  of a stereo rig are in strictly canonical form, then they can be expressed as  $P = [\mathbf{I} \mid \mathbf{0}]$  and  $P' = [SF \mid e']$ [14], where  $S$  is any skew-symmetric matrix. Luong [20] suggests it is suitable to choose  $S = [e']_{\times}$ . We will omit the proof and just verify the result here. Specifically, let three rows of  $F$  are  $f_1^T$ ,  $f_2^T$  and  $f_3^T$ , we have

$$F = [e']_{\times} P' P^+ = [e']_{\times} [e']_{\times} F = e' \times e' \times \begin{pmatrix} f_1^T \\ f_2^T \\ f_3^T \end{pmatrix} = F, \quad (3.3.1)$$

since  $e'^T F = 0$ <sup>1</sup>.

**Result 3.1.** The canonical camera matrices obtained from a fundamental matrix  $F$  are

$$P = [\mathbf{I} \mid \mathbf{0}] \quad P' = \left[ [e']_{\times} F + e' v^T \mid \lambda e' \right] \quad (3.3.2)$$

where  $v$  is any 3-vector and  $\lambda$  a non-zero scalar.

The above conclusion results from the fact that two projection matrices have in total 22 degrees of freedom. However, a fundamental matrix can only eliminate 7 degrees of freedom. Therefore, 15 degrees of freedom remain and they exactly correspond to the degrees of freedom of a  $4 \times 4$  projective transformation.

## 3.4 The fundamental matrix computation

The fundamental matrix represents a basic constraint on two-view geometry, and thus plays an important role in structure recovery from two or more views. Intense research has been done to accurately estimate the fundamental matrix in the presence of

---

<sup>1</sup> Since the  $e'$  is the null space of  $F^T$ , then  $e'^T \cdot f_1^T = 0$ . Therefore  $e' \times e' \times f_1^T = f_1^T$  follows. In the same principle, it also holds for  $f_2^T$  and  $f_3^T$ .

image noise. This section just briefly reviews some approaches for fundamental matrix computation. Some more intensive treatment of this subject can be found in [43] and [36].

Assume that  $x_i = [u_i, v_i]^T$  and  $x'_i = [u'_i, v'_i]^T$  are one pair of corresponding points in two views. The epipolar geometry indicates that, in general, there is a fundamental matrix  $F$  such that  $x_i'^T F x_i = 0$ . In fact we can rewrite this equation as a homogeneous equation

$$\mathbf{u}_i^T \mathbf{f} = 0, \quad (3.4.1)$$

where

$$\mathbf{u}_i = [u_i u'_i \quad v_i u'_i \quad u'_i \quad u_i v'_i \quad v_i v'_i \quad v'_i \quad u_i \quad v_i \quad 1]^T$$

$$\mathbf{f} = [F_{11} \quad F_{12} \quad F_{13} \quad F_{21} \quad F_{22} \quad F_{23} \quad F_{31} \quad F_{32} \quad F_{33}]^T.$$

Consider  $n$  corresponding points. Let  $U = [u_1^T \quad u_2^T \quad \dots \quad u_i^T \quad \dots \quad u_n^T]^T$ . Then

$$U\mathbf{f} = 0 \quad (3.4.2)$$

### 3.4.1 Linear approaches for $F$ computation

Since the determinant of  $F$  is zero, a fundamental matrix  $F$  has only seven degrees of freedom. Therefore, the minimum number of points needed to compute  $F$  is seven. If we apply equation (3.4.2) over 7 points, then the rank of  $U$  is seven. Hence the dimension of  $\mathbf{f}$  is two. Assume two homogeneous solutions of (3.4.2) are  $\mathbf{f}_1$  and  $\mathbf{f}_2$ . The fundamental matrix is then the linear combination of these two solutions. We then constrain the zero-determinant in the prospective fundamental matrix and then obtain a cubic equation. Therefore, there are three solutions for  $F$ . The disadvantage of this approach is that there is no way to find which one is the exact solution if only seven points are given.

An alternative is to try to use a larger data set. Eight or more points are employed to solve (3.4.2). Usually, they are called 8-point algorithm altogether. Because of the presence of noise in practice, the rank of  $U$  may be greater than seven. There are many approaches to solve such an over-constrained linear system. One popular way is to impose a constraint on the norm of solution vector. Usually, the norm can be set to one. Hence the solution is the unitary eigenvector of  $U^T U$  associated with its smallest eigenvalue.

However, the above linear approach gives poor performance in the presence of noise. Two reasons are responsible for this problem. The first is that zero-rank of  $F$  is not imposed during the estimation. The other is that the objective of linear approach is to solve  $\min_f \|U\mathbf{f}\|^2$  under some constraint. However,  $U\mathbf{f}$  only has algebraic (not geometrical) meaning. Let us consider one row of  $U$ , namely,  $\mathbf{u}_i^T$ . The geometrical distance from the vector  $\mathbf{f}$  to the hyperplane determined by  $\mathbf{u}_i$  is  $\frac{\mathbf{u}_i^T \mathbf{f}}{\|\mathbf{u}_i\|}$ . Therefore, it is more reasonable to minimize such a geometrical distance rather than an algebraic distance  $\mathbf{u}_i^T \mathbf{f}$ .

In linear context, one possible modification of minimization of algebraic distances is to normalize the input data prior to performing the 8-point algorithm. Based on this scheme, Hartley put forward an isotropic scaling of the input data [12]:

1. First, the points are translated so that their centroid is at the origin.
2. The points are then scaled isotropically so that the average distance from the origin to all of points is equal to  $\sqrt{2}$ .

Zhang [43] showed that the normalized 8-point algorithm gives comparable performance with some robust techniques to be described in the next section. Moreover,

this algorithm is quick and easy to implement. Hence, in some cases, which are not so critical about the accuracy of the fundamental matrix, this normalized 8-point algorithm is reliable.

### 3.4.2 Nonlinear approaches for $F$ computation

Three nonlinear minimization criteria are discussed here. The first one is to minimize distances of the image points to the epipolar lines. Specifically, consider one observed pair of stereo corresponding points  $(x_i, x'_i)$  and an initial estimation of the fundamental matrix  $F$ . Since the image points are corrupted by noise to a certain extent,  $x_i, x'_i$  and  $F$  do not exactly satisfy the epipolar geometry  $x'^T F x = 0$ . Then the first criterion is interpreted as follows

$$\min_F \sum_i (d^2(x'_i, Fx_i) + d^2(x_i, F^T x'_i)). \quad (3.4.3)$$

From the last section, we find algebraic distance differs from geometrical distance by some scale. Such scale changes with different image correspondences. The second criterion attempts to rescale algebraic distance by different weights. Assume a variety<sup>1</sup>  $\nu_F = x'^T F x$ , the criterion is

$$\min_F \sum_i \frac{(\nu_F)^2}{\sigma(\nu_F)^2}, \quad (3.4.4)$$

where  $\sigma(\nu_F)^2$  is the variant of  $\nu_F$ . If we assume the image points are corrupted by independent Gaussian noise, then the image points' covariant matrices are given by

$$\Lambda_{x_i} = \Lambda_{x'_i} = \sigma^2 \text{diag}(1, 1), \quad (3.4.5)$$

here  $\sigma$  is the noise level. According to the first order or Sampson approximation [14] [26], we have the variance of  $\nu_F$

---

<sup>1</sup> A variety is the simultaneous zero-set of one or more multivariate polynomials defined in  $\mathbb{R}^n$

$$\sigma^2(\mathbf{v}_F) = \frac{\partial \mathbf{v}_F}{\partial x_i}{}^T \Lambda_{xi} \frac{\partial \mathbf{v}_V}{\partial x_i} + \frac{\partial \mathbf{v}_F}{\partial x'_i}{}^T \Lambda_{xi} \frac{\partial \mathbf{v}_V}{\partial x'_i} = \sigma^2(l_1^2 + l_2^2 + l_1'^2 + l_2'^2). \quad (3.4.6)$$

Here  $l_1$ ,  $l_2$ , and  $l_1'$ ,  $l_2'$  are first two elements of  $F^T x'_i$  and  $Fx_i$  respectively.

Since a constant number does not affect the minimization, the second criterion becomes

$$\min_V \sum_i (x_i'^T Fx_i)^2 / (l_1^2 + l_2^2 + l_1'^2 + l_2'^2). \quad (3.4.7)$$

The last criterion minimizes the distances between observed image points and re-projected image points. In Section 3.3, we know that camera projection matrices of a stereo rig can be recovered up to an unknown  $4 \times 4$  projective transformation. Based on the recovered camera projective matrices, the so-called projective reconstruction can be done at this stage. Here, we don't discuss this aspect of techniques. A thorough discussion can be found in [10]. From the back-projected 3D points, we re-project them into the image planes. If we assume that  $\hat{x}_i$  and  $\hat{x}'_i$  are re-projections, then the third criterion is

$$\min_F \sum_i (d^2(x_i, \hat{x}_i) + d^2(x'_i, \hat{x}'_i)). \quad (3.4.8)$$

Some researchers [43] [36] point out that the first criterion is slightly inferior to the last two. However, the computation cost for the last one is highest because it involves two minimization procedures: the first is the minimization in projective reconstruction and the second is the minimization in calculating an optimal fundamental matrix. Therefore, the criterion (3.4.7) is usually recommended.

### 3.4.3 Robust estimation of the fundamental matrix

Up to now, we assume image correspondences are obtained without poor matches. However, due to limited performance of feature detectors and match algorithms, poor

matches (or outliers) are often present during the computation of the fundamental matrix. There are two reasons for this. One is the bad localization of an image point. The other is false match. Usually, an image point deviating from its expected location by more than 3 pixels can be considered as a poor localization. False match means that a detected match is not the correct match.

M-estimators [43] is robust to outliers resulting from poor localization. All estimators we used in the last section rely on least-squares approach. It means a poor localization (and hence a large residual) contributes more to an estimator. Consider the estimator  $estimator = \min_F \sum_i \rho(r_i) = \min_F \sum_i r_i^2$ . The influence of a datum linearly in-

creases with the size of its residual, since  $\frac{\partial \rho(r_i)}{r_i} = r_i$ . As a consequence, the scheme of

M-estimator tries to find a symmetric, positive-definite function with a unique minimum at zero. One choice of such a form of function is the following Tukey function:

$$\rho(r_i) = \begin{cases} \frac{c^2}{6} (1 - [1 - (\frac{r_i}{c\sigma})^2]^3) & \text{if } |r_i| \leq c\sigma \\ (c^2 / 6) & \text{otherwise} \end{cases}, \quad (3.4.9)$$

where  $c = 4.6851$  and  $\sigma$  is given by

$$\sigma = 1.4826[1 + 5/(n - p)] \text{median}_i |r_i|, \quad (3.4.10)$$

where  $n$  is the size of the data set and  $p$  is the dimension of parameters.

From (3.4.9), we know the influence of poor matches (when residuals are greater than  $c\sigma$ ) is refrained by setting residuals to constants. Because of this, M-estimator works well with poor matches resulted from bad localization. However, it does not demonstrate good performance when outliers result from false matches, because it depends heavily on the initial estimation [43].

Least Median of Squares (LMedS), however, overcomes the disadvantage of M-estimator. Its estimator

$$\min_i \text{median } r_i^2 \quad (3.4.11)$$

tries to minimize the median of squared residual for an entire data set.

LMedS is based on the Monte Carlo techniques and thus is difficult to use mathematical formulas to describe it. Usually it first randomly selects  $m$  subsamples of the entire data set. For each subsample, one of the linear approaches described in Section 3.4.1 is employed to provide the initial estimation of the fundamental matrix. One of three criteria (see Section 3.4.2) is then applied to obtain the median of the squared residuals. After repeating the above procedures over all subsamples, the optimal estimation of  $F$  is the one that makes residuals the minimal among all subsamples.

The number of subsamples  $m$  is usually determined by

$$m = \frac{\log(1-P)}{\log[1-(1-\varepsilon)^p]}, \quad (3.4.12)$$

where  $P$  is the probability that at least one sub-sample is good (not seriously polluted by outliers) and  $\varepsilon$  is the proportion of outliers to the entire data set [43].

Since LMedS does not work well in the presence of Gaussian noise [25], Zhang [43] proposed a weighted LMedS procedure, which specifies that when the residual is greater than 2.5 times a *robust standard deviation*  $\hat{\sigma}$ , the correspondent weight for the residual is 0. That means this datum is then discarded. Here  $\hat{\sigma}$  is given by

$$\hat{\sigma} = 1.4826[1 + 5/(n-p)]\sqrt{M_J}, \quad (3.4.13)$$

where  $n$  is the number of the data set,  $p$  is the dimension of the parameters to be estimated and  $\sqrt{M_J}$  is the root of the least median of the squared residual.

It is noted that this weighted LMedS procedure is conducted after the normal LMedS.

## 3.5 The stratification of the 3D geometry

Euclidean space is by far the most familiar space to human perception. However, when our perception moves from 2D (images) to 3D (world), depth is lost. Without some control points in the Euclidean space, there is no way to fully recover the Euclidean structure [5]. However, in many applications, it may not be essential that absolute geometry (i.e., the exact dimension and structure) of the world should be recovered. In fact, we might find it sufficient to have simpler reconstructions (compared with Euclidean reconstruction) of the world on some layers of the 3D geometry. The process in which we identify different layers of the 3D geometry is the so-called stratification of the 3D geometry. Usually, three-dimensional geometry is stratified to four different structures residing in separate layers in the 3D geometry. When arranged in order of complexity and degree of realism, these structures can be listed as: projective structure, affine structure, metric structure, and Euclidean structure.

### 3.5.1 The 3D projective structure

In Section 3.3, we know that, given a fundamental matrix (that means two views), camera matrices of a stereo rig can be recovered up to an unknown  $4 \times 4$  projective transformation  $H$ . The structure recovered from such two camera matrices is then called the 3D projective structure. It is the simplest structure obtained from images.

### 3.5.2 The 3D affine structure

We know that an affine transformation does not change the plane at infinity  $\Pi_\infty$ , as discussed in Section 2.3.3. If  $\Pi_\infty$  can be identified in the projective space, then the 3D projective structure can be upgraded to the affine structure. This structure is closer to the world since parallelism is invariant in the affine space.



One way to identify  $\Pi_\infty$  is to use parallel lines. We know that parallel lines intersect at the infinity. If three *vanishing points*<sup>1</sup> are available, then it is sufficient to construct  $\Pi_\infty$ . The identified  $\Pi_\infty$  can be used to construct a projective transformation  $H$ , which transforms  $\Pi_\infty$  into its canonical form  $[0 \ 0 \ 0 \ 1]^T$ . This  $H$  when acting on other points, can produce an affine structure of the scene.

### 3.5.3 The 3D metric structure

In the 3D metric structure, not only parallelism but also angles and ratios of lengths are preserved. Hence the structure is very similar to the true one — only the dimension of the scene is missing. Concretely, the scene is recovered up to an unknown similarity transformation. A  $4 \times 4$  similarity transformation is a scaled Euclidean transformation,

i.e.,  $\begin{bmatrix} s\mathbf{R} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix}$ , where  $\mathbf{R}$  and  $\mathbf{t}$  are the rotation matrix and translation vector.

The key to metric reconstruction is the identification of the absolute conic. (The absolute conic will be explained in detail in the next chapter.) Since the image of the absolute conic (IAC) is invariant to camera motions, the metric reconstruction can be obtained from an affine reconstruction if enough (at least three) images are given.

One way to identify the IAC is to use vanishing points. Consider three vanishing points  $\mathbf{v}_1$ ,  $\mathbf{v}_2$  and  $\mathbf{v}_3$ , arising from three pairs of mutually orthogonal scene lines.

These three points then give rise to three constraints on IAC  $\omega$ :

$$\mathbf{v}_1^T \omega \mathbf{v}_2 = 0 \quad (3.5.1)$$

$$\mathbf{v}_1^T \omega \mathbf{v}_3 = 0 \quad (3.5.2)$$

$$\mathbf{v}_2^T \omega \mathbf{v}_3 = 0 \quad (3.5.3)$$

---

<sup>1</sup> A vanishing point is the intersection of images of parallel lines.

The above three constraints, together with two other constraints introduced by two circular points in the IAC [18], can be used to solve for five unknown parameters in the IAC.

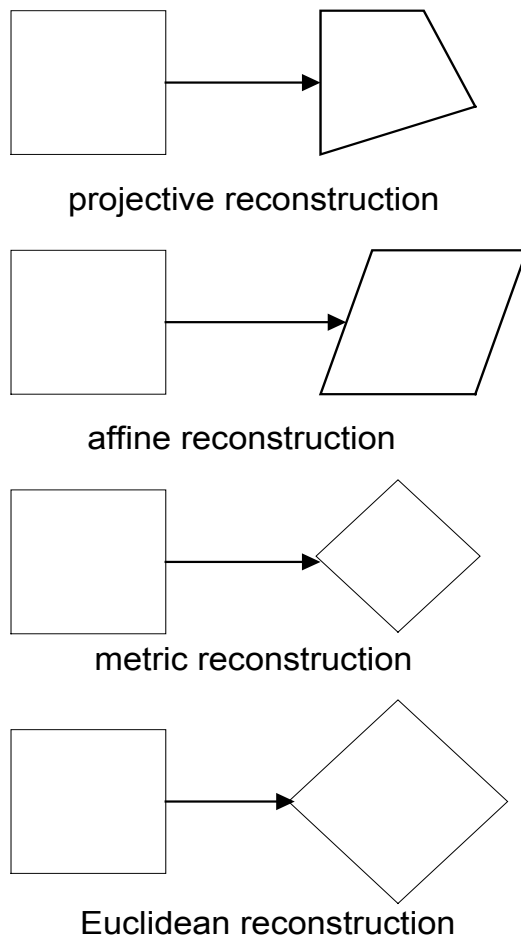


Figure 3.4: Different structures recovered on different layers of 3D geometry

Figure 3.4 shows different reconstructions on different layers of 3D geometry. The left column contains the original objects and the right column contains the reconstructed objects. The first row shows projective reconstruction. The reconstructed object appears to have no resemblance with the original object. Indeed, there is an implicit invariance called cross-ratio beneath the appearance. The second row shows affine reconstruction. The most significant phenomenon is that parallelism is preserved. The third row shows metric reconstruction. In this case both angles and parallelism are

preserved. However, the final dimension is unknown. The last row demonstrates Euclidean reconstruction. In this case, the object is fully recovered.

### 3.5.4 Camera self-calibration, the bond between projective reconstruction and metric reconstruction

Camera self-calibration means automatically calibrate camera's parameters<sup>1</sup> without any 3D information. From the knowledge of Section 3.5.1, 3.5.2 and 3.5.3, we know metric structure can be achieved when affine reconstruction is completed and IAC is identified. However, with the knowledge of camera's intrinsic parameters, metric structure can be directly obtained from projective structure. Specifically, when the intrinsic parameter matrix  $A$  is known, the fundamental matrix can be reduced to the so-called *essential matrix*  $E$ . In fact the relation between  $F$  and  $E$  is

$$E = A^T F A \quad (3.5.4)$$

As a result,  $\text{rank}(E) = \text{rank}(F) = 2$ . The SVD of  $E$  takes the form of  $U \text{diag}(1,1,0) V^T$ , where  $U$  and  $V$  are two orthogonal matrices. Consider the rotation  $R$  and translation  $\mathbf{t}$  between two cameras of a stereo rig. Then we have the following result [9]:

**Result 3.2** Suppose that the SVD of a given essential matrix is  $E = U \text{diag}(1,1,0) V^T$ , and the first camera matrix is  $P = [\mathbf{I} \mid \mathbf{0}]$ . Then there are four possible choices for the second camera matrix  $P'$ :

$$[U W V^T \mid \mathbf{u}_3], \text{ or } [U W V^T \mid -\mathbf{u}_3], \text{ or } [U W^T V^T \mid \mathbf{u}_3], \text{ or } [U W^T V^T \mid -\mathbf{u}_3],$$

where  $W = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ , and  $\mathbf{u}_3$  is the third column of  $U$ .

---

<sup>1</sup> Usually, only intrinsic parameters are calibrated. However, in some special cases (such as stationary cameras), it is possible to obtain extrinsic parameters.

In the four possible forms of  $P'$  above, only one can make the reconstructed points to be in front of both the cameras. Thus with a single point, the correct camera matrix could be found.

The above solution of camera matrix leaves one ambiguity: the scale of translation. Other than that, the metric reconstruction is completed. In turn, this means if  $A$  is automatically calibrated, it is possible to directly upgrade from projective structure to metric structure.

## Chapter 4. Camera self-calibration

Camera self-calibration means camera's parameters can be calibrated without any 3D information. In other words, a camera is calibrated from images alone. Traditional calibration methods need a calibration pattern. The orientation and position with respect to the camera are known for a calibration pattern. These approaches all lead to linear and nonlinear least squares problems, and these solutions can be obtained with high precision [41]. However, such a pattern is not cheap to manufacture (The calibration object used in this thesis costs more than \$3000). Furthermore, in many applications, it is not flexible and even impossible to place a calibration object before a camera. Camera self-calibration is then put forward to solve such problems.

In this chapter, we first discuss Kruppa's equations based camera self-calibration in Section 4.1. Afterwards, some well-known self-calibration algorithms are reviewed. At last, our focal length calibration algorithm is then directly obtained from the simplified Kruppa's equations.

### 4.1 Kruppa's equations based camera self-calibration

Kruppa's equations were firstly discovered by Kruppa in 1913 [16]. However, they were not well known until Maybank and Faugeras introduced them into the field of computer vision for camera self-calibration. Geometrically, "Kruppa's equations impose that the epipolar lines, which correspond to the epipolar planes tangent to the absolute conic, should be tangent to its projection in both images" [50]. We will show this in the next sections.

#### 4.1.1 Absolute conic and image of the absolute conic

The first camera self-calibration algorithm [6] is based on Kruppa's equations. In fact, camera self-calibration is equivalent to recovering the image of a distinguished conic<sup>1</sup> in the plane at infinity  $\Pi_\infty$ . Such a distinguished conic is the so-called absolute conic. Its definition is

$$x^2 + y^2 + z^2 = 0 . \quad (4.1.1)$$

All solutions of equation (4.1.1) are imaginary, however, its properties are a little different from any other conic.

The most important property of the absolute conic is that it maps onto itself when it undergoes the scaled Euclidean transformation<sup>2</sup>.

**Theorem 1:** The absolute conic (AC)  $\Omega$  is mapped onto itself under the scaled Euclidean transformation

*Proof:* Assume that a point  $[x \ y \ z \ 0]^T$  is on the absolute conic. By applying the scaled Euclidean transformation, the transformed point also falls onto  $\Pi_\infty$ . Its first three coordinates  $x'$ ,  $y'$  and  $z'$  are determined by

$$\begin{aligned} x' &= sr_{11}x + sr_{12}y + sr_{13}z \\ y' &= sr_{21}x + sr_{22}y + sr_{23}z . \\ z' &= sr_{31}x + sr_{32}y + sr_{33}z \end{aligned} \quad (4.1.2)$$

In the above equations,  $s$  is the unknown scale,  $r_{ij}$  (with  $i = 1, 2, 3$  and  $j = 1, 2, 3$ ) represent the 9 elements of a rotation matrix. Thus we have

$$x'^2 + y'^2 + z'^2 = s^2(x^2 + y^2 + z^2) = 0$$

The image of the absolute conic (IAC) is the projection of the absolute conic onto the image plane. It is totally determined by the intrinsic parameter matrix  $A$  of a camera. In fact, we have the following property of IAC.

---

<sup>1</sup> A conic is a quadratic in a plane

<sup>2</sup> Scaled Euclidean transformation is also called the similarity transformation

**Theorem 2:** The image of the absolute conic is determined by the intrinsic parameter matrix, and is invariant under rigid displacement of the camera, provided that the camera's intrinsic parameter matrix remains unchanged.

*Proof:* Consider a point  $p = [x \ y \ z \ 0]^T$  in the camera coordinate system. The projection  $[u \ v]^T$  of such a point is given by

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = A \begin{bmatrix} x \\ y \\ z \end{bmatrix}.$$

Thus

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = sA^{-1} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \quad (4.1.3)$$

Substituting (4.1.3) to (4.1.1) yields

$$[u \ v \ 1]A^{-T} \mathbf{I}A^{-1}[u \ v \ 1]^T = 0$$

Thus the coordinates of IAC are determined by  $A^{-T}A^{-1}$ , and totally parameterized by  $A$ . If  $p$  undergoes a rigid displacement (which is equivalent to camera displacement), then its corresponding image is

$$s' \begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = A[R \ \mathbf{t}] \begin{bmatrix} x \\ y \\ z \\ 0 \end{bmatrix} = AR \begin{bmatrix} x \\ y \\ z \end{bmatrix},$$

where  $R$  is the rotation matrix and  $\mathbf{t}$  the translation vector. Therefore the corresponding IAC is

$$A^{-T}R^{-T} \mathbf{I}R^{-1}A^{-1} = A^{-T}A^{-1}.$$

We finally conclude that IAC is invariant under rigid displacement of the camera.

Since IAC is determined by only the intrinsic parameter matrix  $A$ , if we have enough constraining points on the IACs, then  $A$  can be fully recovered from images.

### 4.1.2 Kruppa's equations

Kruppa's equations represent the constraints on camera calibration matrix. Specifically, given two views, Kruppa's equations are in the form of two quadratic equations involving intrinsic parameters. Thus totally three camera motions are needed to determine all five intrinsic parameters.

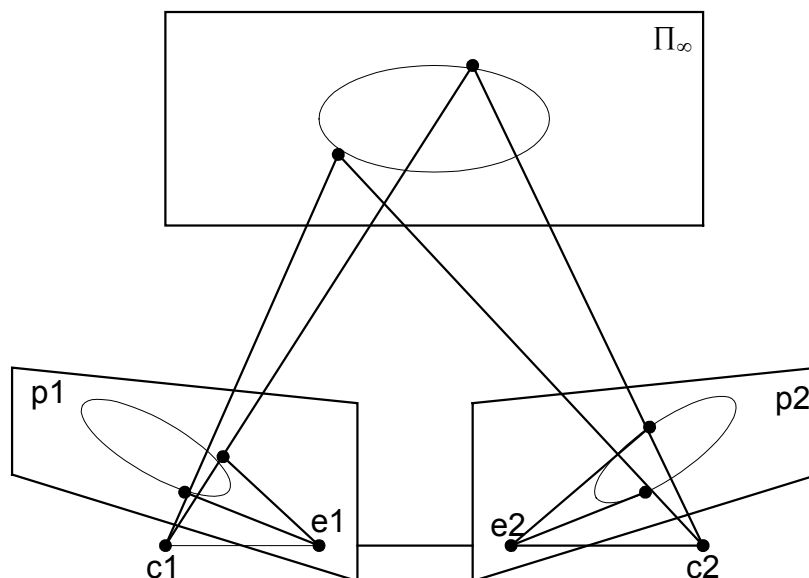


Figure 4.1: Absolute conic and its image

Consider a camera undergoing a rigid displacement. Then the arrangement involving the two location of the camera before and after the motion constitutes a stereo rig. As Figure 4.1 shows,  $c1$  is the first camera center, and  $c2$  is the second camera center. The baseline  $c1c2$  then intersects the two image planes  $p1$  and  $p2$  at  $e1$  and  $e2$ , respectively, which are the epipoles. In Figure 4.1, the conic in  $\Pi_\infty$  is the absolute conic. There are two planes that pass through the baseline and are tangent to the absolute conic. The intersections of these two planes with the two image planes form the correspondent epipolar lines. In Chapter 1, we have known a line  $l$  through two points  $x$  and  $x'$  can be determined by  $l = x \times x'$ . Hence, epipolar lines can be parameterized as the cross product of the epipole and a point at infinity. Specifically, let



$p = [p_1 \ p_2 \ p_3]^T$  be an epipole and  $y = [y_1 \ y_2 \ 0]^T$  be a point at infinity. Then the correspondent epipolar line is  $l = p \times y$ . An equivalent way to treat such a conic is to take it as a dual conic which is enveloped by all of its tangent lines. Here, the parameter matrix of such a dual conic is the inverse of the original parameter matrix. Therefore the parameter matrix of the dual of the image of the absolute conic (DIAC) is  $AA^T$ . Thus,  $l$  is tangent to an IAC if and only if it lies on the dual conic, i.e.,

$$(p \times y)^T AA^T (p \times y) = 0 \quad (4.1.4)$$

Using Kruppa's notation [16], we define

$$D = AA^T = \begin{bmatrix} -\delta_{23} & \delta_3 & \delta_2 \\ \delta_3 & -\delta_{13} & \delta_1 \\ \delta_2 & \delta_1 & -\delta_{12} \end{bmatrix}^1. \quad (4.1.5)$$

Substituting (4.1.5) for (4.1.4),

$$A_{11}y_1^2 + 2A_{12}y_1y_2 + A_{22}y_2^2 = 0, \quad (4.1.6)$$

where

$$A_{11} = -\delta_{13}p_3^2 - \delta_{12}p_2^2 - 2\delta_1p_2p_3$$

$$A_{12} = \delta_{12}p_1p_2 - \delta_3p_3^2 + \delta_2p_2p_3 + \delta_1p_1p_3$$

$$A_{22} = -\delta_{23}p_3^2 - \delta_{12}p_1^2 - 2\delta_2p_1p_3$$

Since IAC is invariant under rigid Euclidean transformation, for the IAC in the second image, (4.1.6) also holds. We just replace  $p_1, p_2, p_3$  with the second epipole's coordinates  $p'_1, p'_2$ , and  $p'_3$  to obtain

$$A'_{11}y_1'^2 + 2A'_{12}y_1'y_2' + A'_{22}y_2'^2 = 0 \quad (4.1.7)$$

---

<sup>1</sup> We just show that  $D$  is a symmetric matrix like this equation demonstrates. In fact, we can easily get what is  $\delta_1, \dots, \delta_{23}$  by computing  $AA^T$ .

Since epipolar lines in the two images in Figure 4.1.1 define the epipolar geometry, there is a  $2 \times 2$  transformation that relates those epipolar lines, i.e.

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} y'_1 \\ y'_2 \end{bmatrix}$$

Let  $\tau = y_2/y_1$  and  $\tau' = y'_2/y'_1$ , equations (4.1.6) and (4.1.7) further reduce to

$$A_{11} + 2A_{12}\tau + A_{22}\tau^2 = 0 \quad (4.1.8)$$

$$A'_{11}(b\tau + c)^2 + 2A'_{12}(b\tau + c)(\tau + a) + A'_{22}(\tau + a)^2 = 0 \quad (4.1.9)$$

The above two equations have the same roots. Therefore the coefficients of  $\tau$  in these two equations differ by a scale  $s$ . Hence we have

$$A_{12}(A'_{22}a^2 + A'_{11}c^2 + 2A'_{12}ac) - (A'_{12}c + A'_{22}a + A'_{11}bc + A'_{12}ab)A_{11} = 0 \quad (4.1.10)$$

$$A_{22}(A'_{22}a^2 + A'_{11}c^2 + 2A'_{12}ac) - (2A'_{12}b + A'_{22} + A'_{11}b^2)A_{11} = 0 \quad (4.1.11)$$

Equations (4.1.10) and (4.1.11) are the so-called Kruppa's equations.

### 4.1.3 Simplified Kruppa's equations

Kruppa's equations given in the last section are not in the explicit form; that is, the intrinsic parameters are implicitly contained in the coefficients. Furthermore, if we follow the above derivation to compute the camera's intrinsic parameters, we will need to compute first the epipoles and then estimate that  $2 \times 2$  transformation matrix. The computation cost is high. The most serious disadvantage is that computation of epipoles is very sensitive to noise. Hence equations (4.1.10) and (4.1.11) usually are not used in practice.

A simplified form of Kruppa's equations based on singular value decomposition of the fundamental matrix was firstly presented in [13]. It states that given one fundamental matrix or two views, there are two equations constraining the camera's intrinsic parameters. All of the parameters are enclosed in a matrix, which can be entirely taken as

an unknown variable. The coefficients with respect to intrinsic parameters are obtained from the SVD of the fundamental matrix.

Let  $F$  be the fundamental matrix constraining the epipolar geometry of a two-camera system. If we transform two image coordinates such that epipoles coincide at the origins and the correspondent epipolar lines have identical coordinates, then the transformed fundamental matrix is in a special form, i.e.,

$$F' = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

If  $T$  and  $T'$  are the corresponding transformations, then the two DIACs are  $TAA^T T^T$  and  $T'AA^T T'^T$  respectively. Since epipoles are at the origin and epipolar lines have identical coordinates, we can parameterize the epipolar lines as  $[\lambda \ \mu \ 0]^T$ . Then equation (4.1.4) can be reformulated as follows:

$$\lambda^2 d_{11} + 2\lambda\mu d_{12} + \mu^2 d_{22} = 0 \quad (4.1.12)$$

$$\lambda^2 d'_{11} + 2\lambda\mu d'_{12} + \mu^2 d'_{22} = 0 \quad (4.1.13)$$

where  $D = \begin{bmatrix} d_{11} & d_{12} & d_{13} \\ d_{21} & d_{22} & d_{23} \\ d_{31} & d_{32} & d_{33} \end{bmatrix} = TAA^T T^T$ , and  $D' = \begin{bmatrix} d'_{11} & d'_{12} & d'_{13} \\ d'_{21} & d'_{22} & d'_{23} \\ d'_{31} & d'_{32} & d'_{33} \end{bmatrix} = T'AA^T T'^T$ .

Since equations (4.1.12) and (4.1.13) have the same roots, their coefficients are identical up to an unknown scale factor, i.e.,

$$\frac{d_{11}}{d'_{11}} = \frac{d_{12}}{d'_{12}} = \frac{d_{22}}{d'_{22}} \quad (4.1.14)$$

This is another form of the Kruppa's equations.

To derive the explicit form of  $d_{11}, d_{12}, d_{22}, d'_{11}, d'_{12}, d'_{22}$ , we first assume that the SVD of  $F$  is  $F = USV^T$ , where  $U$  and  $V$  are orthogonal matrices. If the two nonzero singular values of  $F$  are  $a$  and  $b$ , then the SVD of  $F$  can be expressed as,

$$F = U \begin{bmatrix} a & & \\ & b & \\ & & 1 \end{bmatrix} \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} V^T = U \begin{bmatrix} a & & \\ & b & \\ & & 1 \end{bmatrix} F' \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} V^T. \quad (4.1.15)$$

Let  $T = \begin{bmatrix} a & & \\ & b & \\ & & 1 \end{bmatrix} U^T$ ,  $T' = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} V^T$  and  $C = AA^T$ . Then we have  $d_{11} = a^2 u_1^T C u_1$ ,

$d_{12} = a b u_1^T C u_2$ ,  $d_{22} = b^2 u_2^T C u_2$ ,  $d'_{11} = v_2^T C v_2$ ,  $d'_{12} = -v_2^T C v_1$  and  $d'_{22} = v_1^T C v_1$ . Here

$u_1, u_2, v_1$  and  $v_2$  are first two columns of  $U$  and  $V$ . As a result, Kruppa's equations obtained from the SVD of the fundamental matrix are

$$\frac{a^2 u_1^T C u_1}{v_2^T C v_2} = \frac{a b u_1^T C u_2}{-v_2^T C v_1} = \frac{b^2 u_2^T C u_2}{v_1^T C v_1}. \quad (4.1.16)$$

## 4.2 Review of Camera self-calibration

Since the possibility of camera calibration from images was proved in [6], intense research has been conducted on this topic. Several important algorithms have been proposed. Some of them consider special motions [11][4][21]. Others deal with special objects such as planar scenes [39]. Most of these approaches assume that camera's intrinsic parameters are constant. Recently, some algorithms based on varying parameters were also put forward [24].

### 4.2.1 Self-calibration for stationary cameras

In [11], Hartley gives a linear algorithm when there is no translation between two cameras. Since two cameras cannot construct a stereo rig in pure rotation case, Kruppa's equations are not valid here. However, there is a projective transformation that relates two images obtained from the two cameras. Specifically, consider two camera matrices  $P_1 = A[R_1 \ 0]$ ,  $P_2 = A[R_2 \ 0]$  and a 3D point  $X$ . Then the point in

first image is  $u_1 = AR_1X$  and in second image is  $u_2 = AR_2X$ . Hence we have  $u_1 = AR_1R_2^{-1}A^{-1}u_2$ . Since  $P = AR_1R_2^{-1}A^{-1}$  can be established by image correspondences, the problem is then to find an upper triangular matrix  $A$  that transforms  $P$  into a rotation matrix. Using the property of rotation matrix  $R = R^{-T}$ , we have  $AA^T P^{-T} = PAA^T$ . Such an equation is easily reformulated to the form of  $Xa = 0$ . Therefore this algorithm is linear. After the calibration matrix  $A$  is determined from the above linear equations, 3D points can be recovered and thus we can obtain reprojected image points. Hence an iterative estimation of the calibration matrix  $A$  can be further derived if we try to minimize errors between observed and reprojected image points. However, as Hartley [11] points out, "In the examples used for experimentation it turned out that this (iterative estimation) did not yield very great benefits. The solution for  $A$  given by the non-iterative method was so good that the difference between the estimates found with and without this final estimation step did not differ very significantly". Experiments in [11] show the calibrated focal length are accurate within 8% of the true one.

The disadvantage of Hartley's algorithm is that the camera must be stationary. That means no translation of camera centers is permitted. Since it is difficult to find a camera's center, keeping a camera's center still while rotating it is not easy.

#### 4.2.2 Kruppa's equations based self-calibration for two special motions

From Section 4.1.2, we know that to self-calibrate a camera, at least three images are needed. However Kruppa's equations degenerate in some special cases. Hence there will be situations where three views may not be sufficient to calibrate a camera. In [21], Ma points out that when the rotation axis is parallel or perpendicular to the direction of translation, Kruppa's equations can be rewritten into three linear equations.

However, one of them may depend on the other two. Ma argues that it is possible to find the common scale factor  $\lambda$  generated from Kruppa's equations. Hence Kruppa's equations are in the form of three linear equations. Normally, it is not easy to find such a scale factor, but Ma determines it for the two special motions above. When the rotation axis is parallel to the translation, the square of scale factor is given by  $\lambda^2 = \|F^T \hat{T}' F\|$ , where  $F$  is the fundamental matrix, and  $\hat{T}'$  is the skew symmetric matrix of the normalized translation vector  $T'$ . When the rotation axis is perpendicular to the translation,  $\lambda$  is one of the two non-zero eigenvalues of  $F^T \hat{T}'$ . Since those three linear equations are interdependent, we still need three images to calibrate the camera. However, Ma proposes that cross-multiplying out Kruppa's equations only imposes one constraint on the calibration matrix  $A$  in the perpendicular case. Hence three images are not enough.

Like Hartley's work, Ma's result can only be used in limited scenarios. The more important problem is that in perpendicular case, we cannot know which eigenvalue of  $F^T \hat{T}'$  is correct before we try all solutions. As his simulation shows in this case, two eigenvalues are close to each other and thus it is difficult to determine which one is correct. This accordingly makes calibration difficult.

### 4.2.3 Self-calibration from special objects

Triggs generalized Kruppa's equations by introducing the absolute quadric [38].

Absolute quadric is a  $4 \times 4$  matrix, such as  $\begin{pmatrix} I_{3 \times 3} & 0 \\ 0 & 0 \end{pmatrix}$ . Like the absolute conic, it is also

invariant to the scaled Euclidean transformation. His generalized constraints for camera intrinsic parameters are then called *absolute quadric projection constraints*. Let  $\Omega$

be the absolute quadric and  $\omega$  be the image of the absolute quadric. If  $P$  is the camera matrix which projects  $\Omega$  onto its image  $\omega$ , the constraints are,

$$\omega \times (P_i \Omega P_i^T) = 0 \quad (4.2.1)$$

or in other words,

$$\omega = \lambda_i P_i \Omega P_i^T \quad (4.2.2)$$

where  $\times$  denotes the cross product operation. Eliminating the constants  $\lambda_i$ , we obtain the following equations:

$$\omega_{(jk)} (P_i \Omega P_i^T)_{(j'k')} - \omega_{(j'k')} (P_i \Omega P_i^T)_{(jk)} = 0 \quad (4.2.3)$$

where  $(jk)$  represents the matrix element on row  $j$  and column  $k$ . Geometrically, these constraints mean that an angle formed by two projection planes measured by  $\Omega$  is equal to that contained in image lines measured by  $\omega$ . Kruppa's equations, in this context, are just the projection of these constraints onto the epipolar planes.

Triggs's self-calibration algorithm is based on numerical approaches. Specifically, from the equation, we know the left hand side of the equation is a skew symmetric matrix. Hence for one view, the equation imposes  $9 + 6 = 15$  constraints on unknown variables. In other words, it will generate 15 bilinear equations, where only 5 are linearly independent [38]. However, there are totally 13 (5 for  $\omega$  and 8 for  $\Omega$ ) unknowns in the equation. Thus at least three views are needed to solve the equation. Triggs then employs nonlinear minimization algorithms to optimize the unknown parameters. The object function is equation (4.2.3), and the initial estimation could be  $\omega_0 = \mathbf{I}$  and  $\Omega_0 = \begin{pmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}$ , where  $\mathbf{I}$  is the  $3 \times 3$  identity matrix. The intrinsic parameter matrix  $A$  is then the Choleski decomposition of  $\omega$ .

One special case of Triggs's algorithm is that the object is a planar scene [39]. In this case, one of the images is taken as the planar scene. Hence, this image is related by

homographies<sup>1</sup> with the remaining images. Unlike the equation (4.2.3), homographies rather than projection matrices impose constraints on  $\omega$  (Explicit form of the constraints can be found in [39]). Triggs points out that one view only provides two constraints on  $\omega$ . Hence, five unknowns in  $\omega$  and other four unknown parameters<sup>2</sup> need at least five views.

Homographies computation is simpler and more robust than the fundamental matrix computation. Hence this algorithm generally gives good performance. However, poor initial estimation of  $\Omega$  may make nonlinear estimation fall into local minima.

### 4.3 Focal length self-calibration from two images

In this section, we introduce our calibration algorithm. It is based on the simplified Kruppa's equations (4.1.16). Although the equations give neat constraints on camera's intrinsic parameter matrix, it is not easy to solve for these parameters since there are multiple solutions for these nonlinear equations [13]. Specifically, there are totally five unknown parameters in  $A$ , but one fundamental matrix only gives two constraints. Therefore three fundamental matrices or three images are needed to fully calibrate a camera. However, three images represent six constraints. It is difficult to know whether these six constraints are independent. Even they are actually independent, solutions from any five of the six constraints could be different. Thus there are  $2^5 = 32$  possible solutions. We need to eliminate spurious solutions case by case.

Another problem is that implicit singularities occur when a camera is under some special motions [21, 31]. Singularities generated from both special motions and general motions will be discussed in the next chapter.

---

<sup>1</sup> A homography is the projective transformation in two planes.

<sup>2</sup> Specifically, they are parameters of two circular points [14].



Among the five parameters of  $A$ , focal length is the most important. Actually, in most cases, aspect ratio can be assumed to be 1 and skew factor to be 0. Furthermore, it is safe to assume that the principal point is at the image center. After some proper coordinate transformations, Kruppa's equations in (4.1.16) can be further simplified. Specifically, assume camera's intrinsic parameter matrix

$$A = \begin{bmatrix} f & 0 & u_0 \\ 0 & f & v_0 \\ 0 & 0 & 1 \end{bmatrix},$$

and the fundamental matrix  $F$ . We then transform the image coordinate system such that the principal point is at the origin of the coordinate system. At the new image coordinate system, the intrinsic parameter matrix becomes

$$A' = TA = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

where the transformation matrix  $T$  is

$$T = \begin{bmatrix} 1 & 0 & -u_0 \\ 0 & 1 & -v_0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Accordingly, the new fundamental matrix  $F'$  is  $F' = T^{-T} F T^{-1}$ .

Consider the SVD of  $F'$  is  $F' = U S V^T$ . Then the simplified Kruppa's equations (4.1.16) in turn yield:

$$\frac{a^2 v_1^T \text{diag}(f^2, f^2, 1) v_1}{u_2^T \text{diag}(f^2, f^2, 1) u_2} = \frac{b^2 v_2^T \text{diag}(f^2, f^2, 1) v_2}{u_1^T \text{diag}(f^2, f^2, 1) u_1} = \frac{ab v_1^T \text{diag}(f^2, f^2, 1) v_2}{-u_2^T \text{diag}(f^2, f^2, 1) u_1}$$

Expanding the above equations, we further obtain

$$\frac{a^2 (v_{11}^2 f^2 + v_{12}^2 f^2 + v_{13}^2)}{u_{21}^2 f^2 + u_{22}^2 f^2 + u_{23}^2} = \frac{b^2 (v_{21}^2 f^2 + v_{22}^2 f^2 + v_{23}^2)}{u_{11}^2 f^2 + u_{12}^2 f^2 + u_{13}^2} = -\frac{ab (v_{11} v_{21} f^2 + v_{12} v_{22} f^2 + v_{13} v_{23})}{u_{21} u_{11} f^2 + u_{22} u_{12} f^2 + u_{23} u_{13}}$$

Due to the orthogonality of  $U$  and  $V$ , the three fractions are rewritten as

$$\frac{a^2(1-v_{13}^2)f^2 + a^2v_{13}^2}{(1-u_{23}^2)f^2 + u_{23}^2} = \frac{b^2(1-v_{23}^2)f^2 + b^2v_{23}^2}{(1-u_{13}^2)f^2 + u_{13}^2} = -\frac{abv_{13}v_{23}}{u_{23}u_{13}} = s.$$

$$(4.3.1)$$

Note that the rightmost fraction degenerates to a constant factor  $s$ . Rearranging the equation (4.3.1), we obtain

$$f^2(au_{13}u_{23}(1-v_{13}^2) + bv_{13}v_{23}(1-u_{23}^2)) + u_{23}v_{13}(au_{13}v_{13} + bu_{23}v_{23}) = 0 \quad (4.3.2)$$

$$f^2(av_{13}v_{23}(1-u_{13}^2) + bu_{13}u_{23}(1-v_{23}^2)) + u_{13}v_{23}(au_{13}v_{13} + bu_{23}v_{23}) = 0 \quad (4.3.3)$$

and

$$\begin{aligned} & f^4(a^2(1-u_{13}^2)(1-v_{13}^2) - b^2(1-u_{23}^2)(1-v_{23}^2)) + f^2(a^2(u_{13}^2 + v_{13}^2 - 2u_{13}^2v_{13}^2) \\ & - b^2(u_{23}^2 + v_{23}^2 - 2u_{23}^2v_{23}^2)) + (a^2u_{13}^2v_{13}^2 - b^2u_{23}^2v_{23}^2) = 0 \end{aligned} \quad (4.3.4)$$

These two linear and one quadratic equations are our calibration equations. From equation (4.3.1), we know these three equations are dependent. However, we will find that they degenerate in different cases in the next chapter.

The advantages of the above algorithm are that: (1). Kruppa's equations degenerate to two linear and one quadratic equations, with the focal length of the camera in closed form in those equations; and (2). Singular cases of these equations are explicit and easy to find. We will discuss them in the next chapter.

## Chapter 5 Singular cases analyses

In the last chapter, we know camera self-calibration is equivalent to recovering the image of the absolute conic (IAC). However, some special motion sequences of a camera may lead to a spurious IAC. Accordingly, the calibrated parameters of the camera are not correct. In fact, when these motion sequences happen, camera cannot be calibrated by any algorithm. Sturm calls such motion sequences the critical motion sequences for self-calibration [31]. We call the geometric configurations corresponding to critical motion sequences the singular cases (or singularities) of a calibration algorithm.

In this chapter, Sturm's work on critical motion sequences is presented in Section 5.1. In Section 5.2, we then focus on the discussion of singular cases of our calibration algorithm, which is given in Section 4.2, Chapter 4. We work out two ways to obtain the singular cases. One way is based on heuristic analyses, which are an extension of Sturm's work [32]. The other is based on algebraic derivation. We then find that, for the quadratic equation (4.3.4), the singular cases obtained from both approaches are identical; for the linear equations (4.3.2) and (4.3.3), the two sets of singular cases are a little different. Considering the practical importance, we then argue that a subset of the singular cases obtained from the linear equations, i.e., the coplanar optical axes, is the singular case of our calibration algorithm.

### 5.1 Critical motion sequences for camera self-calibration

The work of Sturm [31][32] on critical motion sequences is presented here. A critical motion sequence is a sequence of camera motions that give spurious results for self-calibration or structure recovery. Naturally, if we cannot self-calibrate a camera from a motion sequence, then structure recovery is also impossible. However, the in-

verse of this conclusion is not necessarily true. Later, we will discuss this point in detail.

### 5.1.1 Potential absolute conics

From the discussion of the last two chapters, we know that camera self-calibration is nothing but the recovery of the image of the absolute conic (IAC), and that metric reconstruction is the recovery of the absolute conic (AC). The main property of IAC is that it is invariant under any rigid motion. However, it is quite possible that the image of a "normal"<sup>1</sup> conic is invariant under some special rigid motions. Hence, self-calibration from these camera motions means the recovery of the image of some conic other than the absolute conic. We call such a conic the Potential Absolute Conic (PAC).

The PAC is one kind of Proper Virtual Conic (PVC). All PVCs are of central symmetry [1], and hence they can be transformed to their Euclidean normal form. The Euclidean normal form of a PVC in  $P^2$  is represented by a  $3 \times 3$  diagonal matrix, where the diagonal elements are the conic's three eigenvalues. If the three eigenvalues are all distinct, then the conic is an ellipse. Otherwise, it is a circle.

### 5.1.2 PAC on the plane at infinity

In this case, Sturm proposes that if the eigenvalues of the PAC are identical, the PAC is exactly the absolute conic (AC). Hence motion sequences are of course not critical. If there are only double distinct eigenvalues, then the eigenspaces of the PAC are a plane and a line orthogonal to the plane. Hence, any rotation about the line or a rotation of  $180^\circ$  about a line in the plane that is incident to the other line, will leave the

---

<sup>1</sup> We say it is normal since it is other than the absolute conic.

PAC unchanged and thus it is critical. If three eigenvalues are all distinct, then the eigenspaces of the PAC are three orthogonal lines. Hence, a rotation about any one of the three lines, or a rotation of  $180^\circ$  about lines perpendicular to a line, preserves the PAC. Such motion is, of course, critical.

### 5.1.3 PAC not on the plane at infinity

In this case, Sturm works out critical sequences in this way: He starts from a PAC and from a specific camera position, and then attempts to find all rigid motions of the camera that will obtain identical images for the PAC.

The above proposal simplifies the problem. Sturm then find that camera centers in critical motion sequences are on two circles at equal distance from the plane supporting the PAC.

Consider the cone  $K$  that contains the projection of the PAC and the camera center. Sturm then draws the following conclusion for critical motion sequences:

1. If  $K$  is a circular cone and the PAC is a circle, then the camera centers can only be in two different positions along the line perpendicular to the plane supporting the PAC. These two positions form a pair of reflections with respect to the plane. The camera may rotate about the line by any angle, or by  $180^\circ$  about a line perpendicular to the line linking the two reflections.
2. If  $K$  is a circular cone and the PAC is an ellipse, then there are only four possible camera positions. All of them are located in the plane that is perpendicular to the supporting plane and contains the main axis of the ellipse. The camera can rotate about the projective axis by any angle, or by  $180^\circ$  about the line perpendicular to it.

3. If  $K$  is an elliptic cone and the PAC is also an ellipse, then there are eight possible camera positions. They form a rectangular parallelepiped. At each position, four orientations are possible [31].
4. If  $K$  is an elliptic cone and the PAC is a circle, then the camera centers are on the two circles (refer to the description in second paragraph of this section). In each position, there are four possible orientations as in case 3.

#### 5.1.4 Useful critical motion sequences in practice

The above description of the critical motions is purely heuristic. However, there are some special cases corresponding to the above analysis, which are practically useful.

We learn from Section 5.1.2 that if two of the PAC's eigenvalues are identical, then the eigenvectors corresponding to the same eigenvalue span a plane  $\Pi$ . Consequently camera centers are on the plane that is parallel to the plane  $\Pi$ , and thus make the motion sequence critical. Therefore, planar motions are critical for camera self-calibration. It is even impossible to do self-calibration for cameras rotating about parallel axis and undergoing arbitrary translation, since it is obvious that in these cases, the cameras will definitely obtain the same images for a PAC. By the same principle, orbital motions are critical for camera self-calibration. It should be noted that pure rotations are not critical for camera self-calibration although there are infinite PACs that give the same projections on the image planes. The reason is that all of the PACs lie on the same projection cone as the absolute conic (AC). However, pure rotations are critical for affine reconstruction and then metric reconstruction since there are infinite PACs not lying on the plane at infinity. Figure 5.1 shows such motions.

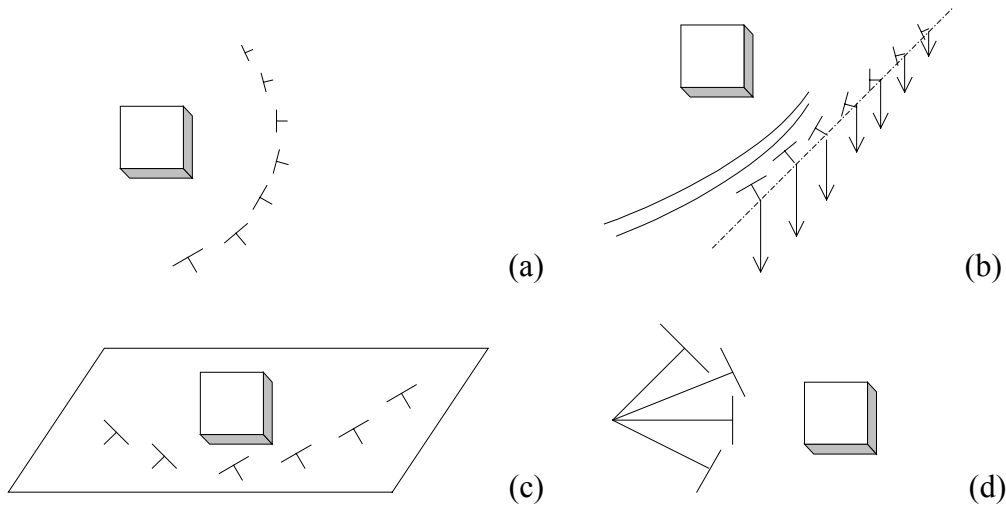


Figure 5.1: Illustration of critical motion sequences. (a) Orbital motion. (b) Rotation about parallel axes and arbitrary translation. (c) Planar motion (d) Pure rotations (not critical for self-calibration but for the scene reconstruction).

## 5.2 Singular cases for the calibration algorithm in Section 4.3

In this section, singular cases of our focal length calibration algorithm are analyzed. We work out two ways to obtain the singular cases. One way is based on heuristic analyses, which are an extension of Sturm's work [32]. They are presented in Section 5.2.2. The other is based on algebraic derivation. We then find, for the quadratic equation (4.3.4), the singular cases obtained from both approaches are identical; for the linear equations (4.3.2) and (4.3.3), the two sets of singular cases are a little different. Considering the practical importance, we then argue that a subset of the singular cases obtained from the linear equations, i.e., the coplanar optical axes, is the singular case of our calibration algorithm.

### 5.2.1 Generic singularities

We first state that based on the assumption of our calibration algorithm in Section 4.3 (that is, only the focal length is unknown but constant), singular cases occur when:

1. The optical axes are parallel to each other or

2. The optical axes intersect in a finite point and the optical centers are equidistant from this point.

For convenience, we call these geometric configurations the generic singularities of the calibration algorithm.

### 5.2.2 Heuristic interpretation of generic singularities

The assumptions of the discussion on critical motions in Section 5.1 are that all of five intrinsic parameters are unknown and constant in motion sequences. However, if the assumptions are relaxed to allow the hypothesis that only the focal length is unknown, the singular cases are made more specific. Sturm's research results from [32] give the background knowledge for obtaining the generic singularities of our calibration algorithm.

Assume that focal length is varied and other parameters are known when a camera undergoes rigid camera motions, Sturm proposes that, in the scenario described in the last paragraph, the projection of a PAC is a proper virtual circle  $\phi$ . It is then argued that when

1. PAC is on the plane at infinity  $\Pi_\infty$ ,

singular cases occur when optical axes are parallel to each other;

2. PAC is not on the plane at infinity  $\Pi_\infty$

there are three different singular cases:

*Case 1.* The optical axes are parallel to each other.

*Case 2.* The optical centers are collinear and the line passing through all the optical centers is perpendicular to the plane supporting the PAC.

*Case 3.* The optical centers lie on an ellipse/hyperbola pair as shown in Figure 5.2. The optical axes are all tangent to the ellipse or hyperbola.



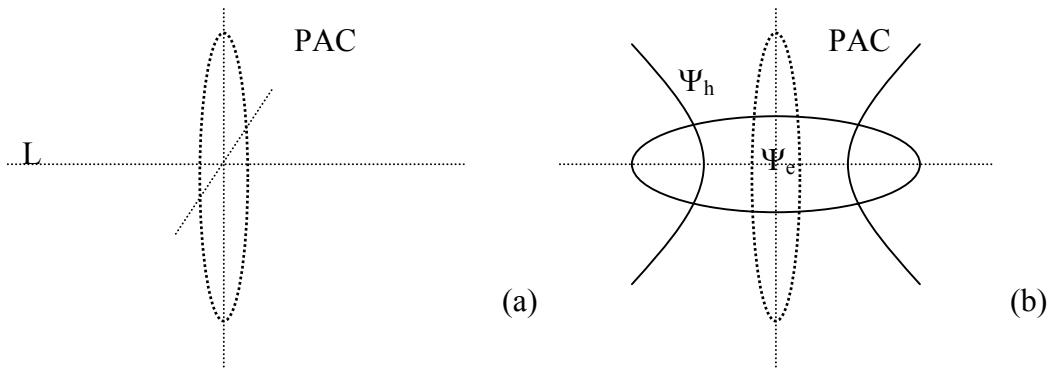


Figure 5.2: Possible camera center positions when the PAC is not on  $\Pi_\infty$ . (a) The PAC is a proper virtual circle. All the camera centers are on the line  $L$ . (b) The PAC is a proper virtual ellipse. All the camera centers are on a pair of ellipse/hyperbola.

Actually, Case 2 occurs when the PAC is a proper virtual circle. The camera orientations can be the rotations about  $L$  except that the camera centers are in two special positions. In these two positions, the projections of the PAC are the same as that of the absolute conic (AC), and thus the cameras can be in arbitrary orientations. In other words, Case 2 means that camera undergoes pure forward translations with two exceptions. Case 3 occurs when the PAC is a proper virtual ellipse. Then the camera centers are on an ellipse/hyperbola pair. The supporting planes of the ellipse, hyperbola and the PAC are mutually perpendicular. The optical axes are then in the directions of the tangents of the ellipse/hyperbola pair.

Now consider a two-view system and assume that camera's focal length is constant. If a camera undergoes general motions, Case 2 does not apply. Consider Case 1, if two optical axes are parallel, the corresponding motion is a critical motion. Consider Case 3, if two camera centers are either on the hyperbola or the ellipse, a critical motion also takes place. Since we have assumed that the focal length is constant, the two camera centers are certainly symmetrical about one of the hyperbola or ellipse's axes. Therefore, the two camera centers are equidistant from the intersection of the two optical

axes. Those singularities are exactly generic singularities we propose in the last subsection. Figure 5.3 shows those generic singularities.

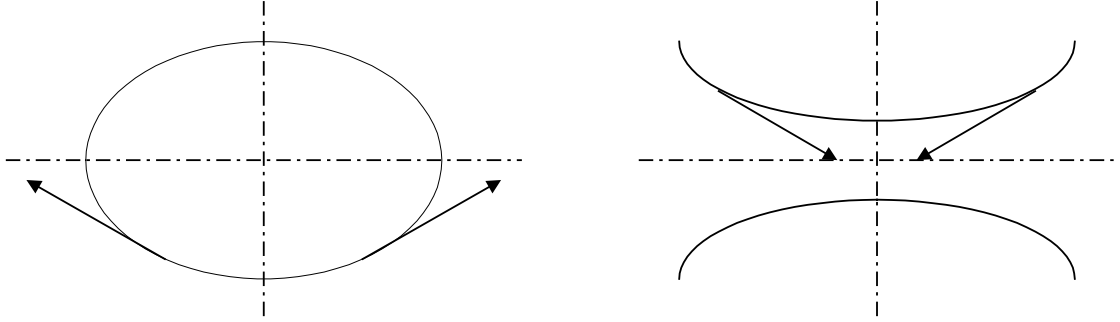


Figure 5.3: Illustration of the equidistant case (arrows show the directions of camera's optical axes)

### 5.2.3 Algebraic interpretation of generic singularities

Since camera's intrinsic parameters except focal length are known, we can transform image correspondences from uncalibrated space to a so-called semi-calibrated space. The corresponding fundamental matrix is then called the semi-calibrated fundamental matrix  $G$ . Specifically, consider the rotation matrix  $R$  and translation vector  $\mathbf{t}$  (Here  $\mathbf{t}$  is the translation that brings the coordinates of the second camera to the first camera. It is different from the  $\mathbf{t}$  defined in Section 3.2). The fundamental matrix is then  $F \sim A^{-T} R[\mathbf{t}]_x A^{-1}$ . Hence  $G$  is

$$G \sim \begin{bmatrix} \tau & 0 & 0 \\ 0 & 1 & 0 \\ u_0 & v_0 & 1 \end{bmatrix} F \begin{bmatrix} \tau & 0 & u_0 \\ 0 & 1 & v_0 \\ 0 & 0 & 1 \end{bmatrix} \sim \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & f \end{bmatrix} R[\mathbf{t}]_x \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & f \end{bmatrix}, \quad (5.2.1)$$

where  $\tau$  is the aspect ratio,  $(u_0, v_0)$  is the principal point and  $f$  is the focal length.

Based on  $G$ , coplanarity of optical axes can be algebraically expressed as  $G_{33} = 0$ . (Coplanarity of optical axes means that two principal points are image correspondences satisfying the epipolar geometry. In  $G$ , the homogeneous coordinates of the two principal points are  $[0 \ 0 \ 1]^T$ . That follows the conclusion). Here  $G_{33}$  is the lower-rightmost element of  $G$ .

We duplicate the equations (4.3.2), (4.3.3) and (4.3.4) here for ease of reference.

$$f^2(au_{13}u_{23}(1-v_{13}^2) + bv_{13}v_{23}(1-u_{23}^2)) + u_{23}v_{13}(au_{13}v_{13} + bu_{23}v_{23}) = 0 \quad (5.2.2)$$

$$f^2(av_{13}v_{23}(1-u_{13}^2) + bu_{13}u_{23}(1-v_{23}^2)) + u_{13}v_{23}(au_{13}v_{13} + bu_{23}v_{23}) = 0 \quad (5.2.3)$$

$$f^4(a^2(1-u_{13}^2)(1-v_{13}^2) - b^2(1-u_{23}^2)(1-v_{23}^2)) + f^2(a^2(u_{13}^2 + v_{13}^2 - 2u_{13}^2v_{13}^2) - b^2(u_{23}^2 + v_{23}^2 - 2u_{23}^2v_{23}^2)) + (a^2u_{13}^2v_{13}^2 - b^2u_{23}^2v_{23}^2) = 0 \quad (5.2.4)$$

Based on the above notations, the coplanarity of optical axes is reinterpreted as

$$G_{33} = au_{13}v_{13} + bu_{23}v_{23} = 0.$$

The linear equation (5.2.2) is singular if and only if

$$u_{23}v_{13}(au_{13}v_{13} + bu_{23}v_{23}) = 0 \text{ and } au_{13}u_{23}(1-v_{13}^2) + bv_{13}v_{23}(1-u_{23}^2) = 0 \quad (5.2.5)$$

The linear equation (5.2.3) is singular if and only if

$$u_{13}v_{23}(au_{13}v_{13} + bu_{23}v_{23}) = 0 \text{ and } av_{13}v_{23}(1-u_{13}^2) + bu_{13}u_{23}(1-v_{23}^2) = 0 \quad (5.2.6)$$

The conditions (5.2.5) and (5.2.6) can be reduced to the following sub-conditions:

$$u_{23} = v_{13} = 0 \quad (5.2.5)$$

$$u_{23} = v_{23} = 0 \quad (5.2.6)$$

$$u_{13} = v_{13} = 0 \quad (5.2.7)$$

$$u_{13} = v_{23} = 0 \quad (5.2.8)$$

$$v_{13} = \pm u_{23} \text{ and } au_{13} = \mp bv_{23} \quad (5.2.9)$$

$$v_{23} = \pm u_{13} \text{ and } av_{13} = \mp bu_{23} . \quad (5.2.10)$$

Among the above six conditions, only equations (5.2.6) and (5.2.7) do not correspond to coplanar optical axes. The rest are all generic singularities.

The quadratic equation (5.2.4) is degenerate when

$$a^2u_{13}^2v_{13}^2 - b^2u_{23}^2v_{23}^2 = 0, \quad (5.2.11)$$

$$a^2(u_{13}^2 + v_{13}^2 - 2u_{13}^2v_{13}^2) - b^2(u_{23}^2 + v_{23}^2 - 2u_{23}^2v_{23}^2) = 0, \quad (5.2.12)$$

and

$$a^2(1 - u_{13}^2)(1 - v_{13}^2) - b^2(1 - u_{23}^2)(1 - v_{23}^2) = 0. \quad (5.2.13)$$

The above conditions then can be reduced to

$$a = b, u_{23}^2 = v_{13}^2 \text{ and } v_{23}^2 = u_{13}^2 \quad (5.2.14)$$

or

$$a = b, u_{23}^2 = u_{13}^2 \text{ and } v_{23}^2 = v_{13}^2 \quad (5.2.15)$$

They are equivalent to

$$a = b, u_{23} = \pm v_{13} \text{ and } v_{23} = \mp u_{13} \quad (5.2.16)$$

$$a = b, u_{23} = \pm u_{13} \text{ and } v_{23} = \mp v_{13} \quad (5.2.17)$$

$$a = b, u_{23} = \pm v_{13} \text{ and } v_{23} = \pm u_{13} \quad (5.2.18)$$

$$a = b, u_{23} = \pm u_{13} \text{ and } v_{23} = \pm v_{13} \quad (5.2.19)$$

Equations (5.2.16) and (5.2.17) correspond to coplanar optical axes, but equations (5.2.18) and (5.2.19) do not.

Analysis is then carried out in two scenarios, i.e., the coplanar optical axes and non-coplanar optical axes. For easier reading, only the main conclusions on singularities appear here. All of algebraic derivations and detail discussions are included in Appendix B.

Based on the above conditions, it can be seen that when two optical axes are coplanar, the two linear equations definitely vanish. However, the quadratic equation vanishes only when the two optical axes are parallel or when the two optical centers are equidistant from the intersection of the two optical axes. Specifically, when the two optical axes are parallel, the quadratic equation may vanish if the second optical center lies on the optical axis of the first camera, or may degenerates into one of the following two linear equations

$$f^2(a^2(1-u_{13}^2)-b^2(1-v_{23}^2))+(a^2u_{13}^2-b^2v_{23}^2)=0 \quad (5.2.20)$$

if  $u_{23} = v_{13} = 0$ , or

$$f^2(a^2(1-v_{13}^2)-b^2(1-u_{23}^2))+(a^2v_{13}^2-b^2u_{23}^2)=0 \quad (5.2.21)$$

if  $u_{13} = v_{13} = 0$

The latter degenerate case happens when the rotation angle measured from the horizontal axis is  $0^\circ$  or  $180^\circ$ . When the two optical axes are not parallel but coplanar, the linear equations still vanish. However, the quadratic equation reduces to one of the two linear equations above and is only singular in the equidistance case.

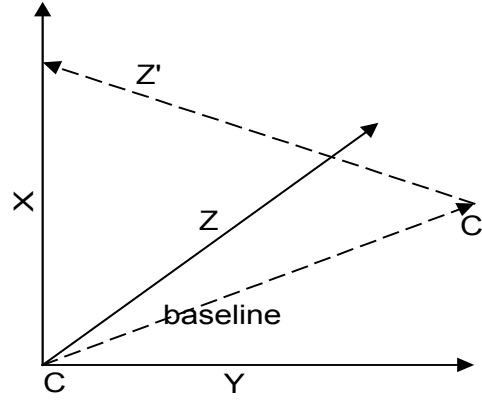


Figure 5.4: Configuration of non-generic singularity for the linear equations

When the two optical axes are not coplanar, it is found that there is one non-generic singularity for the linear equations. That is, the second optical axis lies in the plane, which contains the baseline and is orthogonal to the plane spanned by the baseline and the first optical axis. Of course, this is of little practical importance. Figure 5.4 shows such a configuration. For the quadratic equation, however, there is no non-generic singularity. That means the equation (5.2.18) or (5.2.19) alone can only give spurious solutions, which can be easily eliminated.

#### 5.2.4 Conclusion

From the analysis in Section 5.2.2 and 5.2.3, we know that there is no non-generic singularity for the quadratic equation (5.2.4), i.e., both singular cases obtained from the heuristic analysis in Section 5.2.2 and algebraic analysis in Section 5.2.3 are identical. The two linear equations (5.2.2) and (5.2.3), however, not only degenerate in generic singular cases, but also in non-generic singular cases. Therefore, the singularities of the quadratic equation are a subset of those of the linear equations. However, it is easy to avoid coplanarity of optical axes. Hence, practically, we consider coplanarity of optical axes as the singularity of our calibration algorithm.

## Chapter 6 Experiment results

We carried out a large number of experiments in order to study the performance of the algorithm and examine its singular cases described in Chapters 3 and 4. Section 6.1 presents the results of experiments in which synthetic images were used to assess the performance of the algorithm and detect the various singular cases for the linear and quadratic equations. Experiments on real images were then conducted and results are reported in Section 6.2. First, a special calibration grid was employed in order to obtain good matches. At this stage, the performance of the algorithm was evaluated rigorously. It has been found that the quality of the algorithm was largely determined by the relation between the two optical axes --- whether they were coplanar or not. At last, two arbitrary scenes --- one containing three cups and the other containing a building, were used to calibrate the camera. The results are presented and analyzed in Section 6.2.3.

### 6.1 Experiment involving synthetic object

In this experiment, a synthetic object was used to do calibration. The experiment was conducted in two steps. First, the performance of our calibration algorithm with respect to Gaussian noise level was evaluated. Next, singular cases for the linear and quadratic equations were investigated and verified in the experiment.

#### 6.1.1 Synthetic object and images

The configuration used in this experiment is shown in Figure 6.1. The object is composed of two planar grids that form a  $135^\circ$  angle with each other. In each grid, there

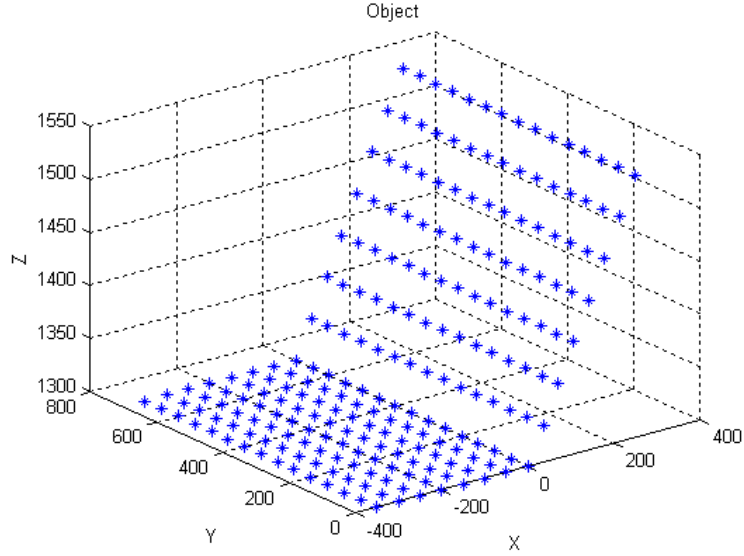


Figure 6.1: The synthetic object

are a total of 105 points. As Figure 6.1 shows, the object is placed at a distance of 1,300 units from the camera center. We assume that two images are taken with a camera motion. Without loss of generality, the first camera matrix is  $P_1 = A[\mathbf{I} | \mathbf{0}]$  and the second camera matrix is  $P_2 = A[\mathbf{R} | \mathbf{t}]$ , where  $\mathbf{I}$  is the  $3 \times 3$  identity matrix,  $\mathbf{0}$  the null 3-vector,  $\mathbf{R}$  and  $\mathbf{t}$  are the orientation and position of the second camera with respect to the first camera coordinate system. The camera's intrinsic parameters are:  $f_u = f_v = 600$ ,  $u_0 = 320$ ,  $v_0 = 240$ , and the skew factor  $\beta = 0$ .

### 6.1.2 Performance with respect to Gaussian noise level

It has been shown that, practically, the coplanarity of optical axes is the singularity of our calibration algorithm in Section 4.3, Chapter 4. Based on this fact, we designed a two-camera system in which the two optical axes are purposely avoided to being coplanar. In this experiment, the image coordinates of the grid points were perturbed by independent Gaussian noise with mean of 0 and a standard deviation of  $\sigma$  pixels. The noise level was varied from 0.1 pixels to 2.0 pixels. For each noise level, a total of 100 trials were performed. Therefore, in each experiment corresponding to each noise



level, there were 100 calibration results in total. The average of these 100 data sets was then taken as the calibrated focal length with respect to this noise level. The estimated focal length was then compared with the "ground truth" given in the last subsection. The relation between the relative error of focal length and noise level is shown in Figure 6.2.

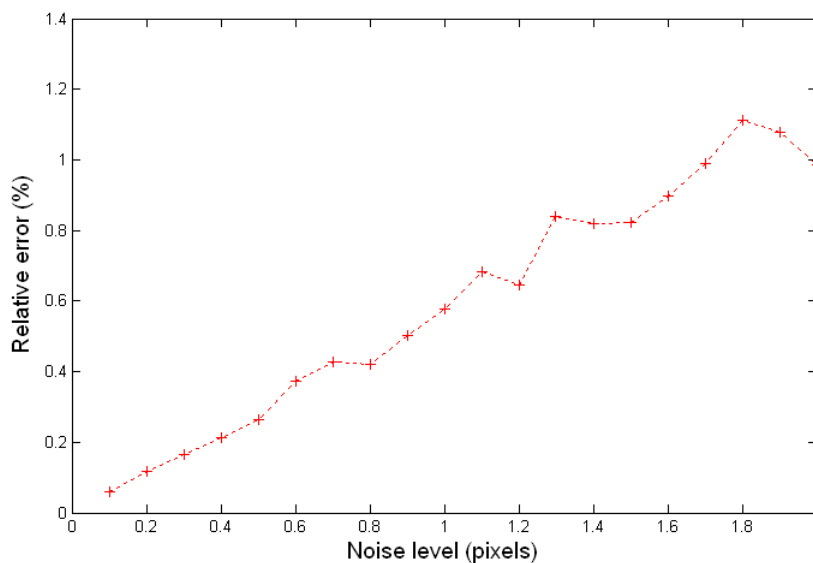


Figure 6.2: Relative error of focal length with respect to Gaussian noise level

The results shown in Figure 6.2 are obtained by using the quadratic equation. It can be seen that the relative errors of focal length generally increase with the noise level. However, at some noise levels such as level 0.8, the errors are less than those in the intervals with less noise levels. A probable reason is that image noise may not strictly be Gaussian noise.

### 6.1.3 Detecting different singular cases for the linear and quadratic equations

As described in the last chapter, the linear equations generally degenerate when the two optical axes are coplanar and the quadratic one degenerates in generic singular cases. In this experiment, we try to simulate the coplanarity of optical axes and generic

singular cases in order to observe the performance of the linear and quadratic equations.

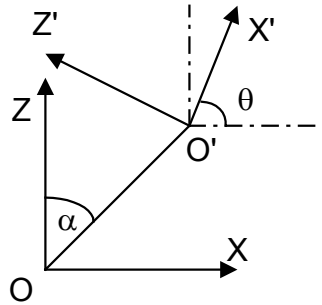


Figure 6.3: Coordinates of two cameras

In order to detect the difference of singular cases between the linear and quadratic equations, we first emulated the coplanarity of two optical axes. Figure 6.3 shows the two camera coordinates. Here, the  $Y$  axis points towards the reader. The translation is on the  $XOZ$  plane. Hence, coplanarity means that the camera rotates only about the  $Y$  axis by  $\theta$ . The baseline forms an angle  $\alpha$  with the  $Z$  axis. If the two optical axes are parallel, then  $\theta = 0^\circ$  (here we only consider  $\theta < 180^\circ$ ). If the two optical axes form the equidistance case,  $\theta = 180^\circ - 2\alpha$ . In this experiment,  $\alpha$  is set to be  $45^\circ$ . Then  $\theta = 90^\circ$  is equivalent to the equidistance case. In a word, when  $\theta$  is equal to  $0^\circ$  and  $90^\circ$ , the calibration algorithm works in generic singular cases.

In this experiment, the noise level is 0.2 pixels and the every experiment associated with one rotation angle was repeated with 100 trials. The result was the average of every 100 data.

Figures 6.4, 6.5 and 6.6 show the performance of the quadratic and linear equations when the two optical axes are coplanar. The horizontal axis represents the rotation angle  $\theta$  and vertical axis represents the relative error of focal length. Figure 6.4 shows that the relative errors of the quadratic equation are 2~7 times less than those of the linear equations. Hence in non-generic singular cases, the quadratic equation demon-

strates better performance than the two linear equations. Figure 6.5 and Figure 6.6 show that when the two-camera setup approaches to be generic singular, i.e.,  $\theta$  approaches to be  $0^\circ$  or  $180^\circ$ , the quadratic equation gives poor performance --- the relative errors increase from nearly 20% to 400%. In contrast to the quadratic equation, the linear equations show less variation of the errors in these two figures. Hence, the linear equations are more stable than the quadratic equation when the two-camera system approaches to be generic singular. However, this phenomenon is of little practical importance since the relative errors are greater than the expected (they are usually should be less than 15%).

In a nutshell, this experiment clearly demonstrates the difference of singularities between the quadratic and the linear equations, i.e., the quadratic equation degenerates in generic singular cases, and the linear equations degenerate when the two optical axes are coplanar.

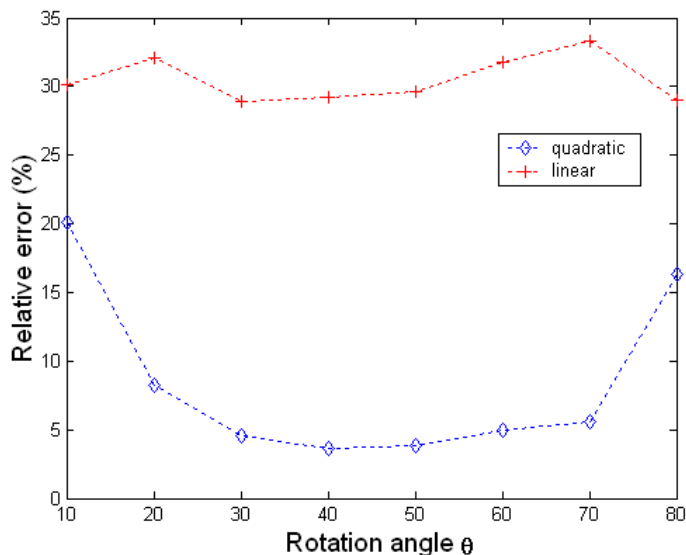


Figure 6.4: Coplanar optical axes (neither parallel nor equidistance case)

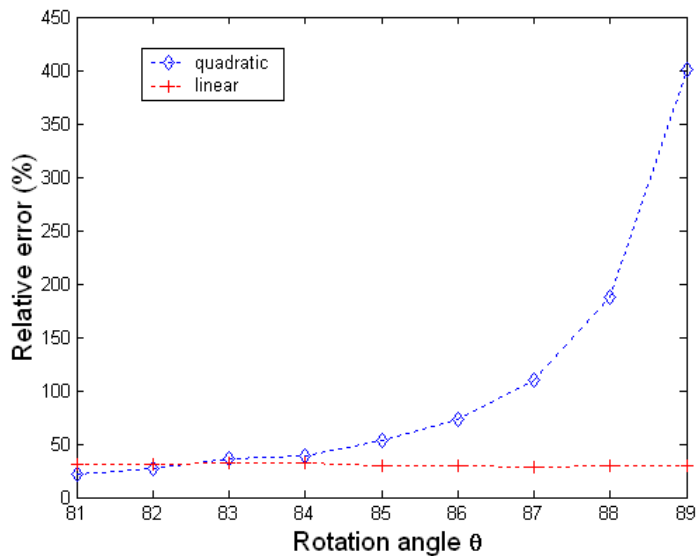


Figure 6.5: The two camera centers are near to be equidistant from the intersection of the two optical axes

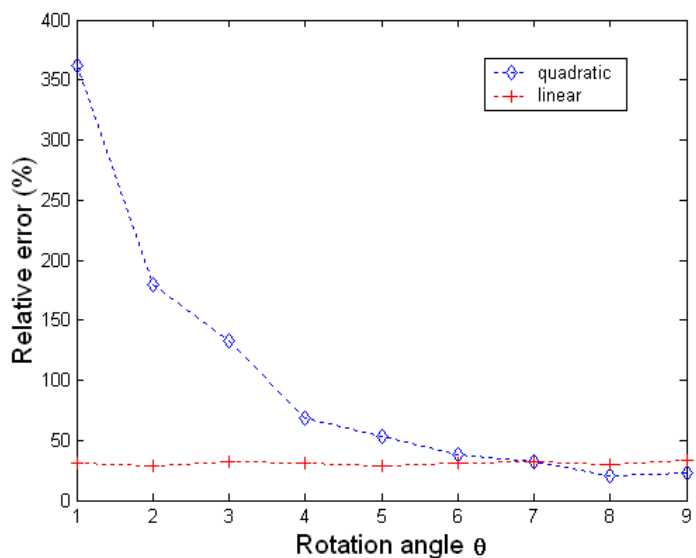


Figure 6.6: The two optical axes are near parallel

## 6.2 Experiment involving actual images

In this part of experiment, we first measured the effect of the assumption of the principal point position on the focal length calibration. Next, tests were carried out to quantify the sensitivity of the algorithm with respect to *a numerical entity* that gauges the coplanarity of optical axes. We then used the calibrated results to do reconstruction

in order to evaluate the effectiveness of this algorithm. All of the above procedures employed the precise calibration pattern routinely used in INRIA-Grenoble. Two sets of arbitrary scenes were then used for self-calibration.

Note: in order to obtain good image correspondences, all experiments consider narrow baseline.

### 6.2.1 Camera setup

Practically, it is easy to avoid coplanarity of optical axes. There are multiple ways to achieve this goal. One approach is as follows: Since it is often safe to assume that the principal point is at the image center, when taking images with a camera, we first focus the image center on one object point and then take one image. Next we move the camera horizontally and try to make the image center nearly at the same object point. Then we tilt the camera upwards or downwards. Such an arrangement ensures that the two optical axes are not on the same plane.

The camera we used in all experiments is a Sony DSC-P31 digital camera with 5mm focal length. We first used the software toolkit Tele2 [23] developed in INRIA-Grenoble to calibrate the camera. The resulting focal length of 625 pixels is used as "ground truth" in the following experiments.

### 6.2.2 Experiment involving images taken from a special object

In this experiment, the special calibration object routinely used in INRIA-Grenoble was applied for calibration. This calibration object consists of three planes, namely, Face 1, Face 2 and Face 3. Face 1 forms a  $90^\circ$  angle with Face 2 and Face 3. The angle between Face 2 and Face 3 is  $120^\circ$ . In Face 1, there are 62 white dots that are carefully designed with the tolerance less than 1mm. Face 2 and Face 3 both have 49 white dots respectively.

In total 10 images were used in this experiment. The images were taken from ten different positions, covering a roughly circular path around the grid. Specifically, first, we took one image in one position. This position was purposely designated as the left-most position. We tilted the camera as described in Section 6.2.1. We then moved to the right to a new position and take another picture. We did this ten times. The tenth position was of course the rightmost position. Since we applied small tilt angles, thus among 45 possible image pairs, some have approximately coplanar optical axes and some are quite distant from that situation. Here these 10 images are then called an image sequence. Figure 6.7 shows some pictures of this image sequence. The resolution of all these images is  $640 \times 480$ .



Figure 6.7: Some images of the calibration grid

## I. Effect of principal point estimation on the focal length calibration

As described before, this focal length calibration algorithm is based on the assumption that the other intrinsic parameters are known. This experiment demonstrates that the principal point estimation error has little effect on the focal length calibration when the aspect ratio is assumed to be 1 and the skew factor 0.

We selected the first and last images from the image sequence. The experiment procedure was as follows: First we assumed that the principal point was at the point that deviates from the image center by  $-25$  pixels, along both the horizontal and the vertical axes. We then used this assumed principal point to calibrate the focal length. We next moved the assumed principal point along the positive horizontal axis by 5 pixels. The new resulting principal point was then used to calibrate the camera. After repeating this procedure by 11 times, we moved back and next moved along the vertical axis by 5 pixels. We then moved along the horizontal axis in steps of 5 pixels as described above. We kept moving until we finally obtained 121 focal lengths.

Table 6.1 and Figure 6.8 show the calibrated 121 focal lengths in total. In Table 6.1, the row represents the displacement of the principal point in horizontal axis. The column represents the displacement of the principal point in vertical axis. In Figure 6.8, the horizontal axis represents the deviation of the principal point from the image center. The vertical axis represents the mean of all focal lengths calibrated from all of the cases when one coordinate of the principal point is constant and the other is changing.

Table 6.1: Calibration results with respect to the principal point estimation

<b>f</b>	<b>-25</b>	<b>-20</b>	<b>-15</b>	<b>-10</b>	<b>-5</b>	<b>0</b>	<b>5</b>	<b>10</b>	<b>15</b>	<b>20</b>	<b>25</b>
<b>-25</b>	624.3	621.1	618.0	615.2	612.5	610.0	607.6	605.4	603.4	601.6	599.9
<b>-20</b>	626.7	623.6	620.7	617.9	615.3	612.9	610.7	608.6	606.7	605.0	603.4
<b>-15</b>	629.2	626.2	623.3	620.7	618.2	615.9	613.8	611.9	610.1	608.4	607.0
<b>-10</b>	631.7	628.8	626.1	623.5	621.2	619.0	617.0	615.1	613.4	611.9	610.6
<b>-5</b>	634.3	631.5	628.8	626.4	624.1	622.0	620.1	618.4	616.8	615.4	614.1
<b>0</b>	636.9	634.2	631.7	629.3	627.1	625.1	623.3	621.7	620.2	618.9	617.8
<b>5</b>	639.6	637.0	634.5	632.3	630.2	628.3	626.6	625.0	623.7	622.5	621.4

<b>10</b>	642.4	639.8	637.5	635.3	633.3	631.5	629.9	628.4	627.1	626.0	625.1
<b>15</b>	645.2	642.7	640.4	638.3	636.4	634.7	633.2	631.8	630.7	629.7	628.8
<b>20</b>	648.1	645.7	643.5	641.5	639.7	638.0	636.6	635.3	634.2	633.3	632.6
<b>25</b>	651.0	648.7	646.6	644.7	642.9	641.4	640.0	638.8	637.8	637.0	636.4

From Figure 6.8, we find that even if the principal point deviates from the image center by 25 pixels along one direction, the relative error is less than 3% of the "true" focal length. The standard deviation of these 121 focal lengths is 11.7 pixels, which is only 1.8% of the focal length. The conclusion is that principal point estimation has little effect on the focal length calibration. Hence it is safe to assume that the principal point is at the image center when we use this algorithm for focal length calibration.

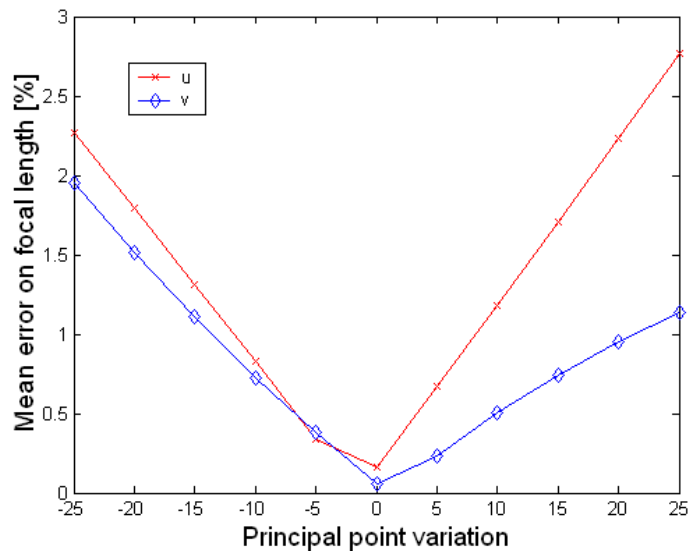


Figure 6.8: Effect of the principal point estimation on the focal length calibration

## II. Experiment considering the stability of the algorithm

In order to show that the algorithm is stable, the whole image sequence was used. The experiment considered all the possible combinations of any two images selected from those 10 images. The final results are presented in Figure 6.10 and Table 6.2. From these results, we can easily find some instances close to the coplanar case. Here, in order to measure how close the two optical axes to become coplanar, we first intro-



duce a middle plane. This middle plane has the same angle with the two planes passing through the baseline and two optical axes. Figure 6.9 shows such a middle plane.

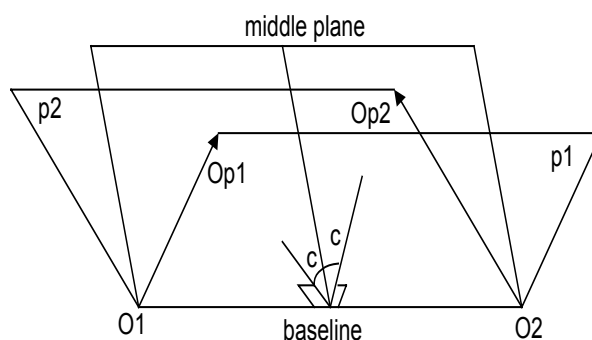


Figure 6.9: The middle plane

In Figure 6.9, the camera center  $O1$  and the optical axis  $Op1$  determine the plane  $P1$ . In the same principle,  $O2$  and  $Op2$  determine the plane  $P2$ . Thus the middle plane is the plane that has the same angle with  $P1$  and  $P2$ . Then whenever the two optical axes are coplanar, the angle  $c$  is zero.

In addition to the middle plane, the angle of the two optical axes is also used to determine whether they are parallel.

For 10 images, there are 45 pairings involving any two images, and thus there are totally 45 data points as shown in Figure 6.10.

Table 6.2: Experiment considering the stability of this algorithm

	ln f	q f	c	aifa	class
1-2	636.2353	636.2395	4.8573	6.0396	1
1-3	<b>600.1616</b>	<b>600.1912</b>	<b>0.5897</b>	<b>22.0678</b>	<b>2</b>
1-4	623.0289	623.0317	6.5393	22.2417	1
1-5	<b>484.2163</b>	<b>486.643</b>	<b>0.1736</b>	<b>36.6282</b>	<b>2</b>
1-6	623.9774	623.9774	7.8822	39.9875	1
1-7	<b>593.3703</b>	<b>593.6393</b>	<b>1.0431</b>	<b>30.9349</b>	<b>2</b>
1-8	621.2885	621.2889	8.1031	35.2266	1
1-9	<b>589.6953</b>	<b>589.71</b>	<b>1.1768</b>	<b>46.3984</b>	<b>2</b>
1-10	625.1382	625.1398	8.1051	51.4649	1
2-3	630.9762	630.9808	3.895	17.5298	1
2-4	617.8473	617.8569	3.5662	16.5518	1
2-5	634.5881	634.5937	4.2864	32.4806	1
2-6	623.6124	623.6145	4.519	34.797	1
2-7	640.0187	640.0226	3.1195	26.3194	1
2-8	617.2789	617.3563	5.1751	29.8387	1
2-9	643.8241	643.8366	3.3721	42.648	1
2-10	623.2157	623.2172	4.0974	46.9018	1

3-4	626.4769	626.4804	4.5722	8.2717	1
3-5	<b>645.546</b>	<b>645.5533</b>	<b>0.4692</b>	<b>14.9909</b>	<b>2</b>
3-6	625.4553	625.4643	6.2595	18.1764	1
3-7	<b>582.4486</b>	<b>582.5931</b>	<b>0.4708</b>	<b>8.8291</b>	<b>2</b>
3-8	619.7153	619.7275	9.1716	14.5359	1
3-9	<b>575.4857</b>	<b>575.5017</b>	<b>0.4711</b>	<b>25.5331</b>	<b>2</b>
3-10	630.2692	630.2941	6.7836	29.6834	1
4-5	626.3529	626.4069	5.4596	19.8588	1
4-6	<b>637.1915</b>	<b>637.1915</b>	<b>1.276</b>	<b>19.3917</b>	<b>3</b>
4-7	627.7637	627.8199	4.8415	13.0145	1
4-8	620.514	620.5269	1.577	13.5955	1
4-9	631.2502	631.3146	5.2192	30.4966	1
4-10	<b>644.4139</b>	<b>644.4146</b>	<b>1.447</b>	<b>32.775</b>	<b>3</b>
5-6	621.9666	621.9669	2.9185	8.8633	1
5-7	<b>750.3711</b>	<b>754.1453</b>	<b>0.0876</b>	<b>6.9522</b>	<b>2</b>
5-8	632.4635	632.4653	14.705	12.221	1
5-9	<b>614.7621</b>	<b>614.7783</b>	<b>1.3369</b>	<b>10.9525</b>	<b>3</b>
5-10	633.8803	633.8822	7.5776	15.2936	1
6-7	626.1481	626.1482	6.6629	9.8358	1
6-8	<b>915.0939</b>	<b>989.8468</b>	<b>0.2226</b>	<b>6.7671</b>	<b>2</b>
6-9	634.6316	634.6531	9.705	15.115	1
6-10	<b>726.1053</b>	<b>727.373</b>	<b>0.685</b>	<b>14.2146</b>	<b>2</b>
7-8	626.9702	626.9791	3.4095	8.711	1
7-9	620.5535	620.5669	1.5752	17.8957	1
7-10	634.9671	634.9674	8.5253	21.2163	1
8-9	630.5471	630.5547	9.518	21.1004	1
8-10	<b>744.1491</b>	<b>757.852</b>	<b>0.9408</b>	<b>20.9112</b>	<b>2</b>
9-10	624.2798	624.2932	2.1835	8.9243	1

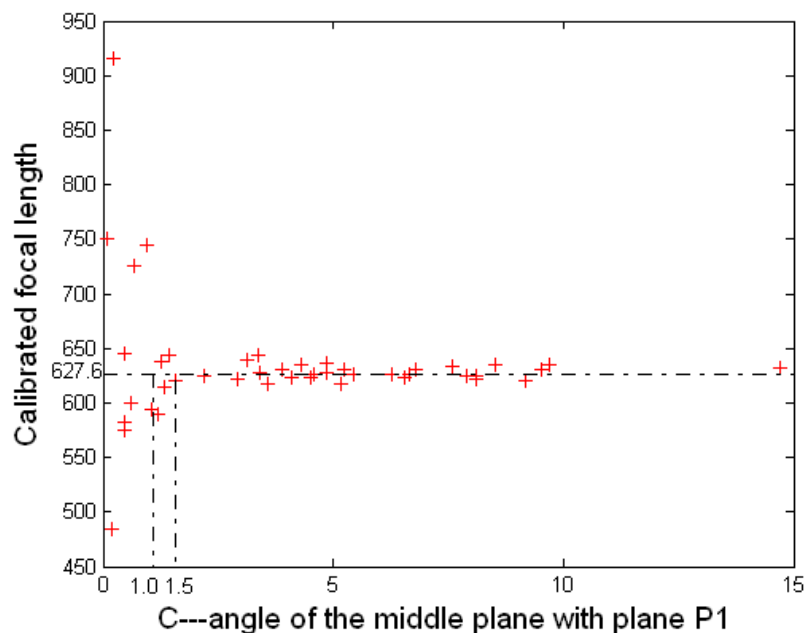


Figure 6.10: Sensitivity of focal length with respect to the angle  $c$

Several remarks can be made from Figure 6.10 and Table 6.2:

- In Table 6.2, the first column represents a possible image pair, i.e., 1-2 represents the combination of the image 1 and 2 and so on. The second column  $ln\_f$  represents the focal length calibrated from the linear equations. The third column  $q\_f$  represents the focal length calibrated from the quadratic equations. The fourth column  $c$  is the angle as shown in Figure 6.9. The column  $aifa$  represents the angle of two optical axes. The last column  $class$  represents the classification of the calibrated focal lengths.
- In this experiment, radial distortion was first corrected. The first order coefficient  $k1$  and the second order coefficient  $k2$  of the radial distortion were  $-3.7791e-7$  and  $1.3596e-12$  respectively.
- From Figure 6.10, we find that the relationship between the focal length and the angle  $c$  are nearly exponential functions. In fact, by inspection of the figure, we can detect that when  $c$  is larger than  $1.5^\circ$ , the calibrated focal lengths are quite stable. The mean of the focal lengths falling into this interval is 627.6 pixels, which is very close to the "true" one. The standard deviation of the estimated focal lengths is about 6.5285 pixels, which is within 1.1% of the "true" focal length. Based on this fact, we classify the data into 3 classes. In class 1,  $c$  is greater than  $1.5^\circ$ , and the algorithm is running safely. The results are quite good. In Figure 6.10, we can easily find when the angle  $c$  is less than  $1.0^\circ$ , the calibrated results are not stable. The errors vary from 25~280 pixels. We designate this case as class 2. In Figure 6.10, the class 3 is obtained when  $c$  is between  $1.0^\circ$  and  $1.5^\circ$ . In this class, the algorithm works in a transitional interval, the calibrated results are not bad, but they are worse than class 1. It is better to avoid this case.

- A careful reader may find that the quadratic equation gives similar performance to the linear equations when two optical axes are nearly coplanar (class 2). The reason is that the current algorithm for the fundamental matrix estimation is not designed for special cases. In fact, coplanarity of the optical axes is a special case. As described in the previous sections, when two optical axes are coplanar, the entry at the third row and third column of the semi-calibrated fundamental matrix  $G_{33}$  is 0. However, in practical cases, we cannot ensure that when optical axes are near coplanar, this element is near zero. In fact, in most cases, this element is larger than what is expected. Hence, the third coefficient of the quadratic equation (5.2.4) is not near zero. It is even larger than the first two coefficients. Thus, the quadratic equation does not work well in these cases. Nevertheless, we conclude that the quadratic equation works marginally better than the two linear equations.

### III. Reconstruction results using the calibrated focal length

Having calibrated the focal length, we can estimate the relative position of the two considered images [14] and carry out a 3D reconstruction of the matched image points [10]. We did this for several image pairs. In order to evaluate the quality of the 3D reconstruction, we compare it to the known geometry of the calibration grid. We take two steps to achieve this objective. Firstly, we fit planes to the 3 subsets of coplanar points (cf. Figure 6.7). We decide on a relative distance to evaluate the coplanarity of points. We first measure the distances of points to the fitted plane. Then we measure the largest distance between pairs of the considered points. We next express in percent, the distances of the points to the plane, relative to that largest distance, the so-called relative distance. Secondly, we measure the angles between each pair of planes and

compare it to "ground truth": one of the grid's planes forms  $90^\circ$  angles with the two others, which themselves form a  $120^\circ$  angle.

The results of our evaluation are displayed in Table 6.3. They are shown for 5 pairs, which share one common image. Note that from the left to the right, the baseline decreases. Row  $f$  contains the calibrated focal lengths. The rows  $A_{xy}$  show the angles between pairs of planes. The rows  $Std_x$  show, for the 3 planes, the standard deviation of the relative distances as described above, which is useful to evaluate the coplanarity of points.

We observe that for the two image pairs with the largest baseline, the angles are all within  $0.3^\circ$  from their true values. With decreasing baseline, the errors increase, both for the angles and the coplanarity measure, although they still stay relatively small.

Table 6.3: Reconstruction results using calibrated focal length

	Ground truth	Pair 1	Pair 2	Pair 3	Pair 4	Pair 5
$f$	625.0	625.1	630.3	633.9	634.9	624.3
A12	90.0	90.26	89.94	91.01	90.94	89.84
A13	90.0	89.74	89.23	92.35	91.49	88.69
A23	120.0	119.73	119.76	120.48	120.66	118.17
Std1	0.0	0.000146	0.000162	0.000255	0.000321	0.000296
Std2	0.0	0.00037	0.000359	0.00029	0.000399	0.000251
Std3	0.0	0.000289	0.000325	0.000522	0.000558	0.000394

### 6.2.3 Calibration using arbitrary scenes

In the previous experiments, since we used a special calibration object, matching was not a serious problem. However, in real applications, matching plays an important role in calibration. This part of experiments covers the complete camera self-calibration procedures --- from corner detection to the fundamental matrix computation. From the calibration results, we can conclude that when matching and the fundamental matrix computation are carefully conducted, this calibration algorithm can give convincing results.

In all the following experiments, the techniques given in [43] were used to obtain correspondent points and fundamental matrix. The software developed on Linux is available at <http://www-sop.inria.fr/robotvis/personnel/zzhang/software.html>.

### I. Calibration with indoor scene

Indoor scenes are quite static and can be easily controlled. In the scene used for this experiment, three cups were placed together. The background was a white wall, so there were few depth cues. We could also move the camera close to the objects in order to capture enough features. In this experiment, the camera function setting used in the previous experiments was employed. Therefore, we can compare the calibrated results with those in the last section. We took in total four images of the three cups. They are shown in Figure 6.11. The calibrated results are presented in Table 6.4.

Table 6.4: Results calibrated from images containing 3 cups

Image pair	Ground truth	12	13	14*	23	24	34
f	625.0	599.2	605.3	584.3	617.7	607.8	624.6

\*: In this case, the  $c$  angle (see last section) is  $1.0114^\circ$ .





Figure 6.11: Images of three cups

Compared with the "ground truth", the maximum relative error is about 6.5%, and the average relative error is less than 5%. After the case of 14 that is near the coplanar case is excluded, the maximum relative error is about 4%, and the average relative error is less than 2.3%.

## II. Calibration with outdoor scene

We used the same camera setting to take 5 images of an outdoor scene. This is a building on the campus of the National University of Singapore. The distance between the camera and the building was about 25 meters. The calibrated results are presented in Table 6.5.

Table 6.5: Results calibrated from images containing a building

Image pair	Ground truth	12	14	15	23	25	34	35
f	625.0	638.4	655.4	597.6	697.8	685.1	589.3	664.3

We note the following points when analyzing the results in Table 6.4 and Table 6.5:

- The first row of Table 6.4 and Table 6.5 is the combination of images. For example, case 12 is the combination of the first and second images. In Table 6.5, camera configurations near the coplanar case are excluded.
- Although the same camera setting was used, the camera focused on objects at different distances, especially in the building case. Hence, relative errors in

the calibrated result were expected up to several percent. Here, the maximum relative error is about 10%, which seems reasonable for this experiment.

- In Figure 6.12, we find the building is of a large depth variety. Although large depth variety may be better for calibration, it may lead to serious problems for matching (A little displacement in an image means a great displacement in the 3D world). Hence large depth variety may be one factor that affects the calibrated results in Table 6.5.



Figure 6.12: some images of a building

### III. The reconstructed model from the arbitrary scenes

Like we did for the images of the calibration grid, we performed a 3D reconstruction of the indoor scene of 3 cups using the calibrated result. We first used the techniques described in [10] to recover the scene's structure. A triangular mesh was semi-automatically adjusted to the reconstructed 3D points, and used to create textured



VRML models. A few renderings of one of the models are shown in Figure 6.13. Due to the sparseness of the extracted interest points, the reconstruction of the scene is not complete of course. However, Figure 6.13 shows that it is qualitatively correct. The second row of the scene shows the coplanarity of the close-ups of the plug, and third row shows the cylindrical shape of the cups.

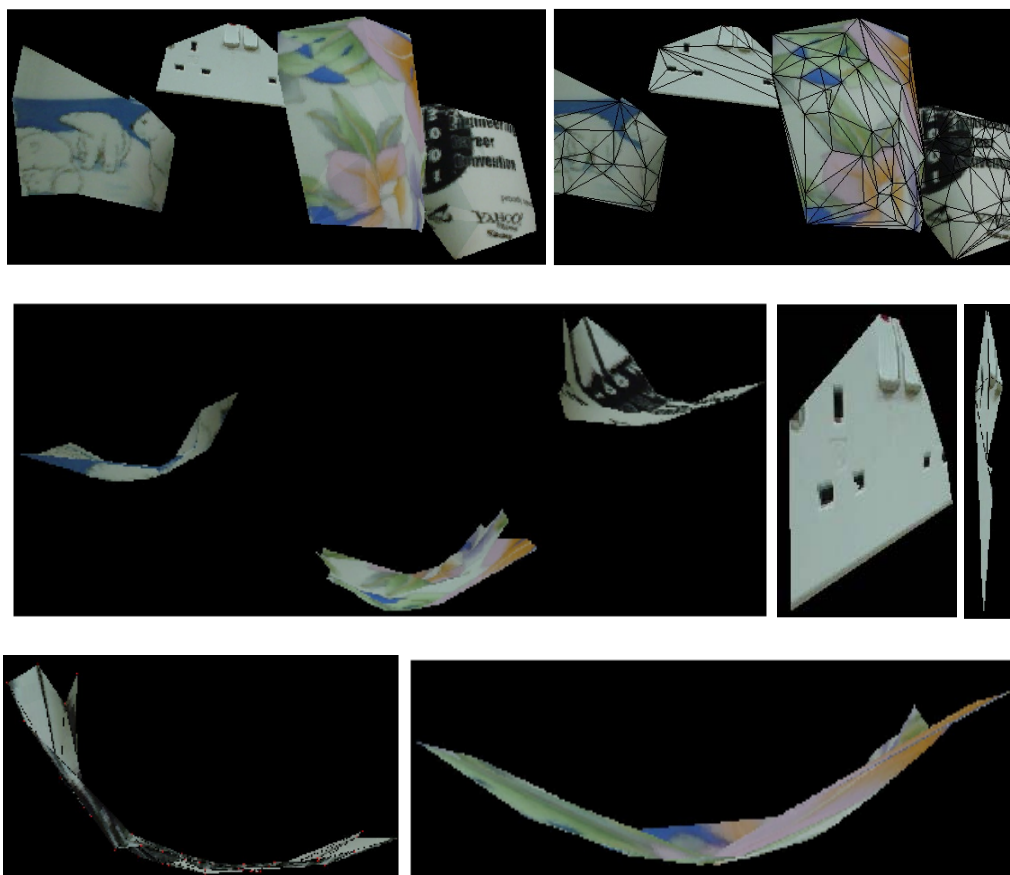


Figure 6.13: The reconstructed cup. First row: general appearance of the scene, once with overlaid triangular mesh. Second row: rough top view of cups and two close-ups of the plug in the background (rightmost image shows the near coplanarity of the reconstruction). Third row: top views of two of the cups, showing that their cylindrical shape has been recovered.

### 6.3 Conclusion

In the above experiments, first we showed that focal length calibration is nearly independent of principal point estimation. Experiments on the special calibration grids demonstrated that the self focal length calibration algorithm given here is robust. Cali-

bration results obtained from arbitrary scenes make certain that this algorithm can be used in many applications. Although we do not hope to use this algorithm to provide calibration results as accurate as those obtained by calibration methods involving calibration grids, the calibrated results are still convincing. We believe it will help to fill the gap in applications concerning automatic structure from motion.

## Chapter 7 Conclusion

The thesis presents a new approach for camera's focal length calibration. The approach assumes only the camera's focal length is unknown and constant. Thus the Kruppa's equations, which are popularly used to self-calibrate a camera are decomposed as two linear and one quadratic equations. Then the first advantage of these calibration equations is that they give closed form solutions.

A conventional wisdom in computer vision community is that Kruppa's equations based camera self-calibration is unstable. Based on Sturm's critical motion sequences analyses for camera self-calibration and structure recovery, we can give all of singular cases for our self-calibration algorithm. These singular cases that we call generic singular cases are nearly correspondent to all of algebraic degeneration of those equations. After excluding those singular cases, we find that our algorithm is really stable and easy to implement.

The work presented here is not near the end. Nonlinear estimation can be included to refine the algorithm in the future research. Focal length might not be necessary constant. If so, zoom and different focus can be employed when we calibrate a camera and thus it is more flexible. Of course, we shall not forget that 3D modeling rather than calibration is our final goal. Thus a complete system should be finally established. This system as we described in the Chapter 1, should include image feature extraction and matching, camera self-calibration, structure from motion and dense model reconstruction. After all of these components are embedded in the system, we can find the input of the system is a sequence of images and the output is the 3D model. The system can then be implanted in robots, media industry and so on.

## Reference

- [1] W. Boehm and H. Prautzsch. Geometric Concepts for Geometric Design. A K Peters, Wellesley, Massachusetts, 1994.
- [2] Z. L. Cheng, A. N. Poo and C. Y. Chen. A linear approach for online focal length calibration. In Proceeding of Model-based Imaging, Rendering, Image Analysis and Graphical Special Effects, pages 63-69, 2003.
- [3] I. J. Cox, S. L. Hingorani, and S. B. Rao. A maximum likelihood stereo algorithm. Computer Vision and Image Understanding, 63(3): 542-567, 1996.
- [4] L. Dron. Dynamic camera self-calibration from controlled motion sequences. In Proc. IEEE Conference on Computer Vision and Pattern Recognition, pages 501-506, 1993.
- [5] O. D. Faugeras. What can be seen in three dimensions with an uncalibrated stereo rig? In Proc. European Conference on Computer Vision, LNCS 588, pages 563-578. Springer-Verlag, 1992.
- [6] O. D. Faugeras, Q. Luong and S. Maybank. Camera self-calibration: Theory and experiments. In Proc. European Conference on Computer Vision, LNCS 588, pages 321-334, Springer-Verlag, 1992.
- [7] O. D. Faugeras. Three Dimensional Computer Vision: a Geometric Viewpoint. MIT Press, 1993.
- [8] C. Harris and M. Stephens, A combined corner and edge detector, Fourth Alvey Vision Conference, pp. 147-151, 1988.
- [9] R. I. Hartley. Estimation of relative camera positions for uncalibrated cameras. In Proc. European Conference on Computer Vision, LNCS 588, pages 579-587, Springer-Verlag, 1992.

- [10] R. I. Hartley and P. Sturm. Triangulation. In DARPA Image Understanding Workshop, Monterey, CA, pages 957-966, 1994.
- [11] R. I. Hartley. Self-calibration from multiple views with a rotating camera. In Proc. European Conference on Computer Vision, LNCS 800/801, pages 471-478. Springer-Verlag, 1994.
- [12] R. I. Hartley. In defense of the eight-point algorithm. IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(6):580-593, October 1997.
- [13] R. I. Hartley. Kruppa's equations derived from the fundamental matrix. IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(2):133-135, 1997.
- [14] R. I. Hartley and A. Zisserman. Multiple View Geometry in Computer Vision. Cambridge University Press, ISBN: 0521623049, page 237, 2000.
- [15] R. Koch. 3D surface reconstruction from stereoscopic image sequences. In Proc. 5<sup>th</sup> International Conference on Computer Vision, Boston, pages 109-114, 1995.
- [16] E. Kruppa. "Zur ermittlung eines objektes aus zwei perspektiven mit innerer orientierung", Sitz-Ber. Akad. Wiss., Wien, math. naturw. Abt. Iia, 122:1939-1948, 1913.
- [17] K. Kululakos and S. Seitz. A theory of shape by space carving. In Proc. 7<sup>th</sup> International Conference on Computer Vision, Kerkyra, Greece, pages 307-314, 1999.
- [18] D. Liebowitz. Camera Calibration and Reconstruction of Geometry from Images, PhD thesis, Dept. of Engineering Science, University of Oxford, 2001.
- [19] H.C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. Nature, 293:133-135, September 1981.
- [20] Q. T. Luong and T. Vieville. Canonical representations for the geometries of multiple projective views. Computer Vision and Image Understanding, 64(2): 193-229, September 1996.

- [21] Y. Ma, R. Vidal, J. Kosecka, and S. Sastry. Camera Self-Calibration: Renormalization and Degeneracy Resolution for Kruppa's Equations. In Proc. European Conference on Computer Vision (ECCV), Trinity College Dublin, Ireland, 2000.
- [22] D. Marr and T. Poggio. A Computational Theory of Human Stereo Vision, Proc. Royal Society of London, Vol. 204 of B, pp. 301-328, 1979.
- [23] M. Personnaz and P. Sturm. Calibration of a stereo-vision system by the non-linear optimization of the motion of a calibration object. Research report, Institut National de Recherche en Informatique et en Automatique, 2002.
- [24] M. Pollefeys, R. Koch, and L. Van Gool. Self calibration and metric reconstruction in spite of varying and unknown internal camera parameters. In Proc. 6<sup>th</sup> International Conference on Computer Vision, Bombay, India, pages 90-96, 1998.
- [25] P. J. Rousseeuw. Robust Regression and Outlier Detection. Wiley, New York, 1987.
- [26] P. D. Sampson. Fitting conic sections to 'very scattered' data: An iterative refinement of the Bookstein algorithm. Computer Vision, Graphics, and Image Processing, 18:97-108, 1982.
- [27] C. Schmid, R. Mohr and C. Bauckhage. Comparing and Evaluating Interest Points, Proc. International Conference on Computer Vision, Narosa Publishing House, pp. 230-235, 1998.
- [28] S. M. Seitz and C. R. Dyer. Photorealistic scene reconstruction by voxel coloring. In Proc. IEEE Conference on Computer Vision and Pattern Recognition, Puerto Rico, pages 1067-1073, 1997.
- [29] J. G. Semple and G. T. Kneebone. Algebraic Projective Geometry. Oxford University Press, 1979.

- [30] M.E. Spetsakis and J. Aloimonos. A multi-frame approach to visual motion perception. *International Journal of Computer Vision*, 16(3):245-255, 1991.
- [31] P. Sturm. Critical motion sequences for monocular self-calibration and uncalibrated Euclidean reconstruction. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, Puerto Rico, pages 1100-1105, June 1997.
- [32] P. Sturm. Critical motion sequences for the self-calibration of cameras and stereo systems with variable focal length. In *Proc. 10<sup>th</sup> British Machine Vision Conference*, Nottingham, pages 63-72, 1999.
- [33] P. Sturm. On focal length calibration from two views. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, Hawaii, pages 145-150, Vol. II, December 2001.
- [34] R. Szeliski and S. Kang, Recovering 3D shape and motion from image streams using non-linear least-squares, DEC technical report 93/3, DEC, 1993.
- [35] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: A factorization approach. *International Journal of Computer Vision*, 9(2):137-154, November 1992.
- [36] P. Torr. Motion Segmentation and Outlier Detection, PhD Thesis, Dept. of Engineering Science, University of Oxford, 1995.
- [37] P. Torr and A. Zisserman. Feature Based Methods for Structure and Motion Estimation. In *International Workshop on Vision Algorithms*, pages 278-295, 1999.
- [38] B. Triggs. Auto-calibration from planar scenes, *Computer Vision ---ECCV' 98*, vol. 1, Lecture Notes in Computer Science, Vol. 1406, Springer-Verlag, pages 89-105, 1998.
- [39] B. Triggs. Auto-calibration and the absolute quadric. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 609-614, 1997.

- [40] R. Y. Tsai and T. Huang, Uniqueness and Estimation of Three-Dimensional Motion Parameters of Rigid Objects with Curved Surfaces, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.6, pp.13-27, Jan. 1984.
- [41] R. Y. Tsai. A versatile camera calibration technique for high accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses, *IEEE Journal of Robotics and Automation*, Vol. 3, No. 4, pages 323-244, Aug. 1987.
- [42] J. Weng, P. Cohen and M. Herniou. Calibration of Stereo Cameras Using a Non-linear Distortion Model. In *Proc. IEEE International Conference on Pattern Recognition*, Atlantic City, New Jersey, pages 246-253, June 16-21, 1990.
- [43] Z. Zhang, R. Deriche, O. D. Faugeras, and Q. Luong. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. *Artificial Intelligence*, 78:87-119, 1995.
- [44] Z. Zhang. Determining the epipolar geometry and its uncertainty – a review. *International Journal of Computer Vision*, 27(2):161-195, March 1998.
- [45] C. Zeller. Projective, Affine and Euclidean calibration in computer vision and the application of three dimensional perception. PhD thesis, Robot Vis Group, INRIA, Sophia-Antipolis, 1996.
- [46] R. Hartley. Estimation of relative camera position for uncalibrated cameras. In *Proc. European Conference on Computer Vision*, pages 579-587, 1992.
- [47] S. Bougnoux. From projective to Euclidean space under any practical situation, a criticism of self-calibration. In *Proc. 6<sup>th</sup> International Conference on Computer Vision*, Bombay, India, pages 790-796, January, 1998.
- [48] A. Heyden and K. Astrom. Euclidean reconstruction from image sequences with varying and unknown focal length and principal point. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*. 1997.



[49] Z. L. Cheng, P. Sturm, A. N. Poo and C. Y. Chen. Focal length calibration from two views: method and analysis of singular cases. Technical report, National University of Singapore, 2003.

[50] M. Pollefeys. Self-calibration and metric 3D reconstruction from uncalibrated image sequences. PhD Thesis, Katholieke Universiteit Leuven, Belgium, 1999.

## Appendix A

### Orthogonal least squares problem

The problem initially originates from finding the solution of a homogeneous equation  $AX=0$ , where  $A$  is an  $m \times n$  matrix with  $m > n$ . Obviously the trivial solution  $X=0$  should be excluded. In order to find a nontrivial solution, the column rank of the matrix is less than  $n$ .

In practice, the matrix  $A$  is perturbed by noise. Thus  $A$  may be full column rank. The problem is then reinterpreted as

$$X = \arg \min_x \|AX\| \quad (1)$$

$$\text{subject to } \|X\|=1^1$$

Applying the Lagrangian multiplier for the minimization object function (1):

$$\min X^T A^T AX - \lambda X^T X \quad (2)$$

Set the first derivative of (2) to be zero. (2) is then:

$$A^T AX = \lambda X \quad (3)$$

The solution is then the eigenvector of the matrix  $A^T A$ .

Place (3) into (1), the solution is  $\lambda$ . Therefore the solution of the orthogonal linear least squares problem is the eigenvector associated with least eigenvalue of the matrix  $A^T A$ .

---

<sup>1</sup> Since multiplying a scalar to a homogeneous equation does not influence the final solution of the problem, it is reasonable to assume the Euclidean norm of the solution is 1.

## Appendix B<sup>1</sup>

### B.1 The equivalent form of the semi-calibrated fundamental matrix

Before the discussion of the coplanar and noncoplanar singular cases, a special parameterization of the relative camera pose should be firstly adopted. Without loss of generality, the first camera is assumed in canonical position, besides that it is rotated about its optical axis (the Z-axis of the reference frame) by a rotation  $R_{Z,1}$ . The optical center of the second camera can then be assumed to lie e.g. in the plane  $X = 0$ , i.e. its coordinates are  $(0, Y, Z)$ . Furthermore, its orientation is given via three basic rotation matrices:  $R_2 = R_{Z,2}R_YR_X$ . It can be shown that the fundamental matrix is then given by:

$$G \sim \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & f \end{pmatrix} R_{Z,2} R_Y R_X \begin{bmatrix} 0 \\ Y \\ Z \end{bmatrix}_x R_{Z,1} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & f \end{pmatrix} \quad (1)$$

Importantly, due to the special form of  $R_Z$ , the matrix  $G$  can be rewritten as:

$$G \sim R_{Z,2} \underbrace{\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & f \end{pmatrix} R_Y R_X \begin{bmatrix} 0 \\ Y \\ Z \end{bmatrix}_x \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & f \end{pmatrix}}_H R_{Z,1}. \quad (2)$$

Due to the special form of  $R_Z$  and the orthogonality of the left and right singular matrices of an SVD, it can be shown that  $G$  and  $H$  have the same singular values, and that the third rows of their respective singular matrices are equal. Specifically, this means that the SVDs of  $G$  and  $H$  lead to the same values for  $a$ ,  $b$ ,  $u_{13}$ ,  $u_{23}$ ,  $v_{13}$  and  $v_{23}$ , and thus to the same calibration equations.

---

<sup>1</sup> The contents in this appendix appear in a pending paper:

Hence, the matrix of the following form

$$H \sim \begin{pmatrix} (Z \sin \alpha - Y \cos \alpha) \sin \beta & -Z \cos \beta & fY \cos \beta \\ Z \cos \alpha + Y \sin \alpha & 0 & 0 \\ f(Z \sin \alpha - Y \cos \alpha) \cos \beta & fZ \sin \beta & -f^2 Y \sin \beta \end{pmatrix} \quad (3)$$

is equivalent to the matrix  $G$  in terms of the calibration equations in this paper. Here  $\alpha$  and  $\beta$  are the angles of the  $X$  and  $Y$  rotations.

Note that the directions of the optical axes of the two cameras are given by:

$$D_1 \sim \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \text{ and } D_2 \sim \begin{pmatrix} -\sin \beta \\ \sin \alpha \cos \beta \\ \cos \alpha \cos \beta \end{pmatrix}$$

## B.2 Coplanar optical axes

Follow the equation (3), coplanar optical axes means  $Y = 0$  or  $\sin \beta = 0$ .

**First case:**  $Y = 0$ . This means that the second optical center lies on the optical axis of the first camera (since both  $X$  and  $Y$  are equal to 0). In this case, the first epipole has coordinates  $(0 \ 0 \ 1)^T$ . However, the first epipole is equal to the third column  $v_3$  of the matrix  $V$  in the SVD of  $H$ . Due to the orthogonality of  $V$ , this implies that  $v_{13} = v_{23} = 0$ . Hence the quadratic equation becomes linear (neglecting the trivial solution  $f = 0$ ):

$$f^2 (a^2 (1 - u_{13}^2) - b^2 (1 - u_{23}^2)) + (a^2 u_{13}^2 - b^2 u_{23}^2) = 0. \quad (4)$$

One necessary condition for the quadratic equation to vanish is  $a = b$ . It can be shown that this happens exactly if  $\sin \alpha = \sin \beta = 0$ . This however means nothing else than that the optical axes are parallel to each other. Hence, there is no non-generic singularity for the quadratic equation in this case.

For the linear equations, it can be shown that the matrix  $HH^T$  has the following eigenvector with a non-zero eigenvalue:

$$\begin{pmatrix} \sin \alpha \\ \cos \alpha \sin \beta \\ 0 \end{pmatrix}.$$

However, this eigenvector is equal to either  $\mathbf{u}_1$  or  $\mathbf{u}_2$ . Hence, either  $u_{13} = 0$  or  $u_{23} = 0$ .

Combining the condition  $v_{13} = v_{23} = 0$ , the linear equations vanish.

**Second case:**  $\sin \beta = 0$ . In this case,  $H^T H$  and  $HH^T$  have  $(1\ 0\ 0)^T$  as an eigenvector with non-zero eigenvalue. Hence, one of the first two columns of  $U$  and one of first two rows of  $V^T$  have this form. However, the column and row indices must be different (otherwise, the  $(1, 1)$  or  $(2, 2)$  element of  $H$  may not be zero). This means that either  $u_{13} = v_{23} = 0$  or  $u_{23} = v_{13} = 0$ , which implies that the linear equations vanish and that the quadratic one becomes linear:

$$f^2(a^2(1-u_{13}^2)-b^2(1-v_{23}^2))+(a^2u_{13}^2-b^2v_{23}^2)=0 \quad (5)$$

if  $u_{23} = v_{13} = 0$  or

$$f^2(a^2(1-v_{13}^2)-b^2(1-u_{23}^2))+(a^2v_{13}^2-b^2u_{23}^2)=0 \quad (6)$$

if  $u_{13} = v_{23} = 0$ .

The equation (5) vanishes when  $a = b$  and  $u_{13}^2 = v_{23}^2$ . In the same principle, the equation (6) vanishes when  $a = b$  and  $u_{23}^2 = v_{13}^2$ . It can be shown that the three eigenvalues of  $HH^T$  are

$$\lambda_1 = 0, \quad \lambda_2 = f^2 Y^2 + Z^2 \text{ and } \lambda_3 = (Z \cos \alpha + Y \sin \alpha)^2 + (Z \sin \alpha - Y \cos \alpha)^2$$

Hence  $a = b$  exactly if

$$(Z \cos \alpha + Y \sin \alpha)^2 = Z^2 \quad (7)$$

$$\text{and } (Z \sin \alpha - Y \cos \alpha)^2 = Y^2 \quad (8)$$

---

<sup>1</sup> We only consider the case, which is independent of the focal length.

However, (7) and (8) are equivalent since both of them mean the two camera centers are equidistant from the intersection of two optical axes. Specifically, a point on the second optical axis is given by:

$$\begin{pmatrix} 0 \\ Y \\ Z \\ 1 \end{pmatrix} + \lambda \begin{pmatrix} 0 \\ \sin \alpha \\ \cos \alpha \\ 0 \end{pmatrix} \quad (9)$$

For non-parallel optical axes ( $\sin \alpha \neq 0$ ), we obtain the intersection point of the optical axes for  $\lambda = -Y / \sin \alpha$ :

$$Q = \begin{pmatrix} 0 \\ 0 \\ Z - Y \frac{\cos \alpha}{\sin \alpha} \\ 1 \end{pmatrix} \quad (10)$$

It is easy to verify that both of (7) and (8) make the two camera centers are equidistant from  $Q$ .

After (7) and (8) are applied to  $H$ , it is easy to find

$$U = \begin{pmatrix} 0 & 1 & 0 \\ Z/d_1 f Y & 0 & -f Y/d_2 Z \\ 1/d_1 & 0 & 1/d_2 \end{pmatrix} \text{ and } V = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -Z/d_1 f Y & f Y/d_2 Z \\ 0 & 1/d_1 & 1/d_2 \end{pmatrix}$$

if  $u_{23} = v_{13} = 0$

$$\text{or } U = \begin{pmatrix} 1 & 0 & 0 \\ 0 & Z/d_1 f Y & -f Y/d_2 Z \\ 0 & 1/d_1 & 1/d_2 \end{pmatrix} \text{ and } V = \begin{pmatrix} 0 & 1 & 0 \\ -Z/d_1 f Y & 0 & f Y/d_2 Z \\ 1/d_1 & 0 & 1/d_2 \end{pmatrix}$$

if  $u_{13} = v_{23} = 0$ , where  $d_1 = \sqrt{(Z/fY)^2 + 1}$  and  $d_2 = \sqrt{(fY/Z)^2 + 1}$

Then we can immediately find in the former case  $u_{13}^2 = v_{23}^2$  and the latter case  $u_{23}^2 = v_{13}^2$ .

That means the equidistance is equivalent to the degeneration of (5) and (6) and hence the quadratic equation.

**Summary.** Whenever the optical axes are coplanar, the two linear equations (in chapter 5) vanish and the quadratic equation (in chapter 5) becomes linear. The latter vanishes exactly if the optical axes are parallel or if the optical centers are equidistant from the intersection of the optical axes. Hence all of singular cases of the quadratic and linear equations in coplanar case are generic singular cases, i.e. the equivalent algebraically singular cases.

## B.3 Non-coplanar optical axes

### B.3.1 Linear equations

As Chapter 5 shows, for linear equations, the non-coplanar singular cases are:

$$u_{23} = v_{23} = 0$$

$$u_{13} = v_{13} = 0$$

**First case:**  $u_{23} = v_{23} = 0$ . In the following, the SVD of  $H$  is considered. As described in section B.1, the first epipole  $v_3$  is  $(0, fY, Z)^T$ . Hence we have:

$$H \sim \begin{pmatrix} (Z \sin \alpha - Y \cos \alpha) \sin \beta & -Z \cos \beta & fY \cos \beta \\ Z \cos \alpha + Y \sin \alpha & 0 & 0 \\ f(Z \sin \alpha - Y \cos \alpha) \cos \beta & fZ \sin \beta & -f^2 Y \sin \beta \end{pmatrix} \quad (11)$$

$$\stackrel{SVD}{\sim} \underbrace{\begin{pmatrix} u_{11} & u_{21} & u_{31} \\ u_{12} & u_{22} & u_{32} \\ u_{13} & 0 & u_{33} \end{pmatrix}}_U \begin{pmatrix} a & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & 0 \end{pmatrix} \underbrace{\begin{pmatrix} v_{11} & v_{12} & v_{13} \\ v_{21} & v_{22} & 0 \\ 0 & fY & Z \end{pmatrix}}_{V^T}$$

From the orthogonality of rows 2 and 3 of  $V^T$ , it follows that  $v_{22} = 0$  and from this, that  $v_{11} = 0$ . From  $H_{22} = H_{23} = 0$ , it also follows that  $u_{12} = 0$ . Hence (11) is rewritten as:

---

<sup>1</sup> Here unitary determine of the orthogonal matrix is not imposed.

$$\begin{pmatrix} (Z \sin \alpha - Y \cos \alpha) \sin \beta & -Z \cos \beta & fY \cos \beta \\ Z \cos \alpha + Y \sin \alpha & 0 & 0 \\ f(Z \sin \alpha - Y \cos \alpha) \cos \beta & fZ \sin \beta & -f^2 Y \sin \beta \end{pmatrix} \sim \begin{pmatrix} u_{11} & u_{21} & u_{31} \\ 0 & u_{22} & u_{32} \\ u_{13} & 0 & u_{33} \end{pmatrix} \begin{pmatrix} a & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & v_{12} & v_{13} \\ v_{21} & 0 & 0 \\ 0 & fY & Z \end{pmatrix}. \quad (12)$$

The right hand side of the equation determines the (3, 1) element of the left hand side to be zero. Thus

$$Z \sin \alpha - Y \cos \alpha = 0 \quad \text{or} \quad \cos \beta = 0$$

is the necessary condition for non-coplanar singular cases for the linear equations in the first case.

If  $\cos \beta = 0$ , then  $\sin \beta = \pm 1$ .  $H$  becomes:

$$\begin{pmatrix} Z \sin \alpha - Y \cos \alpha & 0 & 0 \\ Z \cos \alpha + Y \sin \alpha & 0 & 0 \\ 0 & fZ - f^2 Y \end{pmatrix} \sim \begin{pmatrix} 0 & Z \sin \alpha - Y \cos \alpha & Z \cos \alpha + Y \sin \alpha \\ 0 & Z \cos \alpha + Y \sin \alpha & Y \cos \alpha - Z \sin \alpha \\ t_1 & 0 & 0 \end{pmatrix} \begin{pmatrix} ft_2 & 0 & 0 \\ 0 & t_1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & Z & -fY \\ t_2 & 0 & 0 \\ 0 & fY & Z \end{pmatrix} \quad (13)$$

or

$$\begin{pmatrix} -Z \sin \alpha + Y \cos \alpha & 0 & 0 \\ Z \cos \alpha + Y \sin \alpha & 0 & 0 \\ 0 & -fZ & f^2 Y \end{pmatrix} \sim \begin{pmatrix} 0 & -Z \sin \alpha + Y \cos \alpha & -Z \cos \alpha - Y \sin \alpha \\ 0 & Z \cos \alpha + Y \sin \alpha & Y \cos \alpha - Z \sin \alpha \\ t_1 & 0 & 0 \end{pmatrix} \begin{pmatrix} ft_2 & 0 & 0 \\ 0 & t_1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & -Z & fY \\ t_2 & 0 & 0 \\ 0 & fY & Z \end{pmatrix} \quad (14)$$

where  $t_1 = \sqrt{Y^2 + Z^2}$  and  $t_2 = \sqrt{f^2 Y^2 + Z^2}$ . It is easy to verify (13) and (14) really satisfy the condition of SVD. Thus we find a SVD for the fundamental matrix with  $u_{23} = v_{23} = 0$  when  $\cos \beta = 0$ . The geometrical configuration of this case is the second optical axis points in the X direction, i.e. the normal direction of the plane spanned by the two optical centers and the first optical axis.

If  $Z \sin \alpha - Y \cos \alpha = 0$ , since  $Y \neq 0$  (otherwise the optical axes are coplanar), hence  $\sin \alpha \neq 0$  and the condition is then:

$$Z = \frac{\cos \alpha}{\sin \alpha} Y.$$

$H$  then becomes:



$$\begin{aligned}
H &\sim \begin{pmatrix} 0 & -Z \cos \beta & fY \cos \beta \\ Z \cos \alpha + Y \sin \alpha & 0 & 0 \\ 0 & fZ \sin \beta & -f^2 Y \sin \beta \end{pmatrix} \\
&\sim \begin{pmatrix} 0 - \cos \alpha \cos \beta & f \sin \alpha \cos \beta \\ 1 & 0 & 0 \\ 0 & f \cos \alpha \sin \beta & -f^2 \sin \alpha \sin \beta \end{pmatrix} \quad (15)
\end{aligned}$$

An SVD for this is (possibly up to ordering the singular values):

$$\begin{pmatrix} \cos \beta & 0 & f \sin \beta \\ 0 & t_2 & 0 \\ -f \sin \beta & 0 & \cos \beta \end{pmatrix} \begin{pmatrix} t_1 t_2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & -\cos \alpha & f \sin \alpha \\ t_1 & 0 & 0 \\ 0 & f \sin \alpha & \cos \alpha \end{pmatrix} \quad (16)$$

with  $t_1 = \sqrt{f^2 \sin^2 \alpha + \cos^2 \alpha}$  and  $t_2 = \sqrt{f^2 \sin^2 \beta + \cos^2 \beta}$ . Hence, there is also an SVD of  $H$  that satisfies  $u_{23} = v_{23} = 0$  when  $Z \sin \alpha - Y \cos \alpha = 0$ . The geometrical interpretation of  $Z \sin \alpha - Y \cos \alpha = 0$  is as follows: the second optical axis lies in the plane orthogonal to the plane spanned by the optical centers and the first optical axis and containing the baseline. Of course this case is of little practical importance.

**Second case:**  $u_{13} = v_{13} = 0$ . The analysis can be done analogously as above, leading to the same conclusions (the SVDs are the same, up to swapping of the singular values and corresponding columns of  $U$  and  $V$ ). Which one of the cases  $u_{23} = v_{23} = 0$  or  $u_{13} = v_{13} = 0$  occurs in practice, depends on which one of the singular values is larger.

### B.3.2 Quadratic equation

The conditions in the case of non-coplanar optical axes are:

$$a = b, \quad u_{23} = \pm v_{13} \quad \text{and} \quad v_{23} = \pm u_{13}$$

$$a = b, \quad u_{23} = \pm v_{13} \quad \text{and} \quad v_{23} = \pm u_{13} .$$

**First case:**  $a = b$ ,  $u_{23} = \pm v_{13}$  **and**  $v_{23} = \pm u_{13}$ . Like the disposition in the linear case,  $H$  in this case is:

$$H \stackrel{SVD}{\sim} \begin{pmatrix} u_{11} & u_{21} & u_{31} \\ u_{12} & u_{22} & u_{32} \\ u_{13} \pm u_{13} & u_{33} & \end{pmatrix} \begin{pmatrix} a & 0 & 0 \\ 0 & a & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} v_{11} & v_{12} & v_{13} \\ v_{21} & v_{22} & \pm v_{13} \\ 0 & fY & Z \end{pmatrix} \quad (17)$$

Due to the orthogonality of columns of  $V^T$ , there are scalars  $\lambda$  and  $\mu$  with:

$$\begin{pmatrix} v_{11} \\ v_{21} \\ 0 \end{pmatrix} = \begin{pmatrix} \lambda v_{13} \\ \mp \lambda v_{13} \\ 0 \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} v_{12} \\ v_{22} \\ 0 \end{pmatrix} = \begin{pmatrix} \mu v_{13} \\ \mu v_{23} \\ 0 \end{pmatrix}$$

Then the symmetric matrix  $X = H^T H$  is thus given by:

$$X \sim \begin{pmatrix} \lambda^2 & 0 & 0 \\ 0 & \mu^2 & \mu \\ 0 & \mu & 1 \end{pmatrix} \quad (18)$$

Compared (18) with (3), there follows two sets of equations:

$$Z \cos \beta \sin \beta (Y \cos \alpha - Z \sin \alpha + f^2 Z \sin \alpha - f^2 Y \cos \alpha) = 0$$

$$fY \cos \beta \sin \beta (Z \sin \alpha - Y \cos \alpha - f^2 Z \sin \alpha + f^2 Y \cos \alpha) = 0$$

or equivalently:

$$Z \cos \beta \sin \beta (f^2 - 1)(Z \sin \alpha - Y \cos \alpha) = 0 \quad (19)$$

$$-fY \cos \beta \sin \beta (f^2 - 1)(Z \sin \alpha - Y \cos \alpha) = 0 \quad (20)$$

Excluded the trivial cases  $f^2 = 1$  and  $Z = 0$  and coplanar case  $Y = 0$  and  $\sin \beta = 0$ , the above two equations imply  $\cos \beta = 0$  or  $Z \sin \alpha = Y \cos \alpha$ .

With  $\cos \beta = 0$ , the eigenvalues of  $H^T H$  can be computed to be:

$$\lambda_1 = 0, \quad \lambda_2 = Y^2 + Z^2 \quad \text{and} \quad \lambda_3 = f^2 (f^2 Y^2 + Z^2)$$

Then the condition of the identical eigenvalues gives the two trivial solutions for  $f$ :

$$f^2 = 1 \quad \text{or} \quad f^2 = -\frac{Y^2 + Z^2}{Y^2}$$

Hence there is no geometrical configuration correspondent to the case of  $\cos \beta = 0$ .

Consider now the case  $Z \sin \alpha = Y \cos \alpha$ . Following the same scheme in section B.3.1, the matrix  $H$  becomes

$$H \sim \begin{pmatrix} \cos \beta & 0 & f \sin \beta \\ 0 & t_2 & 0 \\ -f \sin \beta & 0 & \cos \beta \end{pmatrix} \begin{pmatrix} t_1 t_2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & -\cos \alpha & f \sin \alpha \\ t_1 & 0 & 0 \\ 0 & f \sin \alpha & \cos \alpha \end{pmatrix} \quad (21)$$

with  $t_1 = \sqrt{f^2 \sin^2 \alpha + \cos^2 \alpha}$  and  $t_2 = \sqrt{f^2 \sin^2 \beta + \cos^2 \beta}$ .

Applied the initial conditions  $a = b$ ,  $u_{23} = \pm v_{13}$  and  $v_{23} = \pm u_{13}$ , (21) implies:

$$t_1 t_2 = 1$$

$$f \sin \beta = 0$$

$$f \sin \alpha = 0$$

This exactly means  $\sin \alpha = \sin \beta = 0$ , i.e. the two optical axes are parallel and thus the optical axes are coplanar.

**Second case:**  $a = b$ ,  $u_{23} = \pm v_{13}$  and  $v_{23} = \pm u_{13}$ . As the same scheme in the last subsection, there are the same constrains as the equations (19) and (20) and then the same conclusions can be obtained.

**Summary:** There is no singular case for the quadratic equation when the optical axes are not coplanar.