Auto-Calibration and Three-Dimensional Reconstruction for Zooming Cameras

by

Tarik A. Elamsy

A Dissertation
Submitted to the Faculty of Graduate Studies
through the School of Computer Science
in Partial Fulfillment of the Requirements for
the Degree of Doctor of Philosophy at the
University of Windsor

Windsor, Ontario, Canada
2014

Auto-Calibration and Three-Dimensional Reconstruction for Zooming Cameras

by

Tarik A. Elamsy

APPROVED BY:

Dr. W. Lee, External Examiner
Faculty of Engineering, University of Ottawa

Dr. R. Barron
Department of Mathematics and Statistics

Dr. R. Gras
School of Computer Science

Dr. I. Ahmad
School of Computer Science

Dr. B. Boufama, Advisor
School of Computer Science

September 12, 2014

Declaration of Co-Authorship / Previous Publication

I. Co-Authorship Declaration

I hereby declare that this dissertation incorporates material that is the result of joint research, as follows: The entire dissertation and all papers listed below in part II were written with the guidance of my supervisor Dr. Boubakeur Boufama, who provided valuable feedback and editorial input during the writing process. In addition, the three papers listed below in part II were written in collaboration with Dr. Adlan Habed.

I am aware of the University of Windsor Senate Policy on Authorship and I certify that I have properly acknowledged the contribution of other researchers to my dissertation, and have obtained written permission from each of the co-author(s) to include the above material(s) in my dissertation.

I certify that, with the above qualification, this dissertation, and the research to which it refers, is the product of my own work.

II. Declaration of Previous Publications

This dissertation includes parts of 3 papers that have been previously published in peer reviewed journals and conference proceedings. These papers are totally from my own work during my PhD studies in terms of concepts, design, implementations, and writing. In addition to what is mentioned in part I, my supervisor helped me to choose what research problems I have to work on, evaluating and giving his feedbacks about my outcomes, reviewing my papers and dissertation. The above mentioned papers are as follows:

1. T. Elamsy, B. Boufama, and A Habed. Parallel planes identification using uncalibrated zooming cameras. In *Proceedings of the International Conference on Computer and Robot Vision, CRV'13*, pages 174–180, 2013.
2. T. Elamsy, A. Habed, and B. Boufama. A new method for linear affine self-calibration of stationary zooming stereo cameras. In *Proceeding of 19th IEEE International Conference on Image Processing, ICIP'12*, pages 353-356, 2012.
3. T. Elamsy, A. Habed, and B. Boufama. Self-calibration of stationary non-rotating zooming cameras. *Image and Vision Computing*, 32(3):212–226, 2014.

I certify that I have obtained a written permission from the copyright owner(s) to include the above published material(s) in my dissertation. I certify that the above material describes work completed during my registration as a graduate student at the University of Windsor.

I declare that, to the best of my knowledge, my dissertation does not infringe upon anyone's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my dissertation, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained written permission from the copyright owner(s) to include such material(s) in my dissertation.

I declare that this is a true copy of my dissertation, including any final revisions, as approved by my dissertation committee and the Graduate Studies office, and that this dissertation has not been submitted for a higher degree to any other University or Institution.

Abstract

This dissertation proposes new algorithms to recover the calibration parameters and 3D structure of a scene, using 2D images taken by uncalibrated stationary zooming cameras. This is a common configuration, usually encountered in surveillance camera networks, stereo camera systems, and event monitoring vision systems. This problem is known as camera auto-calibration (also called self-calibration) and the motivation behind this work is to obtain the Euclidean three-dimensional reconstruction and metric measurements of the scene, using only the captured images.

Under this configuration, the problem of auto-calibrating zooming cameras differs from the classical auto-calibration problem of a moving camera in two major aspects. First, the camera intrinsic parameters are changing due to zooming. Second, because cameras are stationary in our case, using classical motion constraints, such as a pure translation for example, is not possible.

In order to simplify the non-linear complexity of this problem, i.e., auto-calibration of zooming cameras, we have followed a geometric stratification approach. In particular, we have taken advantage of the movement of the camera center, that results from the zooming process, to locate the plane at infinity and, consequently to obtain an affine reconstruction. Then, using the assumption that typical cameras have rectangular or square pixels, the calculation of the camera intrinsic parameters have become possible, leading to the recovery of the Euclidean 3D structure. Being linear, the proposed algorithms were easily extended to the case of an arbitrary number of images and cameras. Furthermore, we have devised a sufficient constraint for detecting scene parallel planes, a useful information for solving other computer vision problems.

Dedication

To my beloved wife Rana and sons....

Acknowledgement

Though only my name is printed on the title page of this dissertation, this work would not have been possible without the help of many individuals who in one way or another contributed in the completion of this work.

First and foremost, I would like to express my heart-felt gratitude to my family. For my parents who planted the love of science in my heart and supported me in all my pursuits. And most of all for my loving, supportive, encouraging, and patient wife Rana to whom this dissertation is dedicated to. Her faithful support during the years of this Ph.D. is so appreciated.

My deepest and utmost gratitude is to my advisor **Dr. Boubakeur** whom it has been an honor for me to be his Ph.D. student. He did not save any effort in guiding and helping me throughout the years of my studies. He gave me the freedom to explore on my own, but at the same time never saved his guidance when my steps faltered. I appreciate all his contributions of time, advices, ideas, and funding to make my years of Ph.D. study productive and stimulating. The discussions that I had with Dr. Boufama helped me developing many ideas in this dissertation. However, what I learned from him is way more beyond just scientific. That is why I cannot thank you enough.

I also would like to gratefully thank and acknowledge **Dr. Adlan** for his valuable advices and support. I am deeply obliged to him for the many hours of discussions and for helping me to sort out many technical details of my work.

I am especially indebted to my dissertation committee members Dr. Lee, Dr.

Ronald, Dr. Gras, and Dr.Imran. They have been very gracious and generous with their time, ideas, and advices which provided me with deeper insight and forced me to think wider. Thanks so much to all of them.

I would like to also thank the various members of the CVMS lab for their constant support during all the ups and downs of my study years. Thank you all.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

The world has witnessed immense spread of inexpensive, yet high quality, digital imaging devices which are available almost everywhere (still cameras, cell phones, web-cams, tablets, cars, video cameras, security cameras, etc. . .). The massive spread and wide deployment of these imaging devices ignited the skyrocketing rate at which digital images and videos are taken by consumers, and consequently raised more demands and interest for computerized image interpretation. Despite the simplicity of the image acquisition process, a massive amount of information from the surrounding three-dimensional (3D) world is compactly stored into a small hand-size two-dimensional (2D) photo. As visual perception is almost effortless for humans, we are capable of recognizing objects and unfold the three-dimensional world information back from those two-dimensional images. However, when it comes to computer vision machines, such ability is so complex and difficult to imitate. In order to simplify this complexity, computer vision tasks are classified into more restricted domains each with limited and clear goals to be achieved, for example, recognition and three-dimensional structure recovery. In this dissertation, we target the central problem of three-dimensional reconstruction from two-dimensional images.

1.1 The 3D reconstruction problem

Successful solutions for many computer vision tasks may be achieved from 2D image information such as handwriting recognition, Optical Character Recognition (OCR), automatic face recognition, automated medical image analysis, etc. On the other hand, many computer vision applications mandate for their operation reconstructing the rigid 3D scene and information back from two or more images. This is known as the 3D reconstruction problem. Traditionally, the typical solutions are the usage of expensive devices working under controlled circumstances which were mainly used for robotics and inspection applications.

Nowadays, however, a growing demand for 3D information has emerged for applications such as navigation, surveillance, 3D modeling, archeology, visualization, and scene measurements with different levels of required accuracy and 3D data quality. In addition, these applications require camera systems with flexible acquisition procedures using low cost off-the-shelf cameras. Under these requirements, the acquisition of the 3D information of the scene is a more difficult task, as it is often assumed that both the scene and the camera geometry are unknown.

To simplify this complex problem, the framework of 3D reconstruction task is decomposed into a number of manageable subproblems, with clearly defined input and output links between them. This simplifies the problem by allowing researchers to focus on solving more specific subproblems.

The first subproblem in this context is the *feature matching problem*, which is also known as the correspondence problem. This problem concerns relating images by finding the corresponding features between the different images (e.g. points, lines , edges, etc). The result from the feature matching step initiates the next step of *structure from motion* problem. When given two or more related images of a rigid scene, taken by a moving camera with unknown motion and orientation, the 3D structure need to be computed. The simplest case of 3D reconstruction is when the camera parameters are known, i.e. calibrated camera, where only the position and orientation of the camera need to be obtained. Traditionally, the camera can be calibrated in advance, prior to image acquisition, using classical calibration techniques with the aid of special calibration patterns of known geometry. However, in this approach, the camera parameters should be kept constant during image acquisitions, and thus focusing and zooming are prohibited.

However, in the case of unknown camera parameters, i.e. uncalibrated camera, only projective structure of the scene can be obtained. Unfortunately, the projective structure is of very limited use for computer vision, and need to be upgraded to more useful and specialized structure such as affine, metric, or Euclidean. In order to upgrade a projective representation to a metric one, the intrinsic parameters must be recovered. Here comes the importance of auto-calibration, or self-calibration, which is the process of determining the intrinsic parameters (i.e. geometric and optical specifications) of the camera, from point correspondence only and without the aid of calibration pattern. When those parameters are recovered, the camera is said to be "calibrated" [43].

1.2 Zooming and auto-calibration

This dissertation is concerned mainly with the problem of recovering the scene structure from uncalibrated systems consisting of zooming cameras. The low cost of manufacturing high-resolution camera systems, with automated zoom lenses, has widely expanded their deployments. Camera systems with zoom ability are inherently more useful than those cameras with fixed lenses. For some vision tasks, it might be very useful to zoom out to yield a broad overview of a large area, while in other cases it might be very important to zoom in to take a closer look at an object. In general, the flexibility to freely adjust the camera settings to the scene's conditions allows producing better image quality. Camera systems with fixed parameters fail to produce meaningful data in many situations.

Despite these advantageous characteristics, zooming cameras are less commonly used in computer vision tasks, in comparison with imaging systems which rely on cameras with fixed parameters. Very little work has been done in the field of zooming camera auto-calibration. Using zooming cameras in computer vision introduce a wide range of visual processing difficulties, and only few studies have reported integrating image systems with zooming capabilities. The most important and obvious reason is that the camera's intrinsic parameters are immediately lost when the camera changes its setting by zooming. Knowledge of camera's intrinsic parameters are crucial for recovering the metric structure and metric measurements from the 2D images.

In this dissertation, we address the auto-calibration problem of a system of two or more individual stationary zooming cameras. To the best of our knowledge, such configuration of cameras has not been specifically addressed in the literature. This is a commonly occurring configuration often encountered in stereo camera systems mounted over a robot head, surveillance networks, and monitoring of events. In such image capture systems, each camera is physically attached to a static structure (wall, ceiling or tripod) and is only allowed to zoom. This is a challenging configuration in which only complex non-linear solutions exist. As the cameras may frequently zoom and thus mandate re-calibration, a simple and reliable solution is highly desirable.

A stratified approach for auto-calibrating such system of zooming cameras is proposed. The proposed camera auto-calibration method exploits the zooming capability of the cameras in order to directly estimate the location of the plane at infinity. To the best of our knowledge, this is the only method that does so without any assumption, such as a restricted camera motion or scene knowledge. The well-known modulus constraint, used to locate the plane at infinity, is only valid for cameras with constant intrinsic parameters. Furthermore, as its computation is nonlinear, it is not reliable to be used in practice. Because we are considering cameras that are stationary, methods based on restricted camera motion are not valid (e.g. pure rotation, pure translation, and planar motion). Moreover, the proposed method does not require the existence, and hence identification, of any scene constraints, such as, parallel and/or orthogonal lines or planes.

1.3 Objectives

The main objective of this work is to auto-calibrate a vision system, that consists of multiple zooming cameras, and to reconstruct the three-dimensional structure of a scene from two-dimensional images. In particular, the orientation and relative position of the cameras does not need to be known in advance.

This dissertation addresses the following problems in the context of zooming cameras :

- The calculation of the plane at infinity using only linear equations.
- The affine calibration of cameras and the affine 3D reconstruction of an observed scene.

- The metric calibration and metric 3D reconstruction of a scene.

- The automatic detection of the scene parallel planes.

1.4 Contribution

The main contributions of this dissertation are as follows:

- *Affine auto-calibration for a zooming stereo vision system.* A new linear method to compute the affine 3D structure from a stereo zooming camera system has been proposed. Based on the valid observation that, the principal planes before and after zooming provide a pair of parallel planes, we were able to extract constraints on the plane at infinity. Two such pairs of parallel planes, from the stereo pair of cameras, are enough to identify the plane at infinity and, thus allow to upgrade the projective structure to affine. The practical side of this method is that, unlike all other existing approaches, it does not rely on restricted intrinsic camera parameters, nor depends on special camera motions or scene constraints. This work was published in the paper [20].

- *Auto-calibration and 3D reconstruction using a set of zooming cameras.* A stratified auto-calibration approach was proposed and tested. First, the previous method of locating the plane at infinity is extended. In practice, more than two cameras may be available and each camera can capture more than two images, at different zoom settings of its lens. In this case and in order to cope with image noise, it is highly desirable to include all available cameras and images to locate the plane at infinity. Once the latter is retrieved, the no-skew and/or known aspect ratio constraints can be used to linearly estimate the so-called Image of the Absolute Conic (IAC) and hence all the intrinsic parameters. Two methods have been investigated for linearly calculating an estimate of the camera parameters

(a) the well-known linear least-squares through Singular Value Decomposition (SVD) [41]

(b) a Linear Matrix Inequality formulation which allows to enforce the requirement of a positive-definite IAC [59].

Our extensive experiments on simulated and real images, using a variable number of cameras, zoom settings and image noise, have shown that the obtained estimate of the intrinsic parameters are good enough for a simple nonlinear least-squares optimization procedure to converge towards the optimal parameters. This work has been published in the Image and Vision Computing journal [21].

- *An automatic detection of the scene's parallel planes using a zooming camera.* Detecting parallel planes is of great importance for many vision tasks. In our case, detecting parallel planes helps the auto-calibration process as well as the quality of scene reconstruction. A necessary condition is proposed in which parallel scene planes can be detected. Given a priori knowledge about a single pair of parallel planes, it is possible to identify all the other scene's parallel planes from uncalibrated images. We have proposed a new method where we have used the pair of parallel planes, resulting from two images taken by a zooming camera, as our a priori known pair of parallel planes, to automatically identify the scene's parallel planes. This work was published in the proceeding of Computer and Robot Vision (CRV) Conference [19].

1.5 Dissertation organization

The layout of the subsequent chapters is as follows. Chapter 2 briefly describes the needed background material that covers some basic and relevant mathematics of projective geometry and transformations. The pin-hole camera model and parameters are introduced. The concept of epipolar geometry and stratification of projective space is then presented.

A literature survey of camera auto-calibration is presented in chapter 3. The latter covers the different approaches for auto-calibration including methods which rely on special motion or scene constraints. As we are interested in zooming cameras, more attention will be given to auto-calibration of camera systems with varying settings. It will be shown that existing techniques for auto-calibrating zooming camera are nonlinear.

Chapter 4 and 5 respectively address the problem of affine and metric auto-calibration of a system of stationary zooming cameras. In chapter 6, the automatic parallel plane detection method is presented and examined. Finally, we conclude our work in chapter 7.

Chapter 2

Background

This chapter introduces the main geometrical concepts needed in this dissertation. First, it introduces the projective geometry of two and three dimensional spaces and the associated basic geometrical primitives of points, lines, planes, as well as the concepts of conics and quadrics. Moreover, the basic principles of image geometry and the pinhole camera model is also discussed. These concepts allow describing the projection process of world's scenes into images. Next, the concept of epipolar geometry relating information from multiple views of the three-dimensional world is reviewed. At last, extra attention is given to the transformation of the different geometrical concepts across the different geometrical layers including projective, affine, metric and Euclidean. Understanding these concepts will pave the road for developing methods to inverse the projection process, which will be the topic of the next chapter.

2.1 Projective geometry

The three-dimensional world is well described with Euclidean geometry. For example, we can describe a cubic box by different properties such as its size, equal edge lengths, square shaped sides, 90° angles between each pair of intersecting edges, parallel edges and sides,. . . etc. In addition, such properties are preserved under Euclidean transformation (translating and rotating the box will not alter its shape). While Euclidean geometry describes objects in our world so well, it is not the only type of geometry that exist and that we are familiar with! Consider two images of an object, say the same box, taken from different view points. During the image formation, three-dimensional object is projected onto a two-dimensional image plane. It is clear that the properties of the imaged object are no longer preserved and are different even among the two images of the same object. The lengths of the edges are no longer equal, squares became quadrangle, angles are no longer right, parallel edges may appear intersecting. We, as humans, still capable to identify the original Euclidean properties from these images. However, it becomes clear that Euclidean geometry alone is insufficient for machine vision. Euclidean geometry is indeed a subset of what is known as projective geometry. Furthermore, metric and affine are two other less restrictive geometries which come between them.

Since the main inputs to computer vision problems are two-dimensional images of three-dimensional world scene, projective geometry is an indispensable tool to model the perspective projection of the three-dimensional scenes onto a sequence of two-dimensional images. Using projective notation and concepts has numerous advantages. The encountered geometric entities and their relationship in computer vision can be represented in a linearized and compact algebraic form. Projective geometry also unify dealing with both finite and infinite point in the same manner and thus avoids special cases treatment and unnecessary limitations. However, these advantages come at the expense of additional ambiguities in comparison to the ordinary Euclidean space where ratios, lengths, and angles are no longer preserved while parallel lines may intersect!

An n-dimensional projective space is defined by $n+2$ basis points. A points in an n-dimensional projective space is represented by an $n+1$ column vector while projective transformation are represented by $(n+1) \times (n+1)$ matrices. Projective geometry may span any number of dimension, however, only notation and representations of the main geometric entities of the two-dimensional $\mathbb{P}^2$ and three-dimensional $\mathbb{P}^3$ projective spaces are described next. Most of the topics and material in this section can be found in the "Multiple View Geometry" book of Hartely and Zisserman [43]. The exploration is not meant to be extremely through, but to present a handy reference to the most significant working tools that will be needed in later chapters. For precise description and proofs of algebraic projective geometry, one should consult the original text.

2.1.1 Homogeneous coordinates

Homogenous coordinates systems used in projective geometry are quite analogues to the Cartesian coordinates systems used in Euclidean geometry. The Cartesian coordinate of a point $\tilde{\mathbf{q}}$ in n-dimensional Euclidian space is represented by an n-vector: $\tilde{\mathbf{q}} = (\mathsf{q}_1, \ldots, \mathsf{q}_n)^\intercal$. The homogenous representation $\mathbf{q}$ of this same point can be achieved by simply adding an extra component of 1 at the end: $\mathbf{q} = (\mathsf{q}_1, \ldots, \mathsf{q}_\mathsf{n}, 1)^\intercal$. In general, scaling is unimportant, so the point $(q_1, \ldots, q_n, 1)^\intercal$ represents the same point $(\alpha\mathsf{q}_1, \ldots, \alpha\mathsf{q}_\mathsf{n}, \alpha)^\intercal$ for any nonzero scalar α. In more general and compact form, homogeneous coordinates of a point $\mathbf{q}$ in an n-dimensional space is represented by an $(n+1)$ tuple vector as follow:

$$\mathbf{q} \doteq (\mathsf{q}_1, \ldots, \mathsf{q}_n, \mathsf{q}_{n+1})^\intercal \; \textit{or} \; \mathbf{q} \doteq (\tilde{\mathbf{q}}, 1)^\intercal.$$

where $\doteq$ indicates equality up to a non-zero scale factor.

While Cartesian coordinate system is limited to only represent points at finite distance from the origin, homogenous coordinates system is capable of equally expressing both finite and infinite points of the projective space uniformly. This is the most significant advantage of using homogenous coordinate system for projective space. It revokes limitation on designing algorithms as it avoids special cases treatment. An infinite points $\mathbf{q}_\infty$ (known as infinity points or ideal points) in homogenous coordinates is represented by setting the last component q_{n+1} to zero such that

$$\mathbf{q}_\infty \doteq (\mathsf{q}_1, \ldots, \mathsf{q}_n, 0)^\intercal.$$

The Cartesian counterpart $\tilde{\mathbf{q}}$ of a finite homogenous point $\mathbf{q}$ can be obtained back by simply dividing its components by the last one: $\tilde{\mathbf{q}} \doteq (\frac{\mathsf{q}_1}{\mathsf{q}_{n+1}}, \ldots, \frac{\mathsf{q}_n}{\mathsf{q}_{n+1}})^\intercal$. Since infinite points are not defined with Cartesian coordinates, if we try to divide by the last coordinate of a point at infinity, then we get the point $(\frac{\mathsf{q}_1}{0}, \ldots, \frac{\mathsf{q}_n}{0})^\intercal$ and thus $(\infty, \ldots, \infty)^\intercal$ which is infinite.

2.1.2 Two-dimensional projective space

The two-dimensional projective space $\mathbb{P}^2$, also known as the projective plane, is the set of all equivalent three-vectors $(\mathsf{q}_1, \mathsf{q}_2, \mathsf{q}_3)^\intercal$ excluding the null vector $(\,0\,,\,0\,,\,0\,)^\intercal$.

Points and lines in $\mathbb{P}^2$

A point $\mathbf{q}$ in $\mathbb{P}^2$ is represented by a homogenous 3-vector:

$$\mathbf{q} \doteq (\mathsf{q}_1, \mathsf{q}_2, \mathsf{q}_3)^\intercal$$

The set of vectors in $\mathbb{P}^2$ having the third coordinate set to zero, i.e. $(\mathsf{q}_1, \mathsf{q}_2, 0)^\intercal$, represent the set of points at infinity. This subset of infinite points lies on a single line called the *line at infinity*.

A line $\mathbf{l}$ on the projective plane $\mathbb{P}^2$ is also denoted by a 3-vector:

$$\mathbf{l} \doteq (l_1, l_2, l_3)^\intercal$$

Since both lines and points in projective plane are represented with homogenous 3-vectors, both have only 2 degrees of freedom.

A point $\mathbf{q}$ lies on the line $\mathbf{l}$ if and only if their vector products satisfy:

$$\mathbf{q}^\intercal \mathbf{l} \doteq 0 \tag{2.1}$$

Two distinct lines $\mathbf{l}_1$ and $\mathbf{l}_2$ intersect in a point $\mathbf{q}$ given by their cross-product:

$$\mathbf{q} \doteq \mathbf{l}_1 \times \mathbf{l}_2 \tag{2.2}$$

Similarly, two distinct points $\mathbf{q}_1$ and $\mathbf{q}_2$ define the line $\mathbf{l}$ given by:

$$\mathbf{l} \doteq \mathbf{q}_1 \times \mathbf{q}_2 \tag{2.3}$$

Equations (2.2) and (2.3) are just an example of the duality between points and lines in the projective plane. This duality also exists between points and planes in $\mathbb{P}^3$ and indeed for any other higher dimensions of the projective space $\mathbb{P}^n$.

In an alternative and more compact way, equations (2.2) and (2.3) can be rewritten as:

$$\mathbf{q} \doteq [\mathbf{l}_1]_\times \mathbf{l}_2 \text{ and } \mathbf{l} \doteq [\mathbf{q}_1]_\times \mathbf{q}_2 \tag{2.4}$$

where the 3×3 skew-symmetric $[\mathbf{v}]_\times$ for a given 3-vector $\mathbf{v} = (v_1, v_2, v_3)$ is on the form:

$$[\mathbf{v}]_\times = \begin{bmatrix} 0 & v_3 & -v_2 \\ -v_3 & 0 & v_1 \\ v_2 & -v_1 & 0 \end{bmatrix} \tag{2.5}$$

Conics

A conic in projective space $\mathbb{P}^2$ is a curve described by a second degree equation. Such curves can be written in homogenous form as follow:

$$C_1 q_1^2 + C_2 q_1 q_2 + C_3 q_2^2 + C_4 q_1 q_3 + C_5 q_2 q_3 + C_6 q_3^2 = 0$$

Since the conic consist of six homogenous elements (up to a non-zero scale factor), conics has five degree of freedom. The six elements of the conic can be arranged, more conveniently, by a 3×3 symmetric matrix on the form:

$$\mathsf{C} \doteq \begin{bmatrix} C_1 & C_2/2 & C_4/2 \\ C_2/2 & C_3 & C_5/2 \\ C_4/2 & C_5/2 & C_6 \end{bmatrix} \tag{2.6}$$

Due to duality between points and lines in $\mathbb{P}^2$, conics can be *point conics* or *line conic.* A point conic C is the locus of the set of points lying on the conic curve. Any point $\mathbf{q}$ on the conic must satisfy

$$\mathbf{q}^{\mathsf{T}}\,\mathsf{C}\,\mathbf{q} \doteq 0 \tag{2.7}$$

A line conic denoted C^* is the dual representation of the point conic. It can be though of as the envelope formed from the set of all lines tangent to the conic locus. All such lines must satisfy

$$\mathbf{l}^{\mathsf{T}}\,\mathsf{C}^*\,\mathbf{l} \doteq 0 \tag{2.8}$$

It can be shown that the relation between a full rank point conic C and its dual line conic C^* is given by its inverse as $\mathsf{C}^* \doteq \mathsf{C}^{-1}$.

2.1.3 Three-dimensional projective space

The three-dimensional projective space $\mathbb{P}^3$ consist of the set of all equivalent four-vectors $(\mathsf{Q}_1, \mathsf{Q}_2, \mathsf{Q}_3, \mathsf{Q}_4)^\top$ excluding the null vector $(\,0\,,\,0\,,\,0\,,\,0\,)^\top$.

Points and planes

A point $\mathbf{Q}$ in $\mathbb{P}^3$ is represented by a homogenous 4-vector:

$$\mathbf{Q} \doteq (\mathsf{Q}_1, \mathsf{Q}_2, \mathsf{Q}_3, \mathsf{Q}_4)^\top$$

The dual entity of a point in $(\mathbb{P})^3$ is a plane which is also represented by a homogenous 4-vector

$$\Pi \doteq (\pi_1, \pi_2, \pi_3, \pi_4)^\top$$

Any point $\mathbf{Q}$ lying on the plane Π must satisfy:

$$\Pi^\top \mathbf{Q} = \mathbf{Q}^\top \Pi = 0 \tag{2.9}$$

Any three non-collinear points define a plane Π as:

$$\begin{bmatrix} \mathbf{Q}_1^\top \\ \mathbf{Q}_2^\top \\ \mathbf{Q}_3^\top \end{bmatrix} \Pi = \mathbf{A}_q \Pi = 0 \tag{2.10}$$

where the plane Π can be computed as the 4-vector right null space of the matrix of points $\mathbf{A}_q$. Applying the duality principle between points and planes in $(\mathbb{P})^3$, any three non-coincident planes intersect in a point $\mathbf{Q}$:

$$\begin{bmatrix} \Pi_1^\top \\ \Pi_2^\top \\ \Pi_3^\top \end{bmatrix} \mathbf{Q} = \mathsf{A}_\pi \mathbf{Q} = 0 \tag{2.11}$$

where the point $\mathbf{Q}$ can be computed as the 4-vector right null space of the matrix $\mathbf{A}_\pi$ constituted from the 3 planes and has a full rank (i.e. not singular).

Lines

Lines in the projective $3-$diminsional space are self-dual. A point-based line, denoted $\mathbf{L}$, can be defined by any two points on the line, and its dual plane based line, denoted $\mathbf{L}^*$ can be represented by any two distinct planes both containing that line (i.e. intersecting exactly at it). Notional representation of lines in $(\mathbb{P})^3$ is less convenient. Among the different possible ways to represent lines, a line $\mathbf{L}$ and its dual $\mathbf{L}^*$ adopted in this dissertation is represented by 2 × 4 matrix such that:

$$\mathbf{L} \doteq \begin{bmatrix} \mathbf{Q}_1^\top \\ \mathbf{Q}_2^\top \end{bmatrix} \text{ and its dual } \mathbf{L}^* \doteq \begin{bmatrix} \Pi_1^\top \\ \Pi_2^\top \end{bmatrix} \tag{2.12}$$

where $\mathbf{Q}_1$ & $\mathbf{Q}_2$ are any two points on the line $\mathbf{L}$, $\Pi_1^\top$ & $\Pi_2^\top$ are any two distinct planes intersecting at the line $\mathbf{L}^*$.

Quadrics

In projective 3-dimensional space $\mathbb{P}^3$, a quadric has a similar concept of conic in $\mathbb{P}^2$. A *quadric* is a surface such as spheres, paraboloid, and cones. Quadrics are represented by a symmetric 4 × 4 homogeneous matrix $\boldsymbol{\Omega}$. Note that due to symmetrical form, the quadric matrix $\boldsymbol{\Omega}$ depends only on nine parameters (the 10 diagonal and above diagonal parameters minus 1 for scale). $\boldsymbol{\Omega}$ is designated to denote point-based quadrics while its dual $\boldsymbol{\Omega}^*$, is designated to plane-based quadric.

A point quadric $\boldsymbol{\Omega}$ is the locus of all points $\mathbf{Q}$ on its surface which satisfy the homogenous quadratic equation:

$$\mathbf{Q}^\top \boldsymbol{\Omega}\, \mathbf{Q} = 0, \tag{2.13}$$

whereas a *dual quadric* $\boldsymbol{\Omega}^*$ is defined by the locus of all planes Π which satisfy the quadratic equation:

$$\Pi^\top \boldsymbol{\Omega}^* \Pi = 0 \tag{2.14}$$

Similar to the relation between a conic and its dual in projective 2-dimensional space, the relation between a nonsingular (i.e. full rank) quadric $\boldsymbol{\Omega}$ and its dual $\boldsymbol{\Omega}^*$ in

3-dimensional space is given by its inverse $\mathbf{\Omega}^* \doteq \mathbf{\Omega}^{-1}$.

2.2 Transformation

A transformation in the projective space, also known as a *homography*, is a linear mapping from $\mathbb{P}^n \rightarrow \mathbb{P}^n$. Transformations in $n-$diminsional space are represented by homogenous $(n+1) \times (n+1)$ invertible matrices $\mathbf{T}$. The homogenous representation of transformation matrices implies that these matrices are defined up to non-zero scale factor, thus the transformations $\mathbf{T}$ and $\alpha\mathbf{T}$ are the same for all nonzero scalar α. The dual of a transformation $\mathbf{T}$ is denoted $\mathbf{T}^{-\top}$ where $\mathbf{T}^{-\top} = (\mathbf{T}^{-1})^\top = (\mathbf{T}^\top)^{-1}$.

2.2.1 Transformation in 2-dimensional space

In projective plane $\mathbb{P}^2$, a projective transformation is represented by a 3×3 matrix $\mathbf{H}$ of nine homogenous elements. Under such transformation, points are mapped to points and lines are mapped to lines. Projective transformation doesn't preserve angles, ratios, or parallelism. However, collinearity and cross ratios (ratio of ratios) are preserved and remain invariant.

A point $\mathbf{q}$ transforms into point $\mathbf{q}'$ as:

$$\mathbf{q} \rightarrow \mathbf{q}' \doteq \mathbf{T}\,\mathbf{q} \tag{2.15}$$

The corresponding transformation of a line $\mathbf{l}$ is given by

$$\mathbf{l} \rightarrow \mathbf{l}' \doteq \mathbf{T}^{-\top}\,\mathbf{l} \tag{2.16}$$

A conic C and its dual C^* transform as

$$\mathsf{C} \rightarrow \mathsf{C}' \doteq \mathbf{T}^{-\top}\,\mathsf{C}\,\mathbf{T}^{-1} \text{ and its dual } \mathsf{C}^* \rightarrow \mathsf{C}^{*\prime} \doteq \mathbf{T}\,\mathsf{C}^*\,\mathbf{T}^{\top} \tag{2.17}$$

2.2.2 Transformation in 3-dimensional space

Transformation in projective space $\mathbb{P}^3$ follows similar reasoning. A transformation is represented by 4×4 matrix $\mathbf{T}$ of 16 homogenous elements. Under projective transformation, points, planes, and lines are mapped as follow:

$$\mathbf{Q} \rightarrow \mathbf{Q}' \doteq \mathbf{T}\,\mathbf{Q} \tag{2.18}$$

$$\Pi \rightarrow \Pi' \doteq \mathbf{T}^{-\top}\,\Pi \tag{2.19}$$

$$\mathbf{L} \rightarrow \mathbf{L}' \doteq \mathbf{T}\,\mathbf{L} \text{ and its dual } \mathbf{L}^* \rightarrow \mathbf{L}'^* \doteq \mathbf{T}^{-\top}\,\mathbf{L}^* \tag{2.20}$$

where $\mathbf{L}$ and it dual $\mathbf{L}^*$ are the point and plane line representation in section (2.1.3).

Under homography transformation $\mathbf{T}$, quadrics $\mathbf{\Omega}$ and dual quadrics $\mathbf{\Omega}^*$ transforms as

$$\mathbf{\Omega} \rightarrow \mathbf{\Omega}' \doteq \mathbf{T}^{-\top}\,\mathbf{\Omega}\,\mathbf{T}^{-1} \tag{2.21}$$

$$\mathbf{\Omega}^* \rightarrow \mathbf{\Omega}^{*\prime} \doteq \mathbf{T}\,\mathbf{\Omega}^*\,\mathbf{T}^{\top} \tag{2.22}$$

2.3 Geometry of images

This section is mainly concerned with the geometry of image formation and camera model which is of crucial importance to recover the scene $3D$ geometry. The involved camera model during image formation establishes strong relationship between the $3D$ scene points and their corresponding $2D$ image points. Such relationship is strictly governed by the camera intrinsic and extrinsic parameters. Among a number of available camera models, only the *pinhole camera model* is reviewed; which is the mainly and most practically used camera model in solving vision problems.

2.3.1 Pinhole camera model

The pinhole camera model provides good approximation to digital lenses camera with CCD-like sensors. It models the perspective projection of world's points on the two dimensional image plane. For simplicity, consider the *camera center* C is positioned at the origin of the world coordinate system $< X_w, Y_w, Z_w \mid O_w >$ and the camera axes system $< X_c, Y_c, Z_c \mid C >$ is aligned with it. The image plane I is at distance f from the camera center (see Figure 2.1). The optical axis Z_c is the line passing through the optical center C (also known as center of projection, focal point) and perpendicular to the retinal image plane I where it intersects it at the *principal point* $p = (u_0, v_0)^\mathsf{T}$. Note that the image plane I in actual camera is behind the center of projection at distance f and therefore the projected image is inverted. However, for simplicity, this inversion can be avoided by shifting the image plane to the front of the camera center instead as illustrated in Figure (2.1).

Now consider the projection of a single $3D$ world point $\mathbf{Q}$ to the image plane. The emanating line from the $3D$ world‘s point $\mathbf{Q} = (\mathsf{Q}_x, \mathsf{Q}_y, \mathsf{Q}_z, 1)^\mathsf{T}$ intersects the image

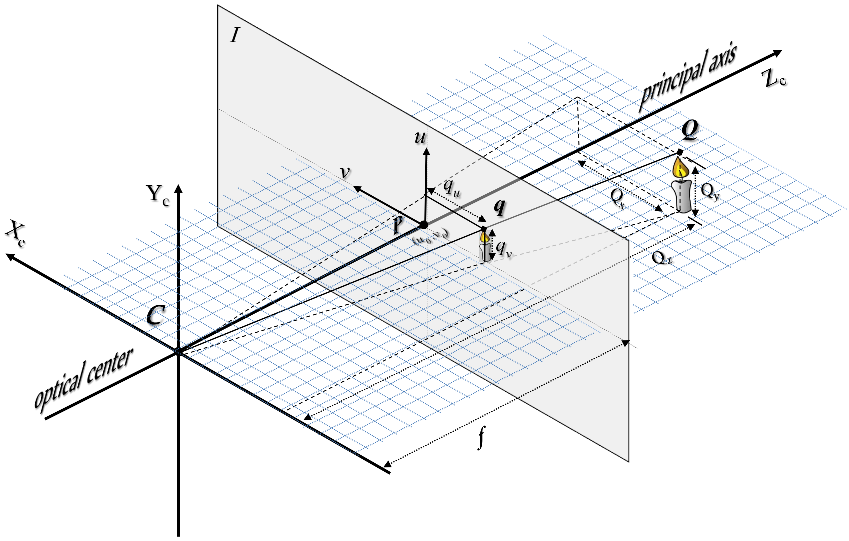


Figure 2.1: *Pinhole camera geometry*

plane at the $2D$ point $\mathbf{q} = (\mathsf{q}_u, \mathsf{q}_v, 1)^\intercal$. Using simple triangulation, it is trivial to conclude that the following ratios are equal:

$$\frac{\mathsf{q}_u}{\mathsf{Q}_x} = \frac{\mathsf{q}_v}{\mathsf{Q}_y} = \frac{f}{\mathsf{Q}_z} \tag{2.23}$$

By rearranging equation 2.23, the coefficient of the image point $\mathbf{q} = (\mathsf{q}_u, \mathsf{q}_v)^\intercal$ can be given by the equations

$$\mathsf{q}_u = f\frac{\mathsf{Q}_x}{\mathsf{Q}_z} \text{ and } \mathsf{q}_v = f\frac{\mathsf{Q}_y}{\mathsf{Q}_z} \tag{2.24}$$

This equation 2.24 can be represented in terms of homogeneous coordinates more conveniently as:

$$\begin{bmatrix} q_u \\ q_v \\ 1 \end{bmatrix} \doteq \begin{bmatrix} fQ_x \\ fQ_y \\ Q_z \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} Q_x \\ Q_y \\ Q_z \\ 1 \end{bmatrix} \tag{2.25}$$

or in more compact and general form as :

$$\mathbf{q} \doteq \mathsf{P}_{3\times4}\, \mathbf{Q}_{4\times1} \tag{2.26}$$

The intrinsic parameters

Points in equation 2.25 expressed in image coordinate system are normally specified in terms of metric units (e.g. millimeters) where the principal point p is the origin of the image coordinate system. However, such points in digital images are expressed in terms of pixels coordinates where the image origin is typically positioned at the upper-left corner. It is therefore important to take in to account the mapping between the image and pixel coordinates systems (see Figure 2.2). In the following, we designate a trailing superscripts to denote the coordinate frame in which the point is expressed. Let the point $\mathbf{q}^{\mathsf{i}} = (q_u^i, q_v^i, 1)^{\mathsf{T}}$ expressed in image coordinate frame and $\mathbf{q}^{\mathsf{p}} = (q_u^p, q_v^p, 1)^{\mathsf{T}}$ represent the same image point $\mathbf{q}$ expressed however in pixel coordinate frame.

To model the relationship between the image coordinate frame and the pixel coordinate frame we need to:

- convert the image coordinates from metric units into pixels.
- translate the origin of the pixel coordinate frame from the principal point to

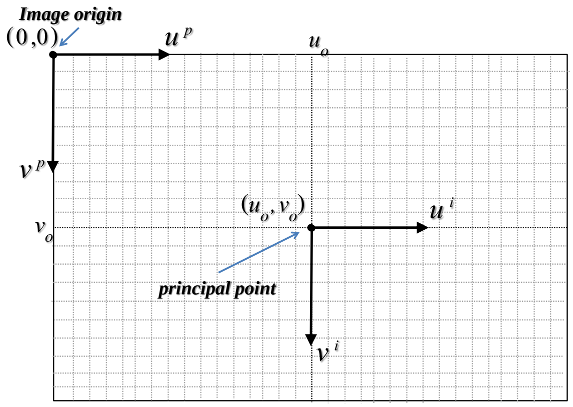


Figure 2.2: *Using different image*$^{(i)}$ *and pixel*$^{(p)}$ *coordinate systems*

the upper-left corner.

These two steps can be represented mathematically as:

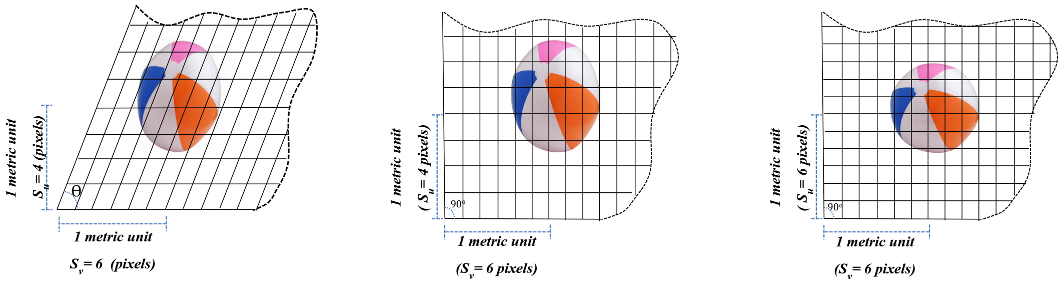
$$q_u^p = s_u q_u^i + u_o \text{ and } q_v^p = s_v q_v^i + v_o, \tag{2.27}$$

where the scales s_u and s_v are the number of pixels per metric unit distance along u and v axial directions in the image coordinate frame, u_o and v_o are the coordinates of the principal point.

Equation 2.27 can be rewritten in terms of homogeneous coordinates as:

$$\begin{bmatrix} q_u^p \\ q_v^p \\ 1 \end{bmatrix} = \begin{bmatrix} s_u & 0 & u_o \\ 0 & s_v & v_o \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} q_u^i \\ q_v^i \\ 1 \end{bmatrix} \tag{2.28}$$

Equations 2.25 and 2.28 can be combined to yield the transformations of the points



(a) non-rectangular pixels (b) rectangular pixels (c) square pixels

Figure 2.3: *Different intrinsic parameters*

from camera coordinate frame to the relative pixel coordinate frame as:

$$\begin{bmatrix} q^p_u \\ q^p_v \\ 1 \end{bmatrix} = \begin{bmatrix} s_u & 0 & u_o \\ 0 & s_v & v_o \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} Q_x \\ Q_y \\ Q_z \\ 1 \end{bmatrix} \tag{2.29}$$

In digital cameras, the image is formed over an array of light sensitive sensor elements called pixels. The physical shape of these pixels has an effect on the coordinate of the image points. This phenomena is illustrated in Figure 2.3. The ratio of the number of pixels per metric unit along the horizontal and vertical image axial directions is the *aspect ratio* τ of the camera and computed as $\tau = \mathsf{s}_u/\mathsf{s}_v$. The angle θ between the pixel sensor's axial coordinates u and v models the skewness of the camera. When the angle is of 90^o, the camera is said to have rectangular pixels. If also the pixel size along vertical and horizontal coordinates are equal (i.e. aspect ratio $\tau = 1$), the camera is said to have square pixels. The skew factor γ is introduced to model such skewness in the sensor where $\gamma = s_v \mathsf{q}^i_v \tan\theta$. It worth mentioning that the angle θ in today's modern CCD/CMOS digital cameras is very close to 90^o and it is often safe to consider zero skew factor $\gamma = 0$. By incorporating the skew factor, Equation

2.29 can be reformulated as:

$$\begin{bmatrix} q_u^p \\ q_v^p \\ 1 \end{bmatrix} = \begin{bmatrix} f_u & \gamma & u_o \\ 0 & f_v & v_o \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} Q_x \\ Q_y \\ Q_z \\ 1 \end{bmatrix} \tag{2.30}$$

or as:

$$\begin{bmatrix} q_u^p \\ q_v^p \\ 1 \end{bmatrix} = \begin{bmatrix} \tau f & \gamma & \mathsf{u}_o \\ 0 & f & \mathsf{v}_o \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} Q_x \\ Q_y \\ Q_z \\ 1 \end{bmatrix} \tag{2.31}$$

where $f_u = fs_u$ is focal length (in pixels) along the u coordinate direction, $f_v = fs_v$ is the focal length (in pixels) along the v coordinate directions, and γ model the skew factor of the pixels. Equation 2.30 can be written in more compact form as:

$$\mathbf{q}^p \doteq \mathsf{K}\,[I \mid 0]\,\mathbf{Q}^c, \tag{2.32}$$

where $\mathbf{q}^p$ is the image point in pixel coordinate frame, K is the camera *intrinsic* matrix, I is the 3×3 identity matrix, and 0 is a 3 null vector, and $\mathbf{Q}^c$ is the $3D$ world point in camera coordinate frame.

The extrinsic parameters

To simplify the previous derivation, we have considered that the camera coordinate system is aligned with the world coordinate system. It is very important to be able to express scene points coordinates in a different coordinate system especially when the relation between these coordinates is unknown. In other words, we need to transform a scene point $\mathbf{Q}^{\mathsf{w}}$ expressed in world coordinate system to point $\mathbf{Q}^{\mathsf{c}}$ expressed in camera coordinate system (see Figure 2.4). This can be done using an 4×4 homogenous rigid transformation which incorporate a 3×3 orthogonal rotation matrix R and a $3-$translation vector $\mathbf{t}$ as follow:

$$\mathbf{Q}^{\mathsf{c}} = \begin{bmatrix} \mathsf{R} & \mathbf{t} \\ 0_3 & 1 \end{bmatrix} \mathbf{Q}^{\mathsf{w}} \Leftrightarrow \begin{bmatrix} Q_x^c \\ Q_y^c \\ Q_z^c \\ 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} Q_x^w \\ Q_y^w \\ Q_z^w \\ 1 \end{bmatrix} \tag{2.33}$$

By inserting Equation 2.33 in 2.30, the full perspective projection model which relates a $3D$ world point $\mathbf{Q}^w = (\mathsf{Q}_x^w, \mathsf{Q}_y^w, \mathsf{Q}_z^w)^\intercal$ in the world coordinate frame to its projection point $\mathbf{q}^p = (\mathsf{q}_u^p, \mathsf{q}_v^p)^\intercal$ expressed in the image pixel coordinate frame can be expressed by:

$$\begin{bmatrix} q_u^p \\ q_v^p \\ 1 \end{bmatrix} = \begin{bmatrix} f_u & \gamma & u_o \\ 0 & f_v & v_o \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} Q_x^w \\ Q_y^w \\ Q_z^w \\ 1 \end{bmatrix}$$

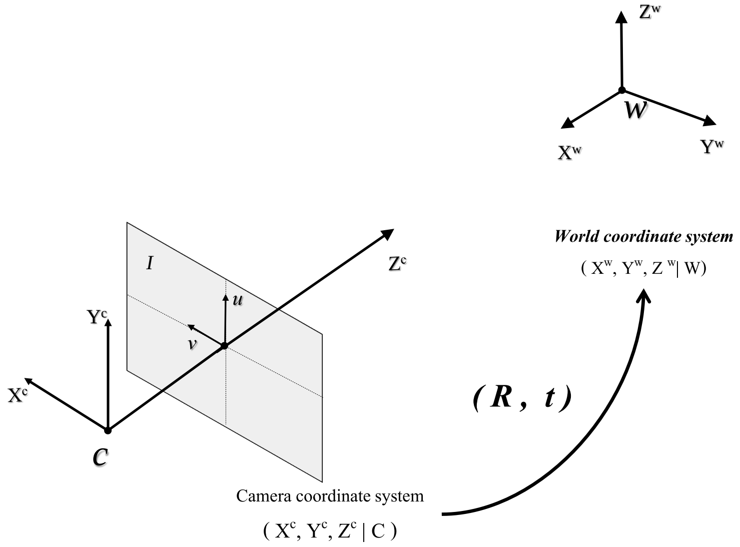


Figure 2.4: *Camera and world coordinate systems*

$$\begin{bmatrix} q_u^p \\ q_v^p \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} f_u & \gamma & u_o \\ 0 & f_v & v_o \\ 0 & 0 & 1 \end{bmatrix}}_{\text{Intrinsic K}} \underbrace{\begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix}}_{\text{Extrinsic}(\mathsf{R},\mathbf{t})} \begin{bmatrix} Q_x^w \\ Q_y^w \\ Q_z^w \\ 1 \end{bmatrix} \tag{2.34}$$

or more compactly:

$$\mathbf{q}^p \doteq \mathsf{K}\,[\mathsf{R} \mid \mathbf{t}]\,\mathbf{Q}^w, \tag{2.35}$$

2.4 Epipolar geometry

The pinhole model, discussed in subsection 2.3.1, describes the geometrical relationship between scene points and their projections on the image plane. Epipolar geometry, on the other hand, describes the geometrical relationship between the projections of scene points of two distinct views. Let I and I' be two distinct image planes observing $3D$ scene points $\mathbf{Q}_i$ as shown in Figure 2.5. Image points $\mathbf{q}_\mathsf{i}$ and $\mathbf{q}'_i$ of the scene point $\mathbf{Q}_i$ in image I and I' respectively. The image of the first camera center C on the second image plane I' is the epipole $\mathbf{e}'$. The second camera center in turn is imaged on the first image plane I as the epipole $\mathbf{e}$. The line segment joining the two cameras' centers C and C' is called the baseline and intersects the two image planes in the epipoles $\mathbf{e}$ and $\mathbf{e}'$. The plane passing through the two cameras' centers C and C' and the scene point $\mathbf{Q}_i$ is the epipolar plane $\mathsf{\Pi}_i$. Each epipolar plane $\mathsf{\Pi}_i$ intersect the first image plane I in the epipolar line $\mathbf{l}_i$ and the second image plane I' in the epipolar line $\mathbf{l}'_i$.

The epipolar geometry is of great importance in the context of 3D reconstruction. It is the only information we can get from uncalibrated images of a rigid scene, thus it is often considered, by many authors, as weak-calibration. Knowing the epipolar geometry of a pair or more of images allows recovering the projective 3D structure of the scene. In fact, Euclidean reconstruction and camera self-calibration methods depends on this weak calibration as a primary step. Furthermore, epipolar geometry is of great importance for stereo matching. Instead of searching the whole image pixel points, epipolar geometry restrict the search of a point in the first view to the corresponding epipolar line in the second view.

2.4.1 The fundamental matrix

The epipolar geometry is represented algebraically by the fundamental matrix. The fundamental matrix F is a 3×3 singular matrix of rank 2 ([43]). For any two images taken by two non-coincident camera centers, the fundamental matrix F constrains all

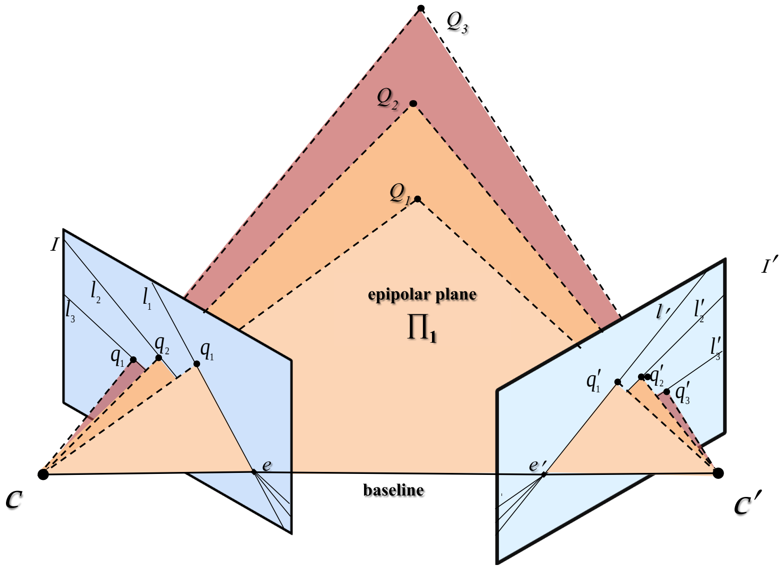


Figure 2.5: *Epipolar geometry*

image points $\mathbf{q}_i$ in the first view with their corresponding points $\mathbf{q}'_i$ in the second view such that:

$$\mathbf{q}_i'^{\top}\,\mathsf{F}\,\mathbf{q}_i = 0 \tag{2.36}$$

One nice property of the fundamental matrix is that its transpose $\mathsf{F}^{\top}$ provides the opposite relation by relating image points $\mathbf{q}'_i$ in the second view with their corresponding points $\mathbf{q}_i$ in the first view (i.e. $\mathsf{F}'{=}\mathsf{F}^{\top}$) and thus

$$\mathbf{q}_i^{\top}\mathsf{F}^{\top}\mathbf{q}'_i = 0 \tag{2.37}$$

Any points in one image is constrained by the fundamental matrix to its corresponding epipolar line in the other view where its corresponding image match must coincide. More specifically,

$$\mathbf{l}'_i = \mathsf{F}\,\mathbf{q}_i \text{ and } \mathbf{l}_i = \mathsf{F}^\top \mathbf{q}'_i \tag{2.38}$$

Finally, observe that for any point $\mathbf{q}_i$, the epipolar line $\mathbf{l}'_i = \mathsf{F}\mathbf{q}_i$ intersect the epipole $\mathbf{e}'$ in the other view (see Figure 2.5). Thus, using the epipolar constraint, Equation 2.36, $\mathbf{e}'^\top(\mathsf{F}\mathbf{q}_i) = (\mathbf{e}'^\top\mathsf{F})\mathbf{q}_i = 0$. This indicate that $e'^\top\mathsf{F} = 0$ and thus $\mathbf{e}'$ is the left null-vector of F. In similar manner, $\mathbf{e}^\top\mathsf{F}^\top = 0$ and therefore $\mathbf{e}$ is the right null-vector of $\mathsf{F}^\top$.

2.4.2 Computation of the fundamental matrix

In general, eight points matches are enough to compute F, linearly, by stacking the epipolar constraint equations for each pair of points $\mathbf{q}_i^\top\mathsf{F}^\top\mathbf{q}'_i = 0$ and solving for the nine unknown elements of F using a least squares approach such as the SVD. Computing the fundamental matrix from point matches was first introduced by Longuet-Higgins [61] as the Essential matrix in which the images are assumed to be calibrated. The generalization of the essential matrix to uncalibrated images, as the fundamental matrix, is due to Fugeras [24]). Boufama in [9] introduced a novel linear method for computing the fundamental matrix, providing good estimation without the need for the compulsory non-linear optimization refitment step. Hartley [38], enhanced the linear computation of the fundamental matrix further by suggesting simple normalization and scaling of the uncalibrated images followed by enforcing rank two constraint. Unique solution from linear computation requires eight point matches. Recall that the fundamental matrix consist of 9 homogenous elements but has only 7 degrees of freedom. One degree of freedom is related to the overall scale factor as F

is a homogenous matrix. Another degree of freedom is removed as F is of rank 2 and has a zero determinant. The fundamental matrix F can be computed non-linearly from seven points [64]. The fundamental matrix has been under extensive research in the last two decades and extensive effort has been put in automatic and robust computation from point as well as line matches (see for example [96], [71], [101]).

2.5 Stratified three-dimensional geometry

According to human perception, obtainable projective three-dimensional structure from point matches can differ very much from the original scene. Perhaps the most important question to address at this point is "How satisfactory is the obtained $3D$ projective structure for performing computer vision tasks?". In projective space lengthes, ratios, and angles are not preserved, parallel line and planes appear intersecting in general, orthogonality is not preserved,...etc. Indeed such projective structure is not satisfactory for the majority of vision task. Fortunately, the projective three-dimensional structure can be upgraded to Euclidean using a proper 4×4 transformation. Actually, there are different classes of geometry between the simplest projective geometry and the most restrictive Euclidean form and such upgrade may also pass over an intermediate affine and metric structure. In order to obtain the proper transformation which upgrade the projective structure to the desired space (i.e. affine, metric, or Euclidean), certain special geometrical entities must be specified. The geometrical hierarchy and transformation grouping are tightly related to the invariants such transformation leaves when applied to these special geometrical entities and properties. It is also important to notice that the different group of transformation are actually subgroups of each other. The Euclidean is a subgroup of metric, and both are subgroup of affine group whereas all of them are subgroup of the projective class.

In this section, the different geometrical transformation groups required to upgrade the $3D$ structure from the simplest projective geometry to the most restrictive Euclidean structure are reviewed. Note that a similar concept can be applied to any n-dimensional space, but the treatment here is limited to the three-dimensional space as it is the most relevant to this dissertation.

2.5.1 Projective transformation

The group of projective transformation is the most general one with weakest structure. It has the least number of invariants and therefore is the super-group of all other groups of transformation. A three-dimensional projective transformation, as seen in section (2.2), is represented b a full rank 4×4 matrix T such as:

$$\mathbf{T}_{proj} \doteq \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \\ p_{41} & p_{42} & p_{43} & p_{44} \end{bmatrix} \doteq \begin{bmatrix} \mathbf{A}_{3\times3} & \mathsf{t}_{3\times1} \\ \mathsf{v}_{3\times1}^{\top} & 1 \end{bmatrix} \tag{2.39}$$

where p_{ij} is a scalar, v and t are $3-$vectors, and $\mathbf{A}$ is an arbitrary 3 matrix. As such transformation is homogenous and defined up to a nonzero scale factor, only 15 elements are essentials. Projective transformation preserves the incidence and collinearity relations of points invariant. The cross-ration (i.e. the ratio of ratios) is an invariant projective property as well. Since T is nonsingular, a dual transformation T^{*} can be obtained by the inverse of its transpose $\mathbf{T}^{*} = (\mathbf{T}^{-1})^{\top} = (\mathbf{T}^{\top})^{-1}$.

2.5.2 Affine transformation

An affine transformation is represented by the homogenous transformation matrix $\mathbf{T}_{aff}$

$$\mathbf{T}_{aff} \doteq \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} \doteq \begin{bmatrix} \mathbf{A}_{3\times3} & \mathsf{t}_{3\times1} \\ \mathsf{0}_{3\times1}^{\top} & 1 \end{bmatrix} \tag{2.40}$$

The affine group of transformation is an intermediate group located between projective and metric groups. It is more restrictive than projective but more general than metric. As affine is a subgroup of projective geometry, all projective invariants are certainly affine invariants as well. The most special thing about affine group of transformations is that they preserve parallelism. A $n-$dimesnional affine space differs than projective space by identifying a special hyper-plane at infinity. As $3-$dimensional space is our concern here, identifying the true plane at infinity Π_∞ in the projective $3-$dimensional space allows to retrieve the affine structure. Since such plane has 3 degrees of freedom, an affine transformation is more restrictive than projective transformation and has 12 degrees of freedom (i.e. 15 d.o.f. projective less 3 for plane at infinity). The plane at infinity has a canonical position $\Pi_\infty = (0,0,0,1)^\top$ in an affine space. Parallel lines and parallel planes intersect at the plane at infinity. Ratios of lengthes along parallel directions are also affine invariants and an affine transformation leaves the plane at infinity globally invariant (i.e. $\Pi_\infty \doteq \mathbf{T}_{aff}^{-\top}\Pi_\infty$).

2.5.3 Metric transformation

Metric transformation is the group of similarity transformations. These transformations correspond to Euclidean transformations (i.e. rotation + translation) with isotropic scaling. In metric transformation case two new properties retains invariants: the angles and relative length, but not the absolute ratios and lengths. Metric transformation is in general the highest level of structure that can be recovered from images, unless knowledge about the exact length or size of an object in the scene is available.

Metric transformations have the following representation:

$$\mathbf{T}_{met} \doteq \begin{bmatrix} \sigma r_{11} & \sigma r_{12} & \sigma r_{13} & t_x \\ \sigma r_{21} & \sigma r_{22} & \sigma r_{23} & t_y \\ \sigma r_{31} & \sigma r_{32} & \sigma r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \doteq \begin{bmatrix} \sigma \mathbf{R}_{3\times3} & \mathsf{t}_{3\times1} \\ \mathsf{0}_{3\times1}^{\top} & 1 \end{bmatrix} \tag{2.41}$$

where r_{ij} are the coefficient of a 3×3 rotational matrix $\mathbf{R}$, $\mathsf{t} = (t_x, t_y, t_z)^{\top}$ a translation vector and σ a nonzero scaling factor. Note that any rotation matrix $\mathbf{R}$ is an orthonormal matrix with unity determinant of 1 and such matrix has only 3 degrees of freedom. Therefore, a metric transformation accounts to 7 independent degrees of freedom: 1 for scale factor σ, 3 for translation vector t, and finally 3 for the rotation matrix $\mathbf{R}$.

Similar to $3D$ affine case where its properties are related to the plane at infinity, the new metric properties are related to a specific imaginary conic on the plane at infinity called the *absolute conic* (AC) and denoted $\mathbf{\Omega}_{\infty}$. The absolute conic is a point conic of imaginary points (no real points). A metric transformation transforms the absolute conic into itself. Every points $\mathbf{Q} = (Q_1, Q_2, Q_3, Q_4)^{\top}$ on a canonical metric conic (i.e. the plane at infinity on the form $\Pi_\infty = (0,0,0,1)^\top$) must satisfy

$$\left.\begin{array}{r} Q_1^2 \;+\; Q_2^2 \;+\; Q_3^2 \\ Q_4^2 \end{array}\right\} = 0 \tag{2.42}$$

Note that algebraic representation for such a conic requires two equations. For this reason, the absolute conic is more practically utilized by its dual in 3-dimensional space: the dual absolute conic denoted $\mathbf{\Omega}_\infty^*$. The dual absolute conic $\mathbf{\Omega}_\infty^*$ can be represented as a single quadric called the absolute dual quadric and introduced to computer vision by [97]. The absolute quadric has a simple canonical form:

$$\mathbf{\Omega}_\infty^* \doteq \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{2.43}$$

Note that the null space of $\mathbf{\Omega}_\infty^*$ is the infinity plane $\Pi_\infty = (0,0,0,1)^\top$. A similar concept exist in $2-$deminsional space where the plane at infinity is the plane under consideration. Under such consideration, ω_∞ and ω_∞^* denotes the two-dimensional representation of the absolute conic and the dual absolute conic respectively. The canonical form of these entities are given by:

$$\omega_\infty \doteq \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \text{and} \quad \omega_\infty^* \doteq \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.44}$$

Since the absolute conic resides on the plane at infinity, it projects on image planes as a conic which depends only on the internal parameters of the camera and independent of its pose or position. This can be observed by noting that points of the infinity plane $\mathbf{Q}_\infty = (Q_1, Q_2, Q_3, 0)^\top = (\tilde{\mathbf{Q}}_\infty, 0)^\top$ projects on camera P as :

$$\mathbf{q} \doteq \mathsf{K}\,[\mathsf{R} \mid \mathbf{t}]\,\tilde{\mathbf{Q}}_\infty$$

and thus points on the image plane and the infinity plane are related by the homograpghy transformation $\mathbf{T}$

$$\mathbf{q} \doteq \mathsf{KR}\tilde{\mathbf{Q}}_\infty = \mathbf{T}\tilde{\mathbf{Q}}_\infty$$

Thus the image of the absolute conic, also known as IAC and denoted ω , can be obtained from the transformation of the $2D$ form of the absolute conic ω_∞ using (2.17) as follow:

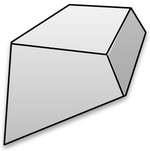
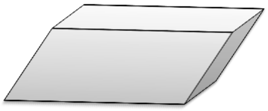
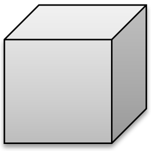
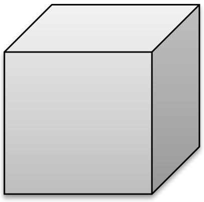
$$\omega = \mathbf{T}^{-\top}\,\omega_\infty\,\mathbf{T}^{-\top} = (\mathsf{KR})^{-\top}\omega_\infty(\mathsf{KR})^{-1} = \mathsf{K}^{-\top}\mathsf{R}^{-\top}\mathsf{R}^{-1}\mathsf{K}^{-1} = \mathsf{K}^{-\top}\mathsf{K}^{-1}$$

Similarly, it can be shown that the image of the dual absolute conic (DIAC) denoted $\omega^* = \omega^{-1} = \mathsf{KK}^\top$. This is of great importance as the camera internal parameters K can be obtained using cholesky decomposition(get the re) as will be shown in next chapter.

2.5.4 Projective to metric stratified transformation

In the previous section, the different geometrical group of transformation where formed assuming a canonical position of the plane at infinity and absolute conic. Driving such transformation is straight forward once we are in Euclidian space. In

Table 2.1: List of the different groups of three-dimensional transformation showing structure ambiguity, the number of degrees of freedom, canonical transformation matrix, related special geometric entities, and invariant properties.

Projective	**Affine**	**Metric**	**Euclidean**
			
15 dof	12 dof	7 dof	6 dof
$\mathbf{T}_{proj} = \begin{bmatrix} \mathrm{A} & \mathrm{t} \\ \mathrm{v}^{\mathsf{T}} & 1 \end{bmatrix}$	$\mathbf{T}_{aff} = \begin{bmatrix} \mathrm{A} & \mathrm{t} \\ 0^{\mathsf{T}} & 1 \end{bmatrix}$	$\mathbf{T}_{met} = \begin{bmatrix} s\mathrm{R} & \mathrm{t} \\ 0^{\mathsf{T}} & 1 \end{bmatrix}$	$\mathbf{T}_{Euc} = \begin{bmatrix} \mathrm{R} & \mathrm{t} \\ 0^{\mathsf{T}} & 1 \end{bmatrix}$
	Plane at Infinity $\prod_\infty$	Absolute Conic $\mathbf{\Omega}^*_\infty$	Absolute Scale
cross-ratio incidence	cross-ratio incidence parallelism	cross-ratio incidence parallelism relative length angle	cross-ratio incidence parallelism relative length angle length volume

computer vision, however, the initial obtainable 3-dimensional reconstruction of the scene's geometrical entities (i.e. points, lines, planes,...etc) from uncalibrated images are up to an arbitrary projective ambiguity. In such projective structure, the plane at infinity and the absolute conic are changed from their canonical position to new unknown location. The general goal of a useful reconstruction, for vision tasks, is to obtain at least a metric representation. The obtained projective structure can be transformed to metric, or Euclidean, using a proper 4 × 4 homography T_{PM} which maps each projective points $\mathbf{Q}^p$ to its metric positions $\mathbf{Q}^m$ as:

$$\mathbf{Q}^m \doteq \mathbf{T}_{PM}\mathbf{Q}^p \tag{2.45}$$

The upgrade homography transformation $\mathbf{T}_{PM}$ can be computed directly in one step or can be stratified into two step transformations: a projective to affine transformation $\mathbf{T}_{PA}$ and an affine to metric transformation $\mathbf{T}_{AM}$. In practice, the advantage of upgrading the projective structure to affine one first is more desirable as it allows a linear upgrade to metric once the affine structure is computed. A complete transformation from projective to metric can be computed afterward as:

$$\mathbf{T}_{PM} \doteq \mathbf{T}_{AM}\mathbf{T}_{PA} \tag{2.46}$$

Projective to affine upgrade

In a given projective 3-dimensional representation, the plane at infinity is no longer has its canonical position. The first step for the obtainment of an affine reconstruction is the identification of the correct plane at infinity. The affine properties of the structure can then be obtained if the chosen plane at infinity, in the given projective structure, is mapped to the true plane at infinity position in the world. In general, plane at infinity can be identified from known affine properties and in particular parallelism. Knowledge about parallel entities in the seen can be translated into constrains on the position of the plane at infinity. For example, two or more parallel lines intersect in a vanishing point on the plane at infinity and three such points are enough to determine the plane at infinity. Actually, locating the plane at infinity from images can be done in different ways and will be discussed more thoroughly in section (auto-calibration).

Once the plane at infinity Π_∞ is located, a simple transformation can be applied to bring back the plane at infinity to the affine canonical value $(0, 0, 0, 1)^\top$. Denoting and scaling the first 3 non-homogenous elements of the plane at infinity such as $\Pi_\infty = (\tilde{\Pi}_\infty, 1)^\top$, It can be simply verified that such transformation is on the form :

$$\mathbf{T}_{PA} \doteq \begin{bmatrix} \mathbf{A}_{3\times3} & \mathsf{t}_3 \\ \tilde{\Pi}_\infty^\top & 1 \end{bmatrix}$$

where $\mathbf{A}_{3\times3}$ is any arbitrary matrix with nonzero determinant and t_3 an arbitrary 3-vector, and $\mathsf{0}_3$ the 3 null vector. For simplicity $\mathbf{A}_{3\times3}$ is normally chosen as the identity $\mathsf{I}_{3\times3}$ matrix and t_3 as the null vector 0_3.

$$\mathbf{T}_{PA} \doteq \begin{bmatrix} \mathsf{I}_{3\times3} & \mathsf{0}_3 \\ \tilde{\Pi}_\infty^\top & 1 \end{bmatrix} \tag{2.47}$$

Note that such transformation maps the plane at infinity to the canonical location $(0, 0, 0, 1)^\top$.

$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \doteq \begin{bmatrix} \mathbf{I}_{3\times 3} & \mathsf{0}_3 \\ \tilde{\Pi}_\infty^\top & 1 \end{bmatrix}^{-\top} \Pi_\infty^\top$$

Affine to Metric upgrade

In order to upgrade an affine 3-dimensional representation of a structure to metric, the absolute conic or one of its associated entities such as its dual must be retrieved. This is possible once the plane at infinity is identified. It is, however, also possible to retrieve both of the abolsute conic and its supporting plane (the plane at infinity) all at once as will be disccused in the (Self-claibraion). The discussion here is limited only to the upgrade from affine to metric.

Combining Equation 2.22 and Equation 2.40 one can verify that the dual absolute quadric transforms as follow :

$$\mathbf{\Omega}_\infty^* \doteq \mathsf{T}\mathbf{\Omega}_\infty^*\mathsf{T}^\top \doteq \begin{bmatrix} \mathbf{A} & \mathsf{t} \\ \mathsf{0}_3^\top & 1 \end{bmatrix} \begin{bmatrix} \mathsf{I}_{3\times 3} & \mathsf{0}_3 \\ \mathsf{0}_3^\top & 1 \end{bmatrix} \begin{bmatrix} \mathbf{A}^\top & \mathsf{0}_3 \\ \mathsf{t}^\top & 1 \end{bmatrix} \doteq \begin{bmatrix} \mathbf{A}\mathbf{A}^\top & \mathsf{0}_3 \\ \mathsf{0}_3^\top & 1 \end{bmatrix} \tag{2.48}$$

Since such transformation leaves the absolute dual quadric unchanged, such transformation must be a similarity transformation (i.e. metric). Under these circumstances the absolute conic and its dual have the following form:

$$\omega_\infty \doteq \mathbf{A}^{-\top}\mathbf{A}^{-1} \text{ and } \omega_\infty^* \doteq \mathbf{A}\mathbf{A}^\top. \tag{2.49}$$

Consequently,a simple choice for the homograpghy transformation from affine to metric can be given by :

$$\mathbf{T}_{AM} \doteq \begin{bmatrix} \mathbf{A}^{-1} & \mathsf{0}_3 \\ \mathsf{0}_3^\top & 1 \end{bmatrix} \tag{2.50}$$

Combining transformation from equation (2.47) and 2.50) a homography for upgrading the structure from projective to metric as:

$$\mathbf{T}_{PM} \doteq \mathbf{T}_{AM}\mathbf{T}_{PA} \doteq \begin{bmatrix} \mathbf{A}^{-1} & \mathsf{0}_3 \\ \tilde{\Pi}_\infty^\top & 1 \end{bmatrix} \tag{2.51}$$

2.6 Conclusion

In this chapter, some basic concepts of projective geometry and transformation were reviewed. These concepts are necessary to describe the image formation process formulating the projection process of a scene into an image. The camera projection matrix was introduced and the epipolar geometry relating multiple views of a scene was discussed. Most importantly, an insight from studying the different geometrical classes and their invariance shows that projective structure can be upgraded to more restricted classes such as affine or metric when certain geometrical entities are identified (e.g. plane at infinity, absolute dual conic). This is of great importance as the main objective in this dissertation is to develop methods to reconstruct the scene by inverting the image projection process.

Chapter 3

Camera Auto-Calibration : Literature Review

3.1 Introduction

By matching image points between two, or more, images of a rigid scene taken from different view points, the three-dimensional representation of these scene's points can be reconstructed. When there is no knowledge about the camera intrinsic and extrinsic parameters, however, such recovered three-dimensional presentation is only up to a projective ambiguity and hence of limited use in solving vision tasks . For instance, relative lengths and angles are no longer preserved and cannot be measured. This ambiguity can be reduced to affine which helps accomplishing a wider range of tasks (e.g. see the work of Hebert et al. [45]). In practice, at least metric representation is required. In such case the relative scene's model pose and size is measurable. In some other more critical vision task the Euclidean measurements is necessary (e.g. robot navigation need to avoid bumping into obstacles).

In order to upgrade a projective representation to metric one, the intrinsic and/or extrinsic parameters must be recovered. Camera calibration in the context of three-dimensional machine vision is the process of determining the intrinsic parameters (i.e. geometric and optical specifications) and the extrinsic parameters (i.e. position and orientation of the camera in the world coordinate system). When these parameters are recovered, the camera is termed "calibrated" [43].

Typically, camera calibration can be done off-line in laboratory setup with very high accuracy. These were the early techniques of photogrammetry and relies on Euclidean (or metric) scene knowledge to infer the intrinsic and extrinsic camera parameters. These early techniques, however, impose great limitation on the practical usage in vision tasks. The presence of a calibration object in the scene is often not possible. In addition, a calibrated camera must maintain its setting fixed; otherwise these parameters will be all lost once the camera adjusts its settings (e.g. focus / zoom) which is often necessary in practice. The advent of auto-calibration and structure from motion, on the other hand, made this possible without previous knowledge of camera parameters. Auto-calibration is concerned with estimating the camera interior and exterior, without the aid of calibration object in the scene, in order to improve the reconstruction to a suitable level for accomplishing the vision task (e.g. affine, metric, Euclidean).

As one of the main contribution of this work is auto-calibration of zooming cameras, the general concept of calibration and the existing literature in this area is reviewed. At first, classical calibration approaches, which requires a certain level of scene knowledge, are briefly reviewed. The bulk of the remaining part of this chapter is dedicated to the subject of auto-calibration of a moving camera observing unknown rigid scene. The majority of the existing approaches are discussed and more details are given when considering topics relevant to the contributions of this work and in particular concerning auto-calibration of zooming cameras.

3.2 Classical calibration

Classical calibration techniques were first developed by photogrammetrists with the goal of obtaining high accurate camera calibration and $3D$ machine vision metrology for aerial imaging and surveying [46]. This is achieved using specially designed calibration devices and camera setup with known Euclidean geometry. The 3×4 camera projection matrix can be computed from the known $3D$ points and their corresponding $2D$ image points. The earliest methods depended on full-scale nonlinear optimization for fitting the $2D$ data of the known Euclidean $3D$ measurements to any arbitrary, yet could be complex, camera model allowing the estimation of lens distortions as well. As non-linear optimization requires a good initial starting point, these methods often start with a simplified linear model, such as Direct Linear Transformation (DLT) of Abdel-Aziz [1] and the later method by Ganapathy [27], before the non-linear refinement takes place.

The main difference between these early methods lies in the type of calibration object and the complexity of the camera model that is used. A detailed review of these methods can be found in the seminal work of Tsai [98] who provided a simple calibration object (known as the Tsai grid) and a reliable two step method for computation. This planar based method has been improved by not restricting it to certain orientation. The most widely used planar based techniques nowadays is the remarkable method proposed by Z. Zhang [105,106]. This method allowed a fast, simple, and stable calibration method to be performed by unskilled general public user where no expensive calibration object is required. A grid pattern printout from a laser printer act as a calibration object with quality good enough for desktop vision systems.

As calibration of zooming camera is a primary concern in this work, classical calibration of zooming camera are discussed below. Willson , in [102] , used an exhaustive approach to model the effect of varying the camera setting on the calibration parameters. The goal was to find a relationship between the change in zoom and focus on the camera intrinsic and extrinsic parameters as a simple function. Controlled by a computer, Willson used a motorized zoom, focus, and aperture camera and measured the camera calibration parameters using Tsai's calibration technique at different configuration. While keeping the aperture fixed, Willson computed the camera parameters at different zoom and focus combination using bivariate polynomials to model the camera intrinsic parameters as a function of zoom and focus. He concluded that there is no simple relation between the camera center, controlled by the motor, and the camera intrinsic variation. Furthermore, by fixing the zoom and focus at specific setting, he tested the effects of changing the aperture on the intrinsic parameters. From this experiment, he noted that changing the aperture does effect some of the intrinsic parameters, but such change has no clear systematic model of relations.

Another similar approach was conducted by Strum [91], who considered self-calibrating a moving camera equipped with a zoom lens. Under pin-hole camera model, Sturm modeled the variation of the five intrinsic parameters subject to zoom. In his model, Sturm showed that the skew angle is close to 90^o degrees and the aspect ratio is almost fixed. As far as the principal point, it has been shown that the principal point position is not stable and varies with the zoom and focus settings. He modeled this variation with a polynomial function that approximates the calibration data. Depending on off-line accurate pre-calibration at different zoom settings, Sturm developed an algorithm which exploits the interdependence of the parameters that needs only simple computation of the roots of univariate polynomials, based on

Kruppa equation, to self-calibrate while moving. The major drawback of this method is the need for a time consuming off-line pre-calibration. In addition, it is not clear if the proposed mechanism will suits other imaging systems other than the one used by Sturm experiment and must be validated experimentally.

3.3 Auto-calibration

Auto-Calibration, or *self-calibration*, is the process of estimating the camera intrinsic and extrinsic parameters without the aid of a calibration object. These methods rely on the usage of point matches between two or more images of an unknown but rigid scene to recover the intrinsic and extrinsic camera parameters. This is convenient in many vision problems whereas the calibration becomes an on-line process (e.g. robotics application).

The theory of auto-calibration was first introduced by Maybank and Faugeras [66]. They showed that locating the absolute conic is equivalent to recovering the intrinsic parameters of the camera. This indicates that there is a virtual calibration object which is present in all scenes. As mentioned in 2.5.3, the absolute conic $\mathbf{\Omega}_\infty$ is a point conic which lies on the infinity plane, thus its relative position and orientation to a moving camera is constant. Under fixed camera settings, the image of the absolute conic projected on the image plane of a moving camera is also constant. As discussed in section 2.5.3, once the image of the absolute conic ω (IAC) or its dual ω^* (DIAC) is identified, it can be used to compute the intrinsic parameters and hence upgrade the reconstruction to metric. As discussed in section 2.5.4, upgrading a projective reconstruction to metric or Euclidian can be attained by the usage of a proper similarity transformation of eight degrees of freedom. For metric reconstruction from uncalibrated images, we seek to find the eight parameters: five parameters corresponding to the calibration matrix $\mathbf{K}$, and the three parameters of the plane at infinity Π_∞. Some auto-calibration techniques solve for those eight parameters directly. Such techniques are, in general, non-linear and encounter problems solving non-linear equations for many parameters at once. On the other hand, a stratified approaches split the computation of these parameters by locating the plane at infinity, i.e. an affine strata, first then recover the other five parameters in a subsequent step. As an advantage, this allows to calculate the remaining five parameters linearly after eliminating the unknown scale factors. However, it is worth mentioning that the hardest step in auto-calibration is actually locating the true plane at infinity [43].

Although the intrinsic and extrinsic parameters are usually unknown, there are some restrictions on them which allows auto-calibration. By exploiting these restrictions, simpler algorithm can be derived. These restrictions can be classified into: restriction on intrinsic parameters and restriction on extrinsic (motion) parameters. More recently, scene knowledge can be employed to constrain camera calibration. However, this is only possible when such constraints do exist in the scene and can be exploited automatically. The earliest auto-calibration methods assumed constant intrinsic parameters. Later, it was shown that it is possible to auto-calibrate from views with some varying intrinsic parameters and thus allowing the camera to zoom. It is important to emphasise that over the last two decades tremendous number of research has been conducted in the context of camera self-calibration, yet there is no simple solution which works for all circumstances. Due to the nature of auto-calibration, it is important to exploit all possible restriction which can be applied in order to achieve a reliable auto-calibration. These restrictions vary according to the vision application under consideration.

3.3.1 Constant intrinsic parameters

Camera auto-calibration methods, using multiple views and assuming constant parameters, were the first to be proposed in the literature. Having a fixed camera setting is equivalent to multiple views from a single moving camera with its setting fixed (i.e., no zooming or focusing). Several methods have been proposed. Below is a list of the most important ones.

Auto-calibration based on Kruppa equations

The first auto-calibration method was due to Maybank and Fougeras [66], and was based on Kruppa equations. Kruppa equations express the relationship that relates the DIAC to the epipolar geometry of a pair of views algebraically. The epipolar geometry of a pair of views, as previously discussed, is encapsulated in the fundamental matrix which can be computed using matched points across two views. Kruppa equations impose that the epipolar plane crossing the pair of camera centers and tangent to the absolute conic must cross the image plane in a line tangent to the image of its dual (i.e. DIAC) as shown in Figure 3.1. There are two such epipolar planes per pair of views and their relation can be related by two independent quadratic equation on the DIAC. Three views at least, taken with constant internal parameters, are required to provide six constraints on the DIAC, which is sufficient to solve for the constant intrinsic parameters.

Unfortunately, Kruppa-based calibration methods exhibit great sensitivity to noise which consequently are not recommended in practice. Increasing the number of views (e.g. five or more) complicates the computations, making the equations impossible to be solved. However, when the focal length is the only unknown, a quadratic expression for focal length in terms of the fundamental matrix can be obtained directly from Kruppa equations [10, 39]. The main special feature about Kruppa equations' based auto-calibration techniques, which could be useful in some cases, is that these methods do not require a set of consistent projection matrices but rather depend only on the epipolar geometry encapsulated in the fundamental matrices for each pair of views. In other words, Kruppa equations do not enforce directly a consistent infinity plane among each pair of views in which the absolute conic must lie on. Perhaps such inconsistency explains why Kruppa based auto-calibration methods performs poorly in comparison with other methods. It is worth mentioning that some variant and improved Kruppa-based auto-calibration methods have been proposed by others over time, see for example [33, 55, 63].

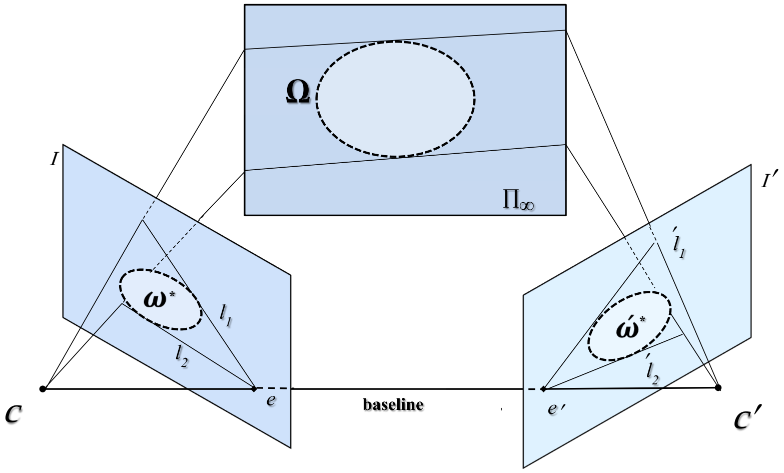


Figure 3.1: *The absolute conic and its images as a calibration device.*

QR-decomposition

Under the assumption of constant intrinsic parameters, Hartley [37] proposed an alternative auto-calibration method which doesn't depend on the absolute conic but rather based on the structure of the projection matrix. Considering an Euclidian transformation matrix $\mathbf{T}$ required to upgrade the camera projection matrices from projective to Euclidian:

$$\mathbf{T} = \begin{bmatrix} \mathbf{K}^{-1} & \mathsf{0}_3 \\ \mathsf{0}_3^\top & 1 \end{bmatrix} \begin{bmatrix} \mathsf{I}_{3\times 3} & \mathsf{0}_3 \\ \tilde{\Pi}_\infty^\top & 1 \end{bmatrix} \tag{3.1}$$

where the $(\tilde{\Pi}_\infty 1)^\top$ is the unknown plane at infinity and the matrix $\mathbf{K}$ represents the constant intrinsic parameters. He derived constraints from the QR-decomposition of the camera projection matrices which in metric frame must yield an upper triangular camera matrix $\mathbf{K}_i$ and an orthogonal rotation matrix $\mathbf{R}_i$ for each camera matrix P_i. The eight unknowns of $\mathbf{K}$ and $\tilde{\Pi}_\infty$ are solved by using a proper non-linear minimization criterion. The minimization process is initialized with an approximate coordinate of the unknown plane at infinity by considering the chirality constraints [36]. Chirality constraints allow to upgrade the projective structure to what is called *quasi-affine* structure. A quasi-affine structure is not a true affine structure but close to it. It avoids splitting the convex hull of the structure across the plane at infinity by simply imposing the fact that all of image points must actually lie in front of the camera. This method encounters convergence problems as it has to solve for many parameters at once. This method has been extended and improved later on for varying intrinsic parameters by first doing an exhaustive search for the true plane at infinity in a bounded space, in which the intrinsic parameters can be computed afterward linearly.

In most of the subsequent auto-calibration works, the elimination of the rotational matrix is derived by implicit multiplication of the rotational matrix by its transpose, instead of explicit QR-decomposition.

Absolute dual quadric

Bill Triggs introduced the dual absolute quadric to computer vision as a convenient way to combine both the absolute conic and its supporting infinity plane in one single geometrical entity [97] . Before reviewing Triggs method, it is worth mentioning that Heyden and Åström proposed an auto-calibration method in [47] and had previously derived similar constraints to the dual absolute quadric, but without providing the geometrical interpretation as was shown by Triggs.

Heyden and Åström started from a projective to metric transformation $\mathbf{T}$ which is required to bring the projective reconstruction to metric. The projective camera matrices $\mathbf{P}_i \doteq \mathbf{K}[\mathbf{R}_i|\mathsf{t}_i]$ can be upgraded to metric by multiplying each camera matrix $\mathbf{P}_i$ by the inverse of the transformation $\mathbf{P}_i\mathbf{T}^{-1}$. By taking the inverse of Hartley' equation in 3.1, and letting the first projective camera matrix $\mathbf{P}_1 = [\mathsf{I}_{3\times3}|\mathsf{0}_{3\times1}]$, the similarity transformation $\mathbf{T}^{-1}$ must be of the form:

$$\mathbf{T}^{-1} \doteq \begin{bmatrix} \mathbf{K} & \mathsf{0}_3 \\ \mathsf{a}^\top & 1 \end{bmatrix} \tag{3.2}$$

where the nonhomogeneous coordinates of the plane at infinity $\tilde{\Pi}_\infty^\top$ are encoded in the three-term vector $\mathsf{a} = -\mathbf{K}\tilde{\Pi}_\infty$. Instead of Hartly's QR-decomposition to eliminate the extrinsic parameters of the projection camera matrices, the author followed a different approach. Starting from the equation:

$$\mathbf{P}_i\mathbf{T}^{-1} \doteq \mathbf{K}[\mathbf{R}_i|\mathsf{t}_i]$$

and by eliminating the last column of the transformation $\mathbf{T}^{-1}$, the author nicely eliminated the three unknown translation vectors t_i such that:

$$\mathbf{P}_i \begin{bmatrix} \mathbf{K} \\ \mathsf{a}^\top \end{bmatrix} \doteq \mathbf{K}\mathbf{R}_i \tag{3.3}$$

Furthermore, they eliminated the rotational matrix by post-multiplying both sides of the equation 3.3 by its transpose.

$$\mathbf{P}_i \begin{bmatrix} \mathbf{K} \\ \mathsf{a}^\top \end{bmatrix} \left[\mathbf{K}^\top|\mathsf{a}\right] \mathbf{P}_i^\top \doteq \mathbf{K}\mathbf{R}_i\mathbf{R}_i^\top\mathbf{K}^\top \tag{3.4}$$

and since $\mathbf{R}_i\mathbf{R}_i^\top$ yeilds the identity matrix $\mathbf{I}$, the equations is simplified to

$$\lambda_i\mathbf{P}_i \begin{bmatrix} \mathbf{K}\mathbf{K}^\top & \mathbf{K}\mathsf{a} \\ \mathsf{a}^\top\mathbf{K} & \mathsf{a}^\top\mathsf{a} \end{bmatrix} \mathbf{P}_i^\top = \mathbf{K}\mathbf{K}^\top \tag{3.5}$$

where λ_i is a non-zero scale factor. This equation is formulated to define an objective function for non-linear optimization to minimize:

$$\mathcal{C}(\mathsf{K}, \mathsf{a}, \lambda_\mathsf{i}) = \sum_{i=2}^{n} \|\mathbf{K}\mathbf{K}^\top - \lambda_i\mathbf{P}_i \begin{bmatrix} \mathbf{K}\mathbf{K}^\top & \mathbf{K}\mathsf{a} \\ \mathsf{a}^\top\mathbf{K} & \mathsf{a}^\top\mathsf{a} \end{bmatrix} \mathbf{P}_i^\top \|^F \tag{3.6}$$

where the expression $\| \dots \|^F$ denotes the Frobenius norm. Each camera provides five equations except for the first one. Thus, a minimum of three views are required to recover the eight unknowns and upgrade to metric. To initialize the non-linear optimization, they suggested guessing the parameters, something that is not always possible in practice. In addition to this disadvantage, the involvement of the additional λ_i unknown scalars for every view causes convergence problem for long sequence of images. At last, the first camera is assumed error free and thus doesn't treat all images equally which may bias the estimation.

These disadvantages have been avoided by using the absolute dual quadric $\mathbf{\Omega}^*_\infty$ introduced by Triggs [97]. The absolute dual quadric is a rank three dual quadric imposed on the infinity plane, where its rim is the absolute conic. The important property of the absolute dual quadric $\mathbf{\Omega}^*_\infty$ is that it combines both affine and Euclidean geometrical entities (i.e. the plane at infinity and the absolute conic) in a single entity which is much easier to use than the absolute conic. Using the absolute dual quadric, the relationship between the absolute conic and its projection in an image is easily obtained using the equation

$$\omega^*_i = \lambda_i \mathbf{P}_i \mathbf{\Omega}^*_\infty \mathbf{P}_i^\top \doteq \mathbf{K}_i \mathbf{K}_i^\top \tag{3.7}$$

This indicates that the projection of the absolute dual quadric of camera i is actually the dual image of the absolute conic which encodes its intrinsic parameters. This is quite similar to Heyden and Åström equation 3.5, where the first camera $\mathbf{P}_1$ is chosen as $[\mathsf{I}_{3\times 3}|\mathsf{0}_3]$ and considering canonical form of the plane at infinity and absolute conic then

$$\mathbf{P}_i \begin{bmatrix} \mathbf{K} \\ \mathsf{a}^\top \end{bmatrix} \left[\mathbf{K}^\top|\mathsf{a}\right] \mathbf{P}_i^\top = \mathbf{P}_i \begin{bmatrix} \mathbf{I} \\ \mathsf{0}^\top \end{bmatrix} \left[\mathbf{I}^\top|\mathsf{0}\right] \mathbf{P}_i^\top = \mathbf{P}_i \begin{bmatrix} \mathbf{I} & \mathsf{0} \\ \mathsf{0}^\top & \mathsf{0} \end{bmatrix} \mathbf{P}_i^\top = \mathbf{P}_i \mathbf{\Omega}^*_\infty \mathbf{P}_i^\top \tag{3.8}$$

In the case of unknown but constant intrinsic parameters, Triggs proposes to use Equation 3.7 to solve for both the absolute dual quadric (10 unknowns) and the dual image of the absolute conic ω^* (5 unknowns), by enforcing the condition that ω^* is the same for all views (i.e. $\omega_i^* = \omega^*$). The scale factors λ_i were eliminated by cross-multiplying the terms of Equation 3.7. Triggs proposed two different methods to solve for the unknowns of $\mathbf{\Omega}_\infty^*$ and ω^*. The first one is a non-linear minimization algorithm which requires three views, while the other one uses a quasi-linear technique and requires at least four views. The non-linear optimization method were reported to be faster and more accurate, but requires an approximate initialization.

Stratified approach & the modulus constraint

Rather than solving for the eight unknowns of the absolute conic and plane at infinity all at once, Pollefeys [73, 74] proposes a stratified approach in which an affine calibration is achieved first by locating the plane at infinity using the so called *modulus constraint*. The unknown intrinsic parameters can be computed through constraints on the dual image of the absolute conic. Under the case of constant camera intrinsic's parameters (i.e. no zooming), the homography induced by the plane at infinity relates a pair of camera matrices, which can be defined as:

$$\mathbf{H}_\infty = \mathbf{KRK}^{-1} \tag{3.9}$$

This indicates that the infinity homography $\mathbf{H}_\infty$ is conjugate to a rotation matrix and thus must have eigenvalues of equal modulus. Taking the scale factor in consideration, the eigenvalues λ_i of such orthogonal matrix are α , $\alpha e^{i\theta}$, $\alpha e^{-i\theta}$, which must satisfy the two constraints $\|\lambda_1\| = \|\lambda_2\|$ and $\|\lambda_3\|^3 = \lambda_1\lambda_2\lambda_3$. This observation were reported first by Luong and Viville in [65], but investigated by Pollefeys who drove a quadratic constraint relating the three unknown parameters of the plane at infinity with the modulus constraint. The earliest method required four views for locating the plane at infinity [75] . The method were enhanced further in [73, 74] to provide more robust result from at least three views by solving set of trivariate quadrics. Geometrically, this is a problem of intersecting three quadratic surfaces and leads to 64 different solutions. Schaffalitzky [84] classified these 64 solutions and reduced the feasible solutions to 21 only. The modulus constraint can be combined with other scene constraints, e.g. vanishing points of parallel lines, to increase robustness or to self-calibrate from two views only. The method were also extended later on to allow calibration with varying focal length as well for a long sequence of images.

Once the plane at infinity is identified, it becomes easy to relate the image of the absolute conics of each view using the infinity homography. Fore example, in the case of constant parameters, Luong and Viville [65] showed that IAC transformation is unchanged using the infinity homograpghy between different views. Algebraically, this can be written as:

$$\omega_i = \mathbf{H}_{\infty}^{-\top} \omega_j \mathbf{H}_{\infty}^{-1} \text{ and } \omega_i^* = \mathbf{H}_{\infty} \omega_j^* \mathbf{H}_{\infty}^{\top} \tag{3.10}$$

where the dual image of the absolute conic $\omega^* = \mathbf{K}\mathbf{K}^\top = \omega^{-1}$.

Equation 3.10 can be used to generate a set of linear equations in the coefficients of ω^* or ω, after enforcing equality of both sides. This was proposed by Hartley [37] to neatly eliminate the unknown scale factors between each pair by scaling each side such that its determinant is unity. Once the IAC or its dual DIAC is computed, it can be refined through a non-linear minimization step using for example, Levenberg-

Marquard algorithm [78].

3.3.2 Varying intrinsic parameters

During the zooming process, the optical center of the camera translates to a new position causing the focal length to vary. The misalignment caused by the mechanical movement of the camera's lenses alters also the principal point position [102]. Such mechanical misalignment is minimal with high quality state-of-the-art cameras, and thus maybe neglected with short focal lengthes adjustment [28]. However, for other common cameras, the change in principal point position due to zooming cannot be ignored by assuming fixed principal point position as the auto-calibration algorithms are highly sensitive [42]. This indicates, in general, that there are at least three varying parameters out of the five unknown intrinsic parameters, as it is often safe to assume constant aspect ratio and skew. The latter depend on the pixel sensors shape and are unaffected by focus and zoom changes. Based on this fact, the problem of auto-calibration from images taken by cameras with different settings, or equivalently from a single camera undergoing a general motion while adjusting its settings, cannot be solved by the previous auto-calibration techniques that assume constant parameters. This limitation is impractical for many vision tasks and even for a sequence of images taken with a single camera as auto-focusing is performed in many circumstances. If all intrinsic parameters are allowed to vary, auto-calibration is not feasible. Fortunately, it was proven that auto-calibration from views of cameras with varying parameters is possible, if at least one parameter is kept constant, but maybe unknown, along a set of views [50].

In early stages and under the assumption of known skew, aspect ratio and principal point, Hartley [35] used a decomposition of the fundamental matrix to find the varying focal lengths and the relative positions of a pair of cameras. The method is based on the fundamental matrix and can be computed linearly using singular value decomposition. However, from a pair of cameras, such approach can recover at most two parameters (five related to extrinsic parameters out of the seven degrees of freedom of the fundamental matrix leaves only two).

With the advances in camera manufacturing, it is often safe to assume that the cameras have square or rectangular pixel sensors. This assumption sets the skew parameter to zero and makes the aspect ratio constant. Under these conditions, Heyden and Åström [48] have proven that it is possible to auto-calibrate a camera with square pixels, allowing the focal length and the principal point to vary freely. This proof was extended in [49, 50] to show that auto-calibration is theoretically possible if at least one single parameter is fixed, but may be unknown, among the whole sequence of views. A non-linear minimization using bundle adjustment technique was proposed. The method requires to run simultaneously over all reconstructed cameras and points. Beside the obvious difficulty dealing with non-linear minimization for many parameters, this method did not address the problem of obtaining a suitable initial estimation, required to properly initialize the non-linear iterative minimization process. Hence, convergence remains a serious problem for this method.

Another independent work by Pollefeys [72] has extended Heyden and Åström proof, showing that auto-calibration is possible under the assumption of rectangular pixels (i.e. zero skew only). An accounting argument was also derived to calculate the minimum number of views required for auto-calibration under any assumption on camera parameters. For n views under general motion, the minimum number of views required to auto-calibrate must satisfy

$$n \times (n_{known}) + (n - 1) \times (n_{constant}) \geqslant 8 \tag{3.11}$$

where, n_{known} and $n_{constant}$ are the number of known and the number of constant (nonchanging) intrinsic parameters, respectively.

This equation shows that each known intrinsic parameter provides n constraints and each constant intrinsic parameter provides $(n - 1)$ constraints. For example, under non-degenerate camera motion, it was shown that if the skew is fixed (does not change), auto-calibration is possible from at least eight views while only four views are required to calibrate under varying focal and principal point but known zero skew and aspect ratio case. A pair of views are sufficient to recover the single varying focal length parameters if the other four parameters are fixed. The significant advantage of Pollefeys' method is that it provides a simple linear method to obtain a close estimation of the parameters, a problem which was not addressed by Heyden and Åström. By assuming that the principal point is known (e.g. assumed to be at the image center) and square or rectangular pixels, linear constraints on the absolute dual quadric can be obtained. These constraints are used to compute an initial estimate of the intrinsic parameters, required to initialize a non-linear minimization process where the focal length and principal points are allowed to vary. However, it was shown in several works that the auto-calibration problem is sensitive to inaccurate localization of the principal point (See for example the work of Hartley [42]).

Within the same vein of ADC, a linear-iterative algorithm was proposed by Seo et al. [88]. The algorithm initially estimate the ADC in a similar way as Pollefeys [72] by normalizing the image coordinates so that the image center is shifted to the initial principal point position. Geometrically, this is equivalent to obtaining three orthogonal vectors and thus three constraints on the ADQ can be obtained, thus allowing the computation of the ADQ from three views or more. This initial step is followed by enforcing rank 3 constraint on the ADQ and re-estimating the variation of the estimated principal point with the initially assumed position. The algorithm then iterates until a stop criterion is satisfied.

A stratified approach was also investigated for calibrating varying intrinsic parameters. Starting from a projective reconstruction, the three unknown parameters of the plane at infinity need to be estimated in order to upgrade the structure to affine and compute the infinity homography. Once the infinity homography is computed, linear constraints on the image of the absolute conic can be transferred across the different views. This indicates that, once the plane at infinity is located, liner upgrade to metric is possible when enough number of images are provided. However, under general camera motion and varying intrinsic parameters, estimating the plane at infinity is a highly non-linear problem. Hartley [41] used chirality inequalities to upgrade the projective camera matrices and structure to quasi-affine in order to bound the location of the plane at infinity. An exhaustive search for the plane at infinity coefficients within the bounded region of the parameter space is performed. Nister [69] in his turn improved the quasi-affine structure computed with chirality inequalities by seeking a reference plane which doesn't split the camera's centers. Qualitative comparison of the methods shows that such obtained quasi-affine reconstruction highly improves the chances of the subsequent non-linear auto-calibration method to converge correctly. It is worth noting that, none of the current approaches provide a linear estimation of the plane at infinity under general motion. As will be discussed in next subsections, affine structure can be *linearly* computed but with the aid of either special motions (e.g. pure translation or pure rotation) or with the aid of scene constraints (e.g. vanishing points).

Recently, several algorithms, assuming cameras with square pixels, have been proposed, with the advantage of being linear. These algorithms are based on different geometrical entities, similar to the previously discussed absolute dual quadric (ADQ), which encodes the absolute conic. In this context, the *Absolute Quadric Complex* (AQC) was proposed by [77]. Under the square pixels restriction and projective camera matrices, two orthogonal lines can be identified which must intersect the absolute conic by means of a quadric in the higher dimensionality space of $\mathbb{P}^5$. This is related to the nature of representing lines in $\mathbb{P}^3$ which is awkward as it requires 6 homogenous terms using Plucker lines (one can refer to the book of Hartley and Zisserman [43]). The AQC is represented by a symmetric 6×6 matrix and thus requires 21 parameters which can be reduced to 19 nonhomogeneous parameters. As each camera provides two constraints from the two orthogonal lines, this explains why such methods based on AQC require at least 10 cameras to be computed. This is a drawback as it does not comply with the theoretical minimal requirements for metric reconstruction in which only 8 unknown parameters required to parameterize the projective to metric upgrade transformation which has 15 degrees of freedom. Despite their linear advantage, such methods are limited in practice due to the requirement of at least 10 cameras, to generate the minimal number of equations to computate the AQC. This number could be insufficient, as many views might fall under or close to the critical motion configuration. Ronda et al. reformulated the AQC showing new properties which allowed them to obtain an enhanced auto-calibration algorithm [80]. The new results yielded closed-form expressions for the intrinsic parameters of the camera, including skew-angle, aspect ratio and, the principal point position, in terms of the AQC. An algorithm was proposed to extract the ADQ from the AQC using simple matrix operations which can be refined using bundle-adjustment, or by a newly proposed algorithm based on minimizing the error in pixel shape. The latter was reported to produce slightly better results with lower computational cost. Just recently, [81], enhanced their previous algorithm by reducing the number of required cameras from 10 to the theoretically minimum of 5. This is achieved by introducing a new geometrical tool, called the Six-Line Conic Variety (SLCV). However, the proposed algorithm is non-linear and requires a bidimensional search using second degree equations.

3.3.3 Auto-calibration from special motions

Auto-calibration can take advantage of certain camera motions to simplify the problem by reducing the number of ambiguities. Some restricted motions may naturally arise in practice as pure translation, pure rotation, and planar motion. However, certain types of motion may fall under the category of *critical motion*, where it is not possible to obtain metric calibration.

Pure translation

Pure translation refers to a translating camera while the intrinsic parameters remain constant. This is equivalent to a single stationary camera obtaining images while the scene is translating. Under pure translation, affine reconstruction can be obtained instantly. This was demonstrated by Moons. et al. [68] who showed that by superimposing a pair of images on each other, a pair of matched points is enough to compute the epipole e. Affine camera matrices from a pair of images under pure translation can be instantly obtained as:

$$\mathbf{P}_1 = [\mathsf{I}_{3\times3}|\mathsf{0}] \text{ and } \mathbf{P}_2 = [\mathsf{I}_{3\times3}|\mathsf{e}]$$

However, as there is no rotation, no constraints on the intrinsic parameters can be obtained. In addition, affine calibration fails if the camera's parameters vary during translation [52]. In the situation where the Euclidean/relative translation along the different axes is known, the relative depth of the points can be recovered as shown in [51]. From several known pure translations, the Euclidian reconstruction and camera calibration of an unknown scene can be obtained linearly as shown in [70]. In fact, this is equivalent to classical calibration from a single view of a scene with known Euclidian geometry.

In a stratified approach, the requirement of pure translation was used to obtain the affine structure first, followed by one or more rotation to obtain metric structure [3]. Under the assumption of known principal point, Pollefeys et al. extended this method to allow self-calibration with the flexibility of varying the focal length [76]. In fact, an initial pure translation step allows computing the infinity homograpghy and thus it is possible to transfer constraints on the image of the absolute conic between views despite varying some, but not all, intrinsic parameters in subsequent views.

Pure rotation

Pure rotation refers to images taken from a stationary camera while rotating around its optical center (see Figure 3.2). As the camera remains stationary, images of the same feature point in two such images are related by $\mathsf{q}_j = \mathbf{H}_{ij}\mathsf{q}_i$, where $\mathbf{H}_{ij}$ is the 3×3 matrix is the infinity homography relating the image point q in the i^{th} & j^{th} pair of images, with $\mathbf{H}_{ij} = \mathbf{K}_j\mathbf{R}_{ij}\mathbf{K}_i^{-1}$. To clarify this, recall that the i^{th} projection matrix

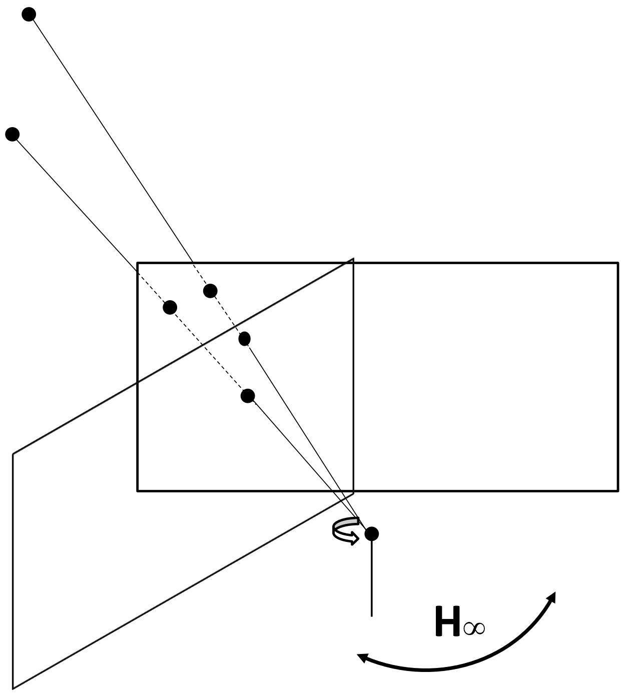


Figure 3.2: *Two images acquired by a rotating camera around its optical center. The infinity homography maps image points between the two images.*

can be represented by $\mathbf{P}_i = \mathbf{K}[\mathbf{R}_i|\mathsf{t}_i]$. Since the camera optical center remains fixed during rotation, this implies that $\mathsf{t}_i = \mathsf{0}$ for all images and thus the projection matrix can be simplified to $\mathbf{P}_i = \mathbf{K}\mathbf{R}_i$. The 3×3 infinity homography can be computed from four or more point matches only. Once the set of homographies $\mathbf{H}_{ij}$ is computed, the dual image of the absolute conic ω^* in a pair of images can be related by the equation:

$$\omega^* = \mathbf{K}_i\mathbf{K}_i^\top = \mathbf{H}_{ij}^\top\mathbf{K}_j\mathbf{K}_j^\top\mathbf{H}_{ij} \tag{3.12}$$

This fact was pointed out by Hartely [40] who wrote liner constraints on the fixed camera parameters (i.e. $\mathbf{K}_i = \mathbf{K}_j = \mathbf{K}$). By using the infinity homography to transfer these constraints across images, it becomes possible to linearly compute ω^* and, hence the calibration matrix $\mathbf{K}$ can be found using Cholesky factorization.

Varying focal length can be linearly estimated in the case of rotating camera with known skew and principal point [89], whereas a non-linear solution was suggested by [17].

As $\omega^* = \omega^{-1}$, by taking the inverse of both sides of Equation (3.12), [16] related the image of the absolute conic between the pairs of views such that:

$$\omega_i = \mathbf{K}_i^{-\top}\mathbf{K}_i^{-1} = \mathbf{H}_{ij}^{-\top}\omega_j\mathbf{H}_{ij}^{-1} \tag{3.13}$$

This allows to obtain linear equations under various possible restrictions including the zero skew, known aspect ratio, and/or know principal point. The linear method has the advantage of being very fast, does not necessitate initialization, and most often provide a solution which can be refined non-linearly. However, it may also fail if the IAC obtained is not *positive definite* which happen in cases of high noise levels, ill-conditioned configurations camera, and critical or near-critical rotational motions. De Agapito et al. also proposed in [18] a non-linear optimal Maximum Likelihood (ML) estimator for the calibration matrices and the motion parameters by performing a final bundle-adjustment. The advantage of the non-linear method lies in its ability to directly parameterize any available constraints on the intrinsic parameters. The non-linear method, usually initialized with the linear method, occasionally fail to converge in ill-conditioned sequences and more often if the principal point is allowed to vary due to zooming.

Rameau et al. [79] proposed a method which employs a Linear Matrix Inequality (LMI) resolution approach for self-calibrating Pan-Tilt-Zoom (PTZ) cameras. The algorithm has provided significant improvement in accuracy and robustness. Using LMI allows incorporating extra constraints on the intrinsic parameters which can be tuned during the estimation process of the intrinsic parameters. As an advantage, the considered constraints are enforced for all views rather than the normal technique which consider recovering the first camera's parameters from which the remaining ones can be recovered.

It is important to emphasize a couple of practical issues related to auto-calibration of rotating cameras. Such rotational motion may arise naturally in many scenarios such as PTZ cameras used in video-conference and surveillance systems, and thus can be considered practical. The main advantage of auto-calibration with cameras rotating around their optical axes, is the availability of simplified linear algorithms and robust feature matching between one-to-one images instead of one-to-many. On the other hand, there are some limitations which need to be highlighted. Although auto-calibration is possible from rotating cameras, nevertheless metric reconstruction from a single rotating camera is not. Moreover, the assumption of pure rotation around the exact optical axis is violated in practice due to misalignment, especially in the case of zooming camera. This may lead to significant errors, threatening its success for indoor applications where the distance to the scene is not large in comparison with the translation of the optical center of the rotational axes. This fact has been confirmed by several authors where, detailed and most comprehensive studies on the misalignment of the optical center and the rotational axes can be found in [44,54,90].

3.3.4 Auto-calibration from scene constraints

In addition to the previous set of restrictions on the intrinsic parameters and/or on the camera motions, scene constraints can be also intergraded in the auto-calibration algorithms. Man-made environments are rich sources of geometrical primitives, and thus can aid in many vision application. For off-line applications and with little human interaction, such knowledge can be identified easily. Incorporating scene knowledge within the auto-calibration framework is of several benefits as it can simplify auto-calibration process, elevate robustness, provides extra constrains to be used in conjunction with other intrinsic and motion constrains, as well as enhancing the recovered 3D models quality. Such knowledge can be information of basic scene primitives such as points, lines, and planes or higher level geometrical scene objects such as circles, cubes, prisms, cylinders, etc. Regardless of the obtained type of scene information, all can be incorporated in the auto-calibration process or the subsequent reconstruction bundle-adjustment.

Instead of using a calibration object of known Euclidean geometry, metric knowledge, such as relative distances or angles in the scene, can be used to obtain metric structure. For these methods, an initial projective reconstruction of the scene points Q^p can be computed by tracking and matching points in two or more images. A second step is used to upgrade this projective structure to metric (or Euclidean) Q^m, by finding an appropriate 4×4 transformation matrix $\mathbf{T}$:

$$\mathsf{Q}^m \doteq \mathbf{T}\mathsf{Q}^p \tag{3.14}$$

If the exact position of some scene points are known, then the transformation matrix $\mathbf{T}$ can be easily computed and the projective structure can be upgraded to Euclidean. This is similar to classical calibration method using a calibration object. However, instead of having a calibration object with high precision known geometry in the scene, one can use other general Euclidean or relative metric information for calibration. Boufama et al. [8] showed that various Euclidean scene knowledge can be incorporated as constraints to upgrade the projective representation to metric or

Euclidean. He derived constraints from various geometrical properties such as coplanar points (e.g. on the ground plane), points which are vertically aligned, as well as known or equal distances between points. Liebowitz and Zisserman [60] investigated the usage of weak metric planar information such as known length ratios, known angles or two equal but unknown angles. Caprile and Torre exploited the usage of three orthogonal vanishing points to allow computing the camera intrinsic parameters [12]. Such vanishing points, for example, can be computed from the orthogonal sides of a building. [7] investigated using scene constraints such as orthogonality, parallelism during the calibration procedure. In fact, many other scene knowledge has been exploited and can be used as constraints to restrict the projective structure to metric or Euclidian such as parallel lines, orthogonal planes, circles, etc. The ability to detect such scene constraints in the scene allows automating the calibration process or enhancing its quality.

It is worth noting, however, that incorporating scene knowledge relying on human interaction is equivalent to classical camera calibration, i.e. using calibration object, but with the advantage of not requiring to place the calibration object in the scene. Such approach is useful for many human-guided application such as off-line 3D modeling from uncalibrated image sequence. On the other hand, for online application from uncalibrated image sequences, reliable and automatic identification of such information remains a hard problem. The following discussion is limited to auto-calibration methods based on scene constraints which can be identified automatically from uncalibrated images.

Schaffalitzky and Zisserman proposed a method for automatic detection and grouping of image elements which are repeated on a scene plane. Such elements can be used for estimating vanishing points and vanishing lines [85, 86]. The authors addressed three classes of commonly occurring types of geometric primitives: (1) equally spaced coplanar parallel lines; (2) a planar pattern with repeated elements by translation in the plane; and (3) a set of elements arranged in a regular planar grid. Recovering vanishing points and vanishing lines allows the recovery of the plane at infinity. This allows the computation of the infinity homography, which enables linear metric upgrade under the assumption of restriction on some intrinsic parameters.

Lines and points are the basic and common geometrical primitives which can be identified and matched robustly across multiple images. Aminitabar and Boufama [2] proposed an algorithm to detect *scene planes* from uncalibrated images, a challenging problem that often leads to extracting large number of undesirable virtual planes. The proposed algorithm is based on homography calculation between three or more point matches from two images. It classifies the estimated planes into virtual and physical scene planes by detecting non-coplanar points inside the convex hull of the group of point used in the homography estimation. They provided different confidence levels for extracted planes classifying them gradually from most likely virtual planes to most likely physical ones.

The relation between a pair of parallel planes can be very useful for reducing the projective ambiguity of the reconstruction. In particular, parallelism is an affine invariant feature, and thus if identified can be employed to reduce the projective ambiguity to affine. Similarly, orthogonality is a metric invariant property and a pair of orthogonal lines, or planes, help obtaining metric reconstruction.

Using a set of extracted planes from a pair of uncalibrated images, Habed et al. proposed a method to distinguish and identify parallel pairs [2]. Relying on the fact that parallel planes intersect at infinity, a linear relationship between the inter-image homographies of the parallel planes and the plane at infinity is devised. Under the assumption of constant camera intrinsic parameters, they combined this relationship with the modulus constraint for parallel planes identification. Detecting parallel planes allows identifying vanishing lines, i.e. the intersection of two parallel planes at infinity, which places two constraints on the three unknown terms of the plane at infinity and thus with two pairs of parallel planes an affine calibration can be obtained linearly [31]. In chapter (6), a similar method for parallel planes identification is presented. However, the proposed method is relaxed from the restriction of constant camera parameters.

Another important and very helpful relationship between a pair of planes is the perpendicularity which is a metric invariant. The angle θ between any two planes Φ and Ψ can be computed using the absolute dual quadric (Ω^*_∞) as:

$$\cos(\theta) = \frac{\boldsymbol{\Phi}^\top \Omega^*_\infty \boldsymbol{\Psi}}{\sqrt{(\boldsymbol{\Phi}^\top \Omega^*_\infty \boldsymbol{\Phi}).(\boldsymbol{\Psi}^\top \Omega^*_\infty \boldsymbol{\Psi})}} \tag{3.15}$$

Considering orthogonal pair of planes (i.e. $\theta = 90^o$) and by ignoring the denominator of equation (3.15), the relationship between a pair of orthogonal planes simplifies to:

$$\boldsymbol{\Phi}^\top \Omega^*_\infty \boldsymbol{\Psi} = 0 \tag{3.16}$$

Benefiting from the linear nature of this equation (3.16), Huynh and Heyden proposed a scheme for incorporating orthogonal scene planes in the framework of camera auto-calibration [53]. Imposing orthogonal scene planes in the auto-calibration and reconstruction provides extra constraints that makes auto-calibration from fewer number of images possible and, help obtaining better 3D reconstruction quality. Incorporating the scene orthogonal planes constraints can be at the initial step of estimating the absolute dual quadric (Ω^*_∞) as well as the subsequent bundle adjustment refinement step. However, no method proposed for automatic identification of orthogonal scene planes and thus the proposed framework is limited to human-guided 3D modeling applications.

Chapter 4

Affine Auto-Calibration and 3D-Reconstruction

Three-dimensional reconstruction of a scene from two or more images is of significant importance for many computer vision applications. An initial projective point-wise structure can always be recovered from feature correspondences tracked through an uncalibrated image sequence [25,26,82,93]. Such projective structure is often of limited use for the majority of computer vision problems. The initial projective structure, however, can be upgraded to a more specialized one, i.e. metric or Euclidian, once the camera's intrinsic and/or extrinsic parameters are known. In the case of frequently zooming cameras, these parameters are continuously changing and thus need to be recalibrated. The three-dimensional Euclidian structure problem of a "possibly" zooming and moving camera is nonlinear and challenging to solve [5,48,69,81,99,104]. The reason, these methods seek recovering many unknown parameters, for each camera, directly in a single step. Stratified auto-calibration methods, on the other hand, simplify this problem by first obtaining an affine calibration and structure, from which linear metric/Euclidean calibration and structure upgrade can be followed. The difficult step, however, is to precisely locate the plane at infinity with no prior knowledge about the scene and is the primary contribution of this chapter.

The scaled Euclidean structure, i.e. metric, provides the ultimate source of information that vision tasks strive to obtain and will be the topic of chapter 5. On the other hand, affine structures have ample amount of information for many vision applications. An affine reconstruction preserves parallelism of lines and planes, the ratios of lengths of parallel line segments, as well as ratios of areas on parallel planes. For example, using affine prosperities and structure, Beardsley et al. [6] proposed navigation method and Criminisi et al. [14] showed how relative people's height can be measured from affine structure.

This chapter provides a linear method to affine auto-calibrate a pair of stationary zooming cameras with unknown translation and orientation between them. Affine calibration is equivalent to locating the plane at infinity. Once the latter is located, the projective camera matrices and the projective three-dimensional structure of the scene can be upgraded to affine.

All techniques for locating the plane at infinity depend on restrictions on the camera intrinsic parameters, special camera motion, or scene constraints. In the case of a moving camera with constant parameters, the modulus constraints [74] can be used to recover the plane at infinity by solving a set of nonlinear polynomial equations. In addition to the inherent difficulty of solving nonlinear equations and the multiple possible solutions, these constraints cannot be used when the camera parameters are allowed to change its setting by zooming. To overcome this limitation, the problem has often been simplified by making unrealistic assumptions on the rigidity of the principal point [72].

Simple linear affine calibration and estimation of the plane at infinity can be achieved in situation of restricted camera motion. These motions are generally pure translation [34, 51, 67, 68, 83]. Pure translating camera refers to a moving camera without rotation while keeping the intrinsic parameters fixed. However, for images taken by different and unknown intrinsic's parameters these algorithms will fail [52, 56]. This is the situation when the camera zooms during the motion causing the focal length and principal point to vary. Other methods, such as parallel screw axis or planar motion, are also considered in the literature [4,22,23,58]. As far as stationary cameras are concerned, the assumption of a mandatory pure rotation of the camera has been proposed in the literature [18, 40]. As previously discussed in chapter 3, pure rotations allow for a linear calculation of inter-image homographies induced by the plane at infinity from which the camera parameters can be retrieved linearly. However, moving cameras in a pure rotation motion is not feasible in practice and such an assumption is only plausible when the camera is far from the scene [44,54,90]. Moreover, the 3D structure of the scene cannot be recovered from a single rotating camera, even if the intrinsic parameters are known. Hence, at least one additional image, captured from a different position in space, is needed. Other approaches for locating the plane at infinity are based on scene constraints. Identifying scene's parallel lines or planes helps estimating *vanishing points*and *vanishing lines*, thus allowing estimating the plane at infinity. However, such methods are limited to scene's where such parallel geometrical primitives do exist and can be automatically identified.

This chapter addresses the problem of affine auto-calibration of an imaging system comprised of two stationary zooming cameras located at distinct unknown positions and orientations in space. The case of stationary zooming but non-rotating cameras has not been addressed in the literature. This is a typical configuration in which each zooming camera is physically attached to a static structure (wall, ceiling or tripod) often encountered in stereo camera systems, surveillance networks and monitoring of all sorts of events. Because the cameras are not rotating, the methods designed for stationary rotating cameras cannot be employed to self-calibrate each camera independently. Solutions designed for moving cameras make no distinction between images taken by stationary cameras and those which are not, leading to unnecessarily complicated nonlinear equations.

Our approach fundamentally differs from all existing self-calibration approaches as it locates the plane at infinity by exploiting the very fact that a camera has zoomed. Indeed, all existing methods, whether dealing with the case of a stationary or moving camera, do not exploit zooming as the camera may or may not have done so. Our proposed method is based on some important observations we have made on the results of the experiments conducted by Willson in his work on designing an active model for zoom lenses [102]. Indeed, the change that affects the intrinsic parameters of a camera while zooming is the result of the displacement of both its optical center and image plane which may possibly undergo a mostly partial rotation. In the existing methods that deal with a moving zooming camera, these changes are absorbed by the rigid motion between the views and hence cannot be exploited independently to support the self-calibration process. In the case of a stationary rotating and zooming camera, these changes have not been exploited but rather neglected. In contrast to all these methods, our affine self-calibration relies specifically on the motion of the optical center and the image plane of the camera on which mild and valid constraints, verified in [102] on several cameras, are imposed .

This chapter is organized as follows: Section 4.1 presents some necessary background and preliminaries. In Section 4.2, we describe the zooming camera model that we have considered for developing our method and its relationship to the plane at infinity. A simple linear affine auto-calibration method is then described, which constitutes the main contribution of this chapter. Obtained experiments and results are described and discussed in Section 4.3. Section 4.4 concludes this chapter.

4.1 Background and preliminaries

Consider a static scene observed by a (stereo) pair of uncalibrated stationary non-rotating but zooming cameras. The two cameras are placed at distinct positions in space and have different orientations. We assume throughout that each camera i ($i = 1, 2$) captures images at two distinct settings in the subset $\mathcal{S}_i = \{1, 2\}$ of possible zooming configurations of its lens. Neither the cameras nor the scene are physically displaced or rotated between the shots. However, because the geometry of a camera changes under zooming effect, we assume throughout that pairs of images captured by the same camera with two distinct zoom settings as if they had been captured by two distinct cameras each of which following the well-known pinhole model.

4.1.1 Projective scene and cameras

At any given zoom setting $s \in \mathcal{S}_i$, a camera i maps any world point Q onto the image point $q_{i,s}$. Expressing world and image points by their homogeneous coordinates, this mapping is described up to a scale (hence $\sim$) through a 3×4 projection matrix $\mathsf{P_{i,s}}$ as follows:

$$\mathsf{q_{i,s}} \sim \mathsf{P_{i,s}Q}. \tag{4.1}$$

We assume throughout that all four projection matrices $\mathsf{P_{i,s}}$ have been calculated from point correspondences with respect to a common projective reference frame. Note that, although we are dealing with stationary cameras, such matrices can always be calculated. Indeed, since the two physical cameras are located at different positions and orientations in space, a projective structure of the scene can be triangulated from two images - one from each camera - while the remaining projection matrices, also consistent with the chosen frame, can be calculated by back-projection on the two other images. In practice, a projective set of projection matrices can be obtained using virtually any off-the-shelf method [43]. In the present work, we have used Rothwell's linear method [82] to do so. The matrices thus obtained allow only for the recovery of the scene and cameras up to common but unknown projective ambiguity. This ambiguity can be reduced to an affine one by means of an adequate transformation represented by a regular 4 × 4 matrix

$$\mathsf{H} \sim \begin{bmatrix} \mathsf{P} \\ \mathsf{\Pi}_{\infty}^{\mathsf{T}} \end{bmatrix} \tag{4.2}$$

obtained by stacking some arbitrary 3×4 matrix P (generally one of projective projection matrices) and a row 4-vector $\mathsf{\Pi}_{\infty}^{\mathsf{T}}$ representing the generally unknown coordinates of the plane at infinity in the current projective frame. This transformation, which restores parallelism in the estimated structure, maps every scene point Q to its new location $\hat{\mathsf{Q}} \sim \mathsf{HQ}$ and turns the projection matrices into $\hat{\mathsf{P}}_{\mathsf{i,s}} \sim \mathsf{P}_{\mathsf{i,s}}\mathsf{H}^{-1}$.

4.1.2 Camera matrix and world planes

The rows of a 3×4 projective camera matrix P are 4-vectors representing the homogeneous coordinates of three planes Π, Ψ and Φ. These planes can be inferred geometrically as specific world planes, depicted in Figure 4.1, and intersecting in the camera center C.

$$\mathsf{P} \sim \begin{bmatrix} \Pi^\intercal \\ \Psi^\intercal \\ \Phi^\intercal \end{bmatrix}. \tag{4.3}$$

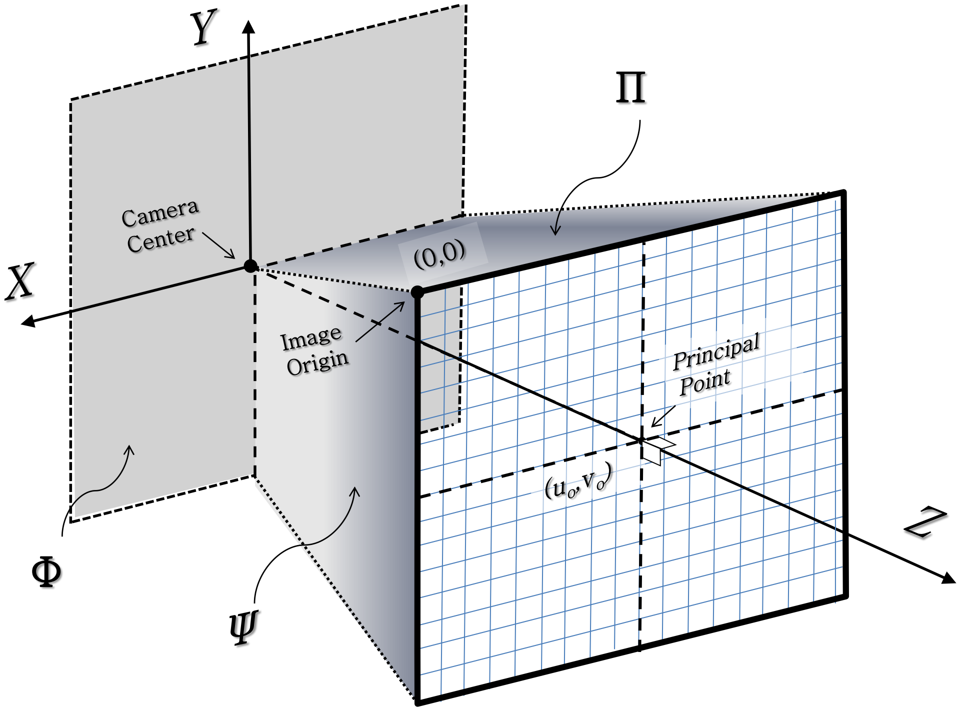


Figure 4.1: *The three world planes defined by the three rows of the camera matrix.*

The plane with coordinates Π is the plane passing through the camera center and the image's vertical axis, i.e. the line $u = 0$. In this manner, a 3D point Q on the plane Π satisfies $\Pi^\intercal Q = 0$ and, hence, is projected onto an image point whose coordinate vector is of the form $PQ \sim (0, v, w)^\intercal$. Similarly, the plane with coordinates Ψ is the one containing the camera center and passing through the image's horizontal axis, i.e. the line $v = 0$. Hence, a point Q on the plane Ψ satisfies $\Psi^\top Q = 0$ and projects onto an image point $PQ \sim (u, 0, w)^\intercal$. In particular, the plane Φ, represented by the 3rd row of P, is the principal plane [43][1]. The principal plane is the plane containing the X and Y axes of the camera's reference frame, hence parallel to the image plane $\mathcal{I}$ and containing the camera center. It is the plane of equation $\mathsf{\Phi}^{\mathsf{T}}\mathsf{Q} = 0$ representing the set of all points Q projected onto image points with coordinates $\mathsf{PQ} \sim (\mathsf{u}, \mathsf{v}, 0)^{\mathsf{T}}$, i.e. points at infinity on the image plane. The method proposed in this chapter exploits the motion of the principal plane that accompanies the displacement of the camera center under zooming effect.

4.1.3 Parallelism and the plane at infinity

It is well-known that parallelism is invariant under affine (hence metric) transformations. This property is often exploited to locate the plane at infinity by detecting and establishing correspondences of *vanishing points* or *vanishing lines* across images. The plane at infinity can be computed from three such vanishing points [12] or from a single vanishing point and a vanishing line [92]. The most general way of locating vanishing points is by determining the intersection point of the images of lines that are parallel in the scene. Furthermore, as parallel planes intersect in a vanishing line, the latter can be located by reconstructing these planes in some projective frame and back-projecting their common line onto the images. It has recently been shown that the plane at infinity can also be located from scenes with two pairs of parallel planes determining two vanishing lines without the need for reconstruction [31]. This is achieved through a linear relationship between parallel scene planes and the plane at infinity. To briefly describe this, consider a 3D scene consisting of two distinct and parallel scene planes Π_1 and Π_2. Since these two planes are parallel to each other, they meet in a line on the plane at infinity Π_∞. As a consequence, the coordinates of

[1] The term focal plane is also used in the literature.

the plane at infinity and those of Π_1 and Π_2 are linearly dependent. Such dependency can be expressed by

$$\mathsf{\Pi}_\infty \sim \alpha_1 \mathsf{\Pi}_1 + \alpha_2 \mathsf{\Pi}_2 \tag{4.4}$$

where α_1 and α_2 are non-zero scalars and $\mathsf{\Pi}_1$ and $\mathsf{\Pi}_2$ are the homogeneous coordinate vectors of the planes Π_1 and Π_2, respectively.

4.2 Zoom-based affine auto-calibration

In this section, we present and describe an affine auto-calibration method for a stereo pair of stationary non-rotating zooming cameras. We discuss the effect of zooming on the camera model, which is our main ingredients used in our method for locating the plane at infinity. It will be shown that the principal planes corresponding to distinct zoom settings of a stationary camera, are parallel to one another. A linear method for calculating the plane at infinity is presented.

4.2.1 The effect of zooming on the camera model

Consider a camera that is physically fixed in space, e.g. on a tripod, capturing two or more images at different settings of its zoom lens. At any given setting $s \in \mathcal{S}_i$ of its zoom lens, a camera i is described by its image plane $\mathcal{I}_{i,s}$ and by its optical center $C_{i,s}$ (see Figure 4.2). The optical center $C_{i,s}$, in which all light rays emanating from the scene intersect, is located at a focal distance $\mathsf{f}_{\mathsf{i},\mathsf{s}}$ from the image plane, along the optical axis of the camera. The latter is perpendicular to and intersects $\mathcal{I}_{i,s}$ in the principal point $c_{i,s}$.

Under the effect of zooming, the optical center $C_{i,s}$ undergoes a displacement to

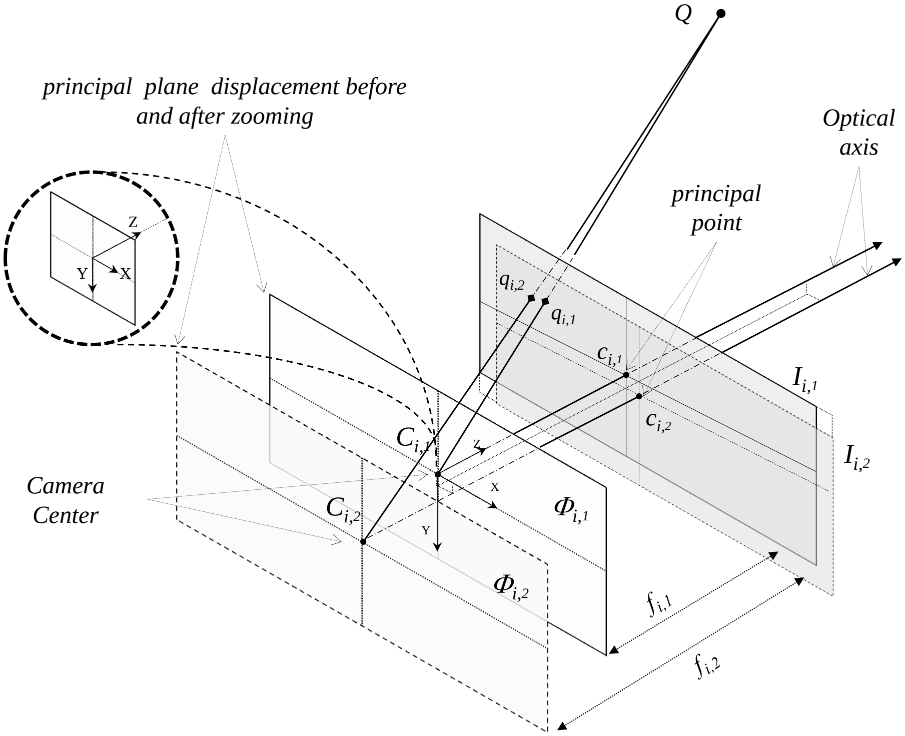


Figure 4.2: *Zooming camera model*

a new location $C_{i,s'}$ at a focal distance $\mathsf{f}_{i,s'}$ from the image plane. This repositioning of the lens, carried out by automated zooming hardware or by manual lens change, does not affect only the focal length of the pinhole camera model, but also other parameters. In fact, a change in the configuration of a camera lens due to zooming results in the repositioning of both the optical center and the image plane. In his design of an active model for zoom lenses [102], Willson has carried out a series of experiments in which a pattern-based calibration of a stationary camera is repeated at various zoom settings using several zoom lenses. This was to identify the camera parameters that must be allowed to vary with the zoom versus the ones that can be fixed in the zooming camera model. The results of Willson's experiments show that the optical center is dominantly shifted (possibly by several tens of millimeters) along the $Z-$axis of the camera (the optical axis) and that its displacements along the $X-$ and $Y-$ axes are small and hence neglected in his model. This does not imply that the position of the principal point remains stable since the image plane is also displaced under the effect of zooming. In particular, the image plane was found to undergo a mostly translational motion which is not necessarily parallel to the $Z-$axis and hence affecting the position of the principal point. In Willson's model, the displacement of the image plane is represented by the fact that both the optical center and the focal length are allowed to vary independently from one another and by allowing the principal point to shift as well. Note that the goal in Willson's work was to obtain a simple model involving only the most influential parameters.

4.2.2 Parallel principal planes

Based on the previous description, the zooming process incorporates displacing the camera center to a new location whilst preserving the orientation of the camera (i.e. no rotation). This displacement of the camera center in general alters three out of the five intrinsic parameters: the skew and aspect ratio remain very stable while the principal point and focal length vary. From this point of view, a stationary zooming camera can be viewed as a mechanism for obtaining images from pure translational motion. Obtaining two views from pure translation is hard in practice and requires a high degree of accuracy which might not be achievable, except with special equipment and often in a laboratory setup. However, we should note that although an affine reconstruction is possible from two views of a camera with fixed parameters undergoing a pure translation, it is impossible to obtain an affine reconstruction from two views in the case of a zooming camera, i.e. varying intrinsic parameters with pure translation motion constraints as proved in [52]. Furthermore, the modulus constraints cannot be used in the case of zooming or varying camera parameters. The only remaining existing possibility is to rely on scene constrains such as parallel lines and planes. This chapter introduces a new method which neither relies on scene constraints nor on explicit motion constraints.

The affine self-calibration method we propose, relies on less restrictive constraints on the geometry of a zooming camera than Willson's. Indeed, we rely on the fact that the optical center is mostly displaced along the $Z-$axis but, unlike Willson's model, we allow it to also shift along the $X-$ and $Y-$ axes by any amount. More importantly, we consider the image plane after zooming parallel to the one before zooming. This assumption includes the case in which the image plane undergoes a pure translation (as in Willson's model) but also allows the image plane to rotate around any axis parallel to the $Z-$axis. Note that our assumptions imply that all the intrinsic parameters are free to vary. Under these assumptions, we achieve our affine self-calibration goal by tracking the motion of the principal plane Φ (introduced in Section 4.1.2) of the camera which we denote hereafter $\Phi_{i,s}$ (Figure 4.2), i.e. the plane containing the optical center of camera i at zoom setting s and parallel to the image plane. Because $\Phi_{i,s}$ contains $C_{i,s}$, this plane is also displaced under the effect of zooming to overlap $\Phi_{i,s'}$ containing the new camera center $C_{i,s'}$. Since our assumption is that the image plane after zooming is parallel to the one before zooming, then so are the principal planes before and after zooming.

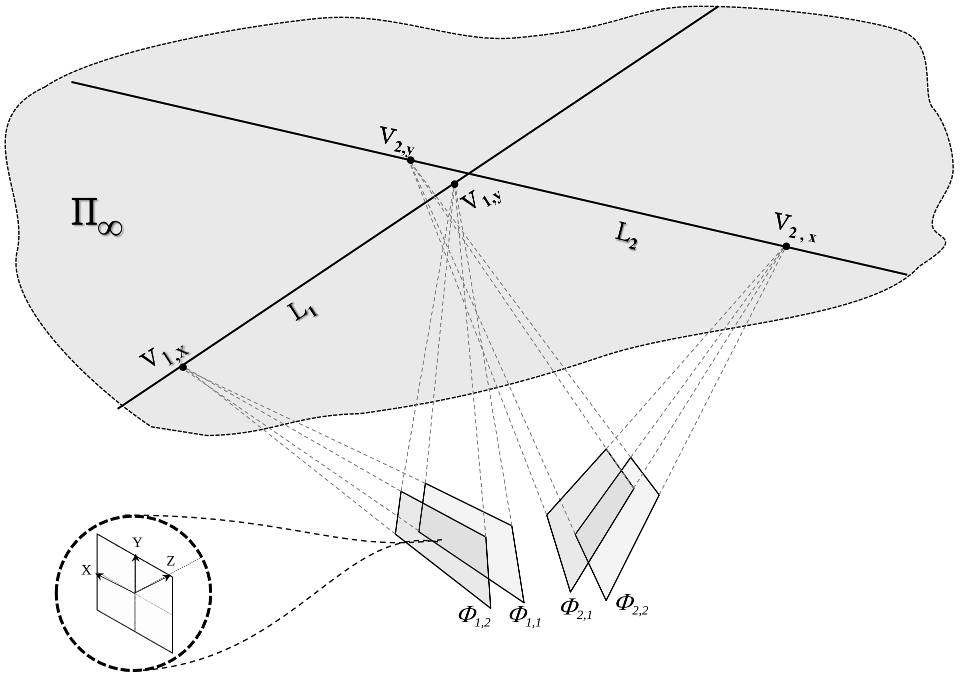


Figure 4.3: *Each pair of parallel planes intersect in a line at infinity.*

4.2.3 Locating the plane at infinity

Let $\mathsf{\Phi}_{\mathsf{i,s}}$ and $\mathsf{\Phi}_{\mathsf{i,s'}}$ be the homogeneous coordinate vectors of the principal planes at two distinct zoom settings of camera i. The two planes represented by these coordinates are parallel, if considered in any metric or affine frame, and hence intersect the plane at infinity in a line. Although parallelism is not preserved under projective transformations, the linear relationship of parallel planes (see Equation 4.4) still holds, hence we have:

$$\alpha_{\mathsf{i,s}}\mathsf{\Phi}_{\mathsf{i,s}} + \alpha_{\mathsf{i,s'}}\mathsf{\Phi}_{\mathsf{i,}} = \mathsf{\Pi}_\infty \quad \text{and} \quad i = 1,2 \tag{4.5}$$

where, $\alpha_{\mathsf{i,s}}$ and $\alpha_{\mathsf{i,s'}}$ are non-zero scalars and, the coordinate vectors $\mathsf{\Phi}_{\mathsf{i,s}}$, $\mathsf{\Phi}_{\mathsf{i,s'}}$ and $\mathsf{\Pi}_\infty$ may be expressed in any reference frame.

As a consequence, the coordinate vectors of the principal planes may be provided by the last rows of the associated projective camera matrices $\mathsf{P}_{\mathsf{i,s}}$ and $\mathsf{P}_{\mathsf{i,s'}}$ (see subsection 4.1.2) whose calculation only requires feature correspondences across images. When considering a single camera i at two distinct zoom settings, the linear relationship (4.5) provides four independent equations in six unknowns: $\alpha_{\mathsf{i,s}}$, $\alpha_{\mathsf{i,s'}}$ and the four coordinates of the plane at infinity $\mathsf{\Pi}_\infty$. These equations define a one-parameter family of points describing the line L_i on the plane at infinity at which $\Phi_{i,s}$ and $\Phi_{i,s'}$ intersect, as shown on Figure (4.3). All principal planes originating from the same camera meet in this line .

In order to retrieve the plane at infinity, at least two distinct lines on this plane are necessary. Such lines can be obtained from two or more distinct zooming cameras in general position. For computation of the plane at infinity, consider the pair of zoom images taken by each camera i, $i = 1, 2$ and by substitution in (4.5):

$$\alpha_{1,1}\mathsf{\Phi}_{1,1} + \alpha_{1,2}\mathsf{\Phi}_{1,2} = \mathsf{\Pi}_\infty \tag{4.6}$$

and

$$\alpha_{2,1}\mathsf{\Phi}_{2,1} + \alpha_{2,2}\mathsf{\Phi}_{2,2} = \mathsf{\Pi}_\infty \tag{4.7}$$

Combining equations 4.6 and 4.7 provides:

$$\alpha_{1,1}\mathsf{\Phi}_{1,1} + \alpha_{1,2}\mathsf{\Phi}_{1,2} - \alpha_{2,1}\mathsf{\Phi}_{2,1} - \alpha_{2,2}\mathsf{\Phi}_{2,2} = 0 \tag{4.8}$$

where 0 is a 4×1 null vector. Equation (4.8) is equivalent to a system of linear equations on the form $Ax = 0$, where the 4×4 matrix A constituted from the four principal planes column vectors $\Phi_{1,1}, \Phi_{1,2}, \Phi_{2,1}$ and $\Phi_{2,2}$ and the 4-vector x correspond to the four unknown scalars $\alpha_{1,1}, \alpha_{1,2}, \alpha_{2,1}$, and $\alpha_{2,2}$. Such system can be solved easily using Singular Value Decomposition (SVD). Once the scalars $\alpha_{i,j}$ are recovered, the plane at infinity can be computed by substituting the corresponding scalars $\alpha_{i,j}$ in the equations (4.6) or (4.7). In practice, we set $||\mathsf{\Phi_{i,s}}|| = 1$ to achieve numerical stability.

4.3 Experiments and results

We have carried out several experiments that have validated the proposed method. In particular, we have used off-the-shelf low-end cameras to capture our images. In all our experiments, we have used the method reported in [82] to compute a consistent set of projection matrices for all acquired images. Furthermore, only linear calculations have been employed. Theses results might likely be improved, if non-linear optimization is added at different stages.

The quality of the results is assessed through the RMS error of the 3D reconstruction in comparison to the ground truth. In addition, visual 3D reconstruction of different indoor and outdoor scenes are also presented. Also we have compared our method to two other well known methods for recovering the affine reconstruction. Note that these methods are not automatic as they rely on scene constraints, such as *scene parallel lines* and *scene parallel planes*. Our method on the other hand, does not require any scene constraint and uses only point correspondences across images. The obtained results are comparable to these two constraint-based methods, even though such comparison is unfair.

In order to evaluate the obtained affine reconstruction, the reconstructed points are first aligned with the Euclidean ground truth data via an affine transformation, then the RMS error is calculated. A linear affine transformation T_a can be calculated from 4 or more of such points (see for instance Affine Direct Linear Transformation DLT in [43]).

Beside using the RMS error as a quality measurement, we have also provided visual representation of the reconstruction, in a wireframe model representation. At the same time, to make any affine distortion more visible, the reconstructed points are translated to the coordinate origin while anisotropically scaled by making the 3 dimensions of the scene approximately round. Note that such applied transformation is affine.

4.3.1 Simulations

In each simulation, we have randomly generated a cloud of 125 points confined within the unit sphere along with a pair of cameras (see Figure (4.4)). Each camera was roughly pointing at the center of the sphere, from which it was randomly located at a mean distance of 3 meters and 25 cm standard deviation. The generated cameras were created to simulate a zooming camera with a zoom length capabilities of 12.5 - 35 mm, a CCD array of 8 × 8 mm and 64 pixels per millimeter. For each generated camera, we capture two images (before and after zooming) by projecting the scene points onto 512 × 512 pixels images. The first captured image by each camera is taken with initial camera parameters of focal length 800 pixels (12.5 mm), zero image skew, unit aspect ratio, and principal point located at the center of the image. For the second image taken by each camera, and in order to simulate zooming, the focal length increased randomly to a length within the range of 15-35 mm (960-2240 pixels). The optical center of the camera is translated by a relative amount within the range of 2.5 and 22.5 mm along the optical axis from its initial position before zooming.

Table 4.1: 3D errors: simulated scenes and cameras.

Noise(pix)	0.0	0.2	0.4	0.6	0.8	1	1.2	1.4	1.6	1.8	2.0
Mean RMS(%)	0.0 %	3 %	3.5 %	4.0 %	5.0 %	5.5 %	6.0 %	6.5 %	7.0 %	7.7 %	8.0 %

For each scene and camera, we progressively corrupt the pixel coordinates by a zero-mean Gaussian noise with standard deviation in the range 0 to 2 pixels (with a 0.2 pixel step). The plane at infinity is estimated using our method and the affine 3D structure is obtained by triangulation and aligned with the original data via the best affine transformation.

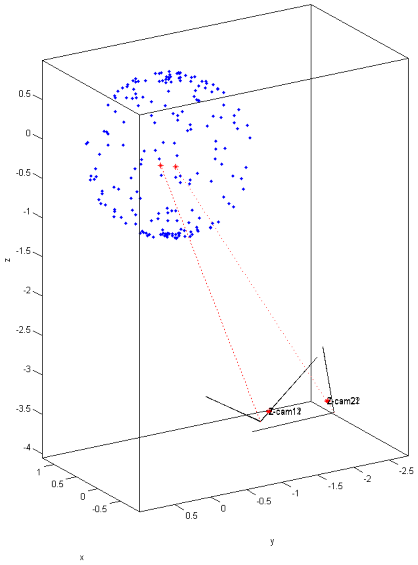


Figure 4.4: *Simulations setup*

The mean 3D relative RMS error over 1000 independent trials for each noise level, reported in Table 4.1, was used as a quality measurement. The results we have obtained show that, the quality of the affine reconstruction is perfect in the absence of image noise which confirms our method theoretical correctness. Trivially the quality progressively deteriorates with the increasing amplitude of pixel noise, but the relative error remains within an acceptable range.

4.3.2 Laboratory experiments

Here, we have used a scene with known geometry, considered to be our ground truth even though its measurements have been obtained using a low-end ruler. The scene, a 184 × 244 × 244 mm cuboid-shaped calibration pattern with 30 × 30 mm black & white squares (see Figure 4.5), was imaged by three different low-end (cost below $100 each) digital cameras with motorized zoom. The three cameras consist of a *Kodak EasyShare*, a *Sony Cyber-shot DSC-S930* and a *Sony DSC-W560*. Each camera captured two images, at two different zoom settings, while mounted on a tripod located at about 2 meters from the scene. A total of 161 feature points were extracted and matched across the six images.

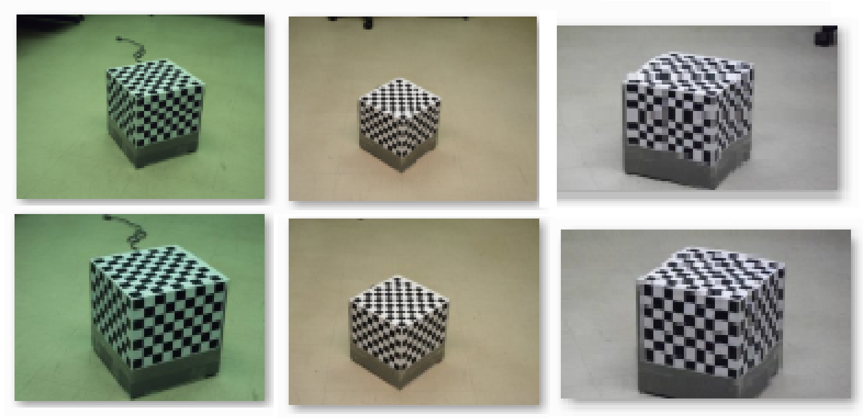


Figure 4.5: *The three pairs of images used in the minimum case*

Using each pair of cameras (total of 3 different cases), the proposed method was able to recover the affine 3D structure. A sample affine 3D reconstruction with

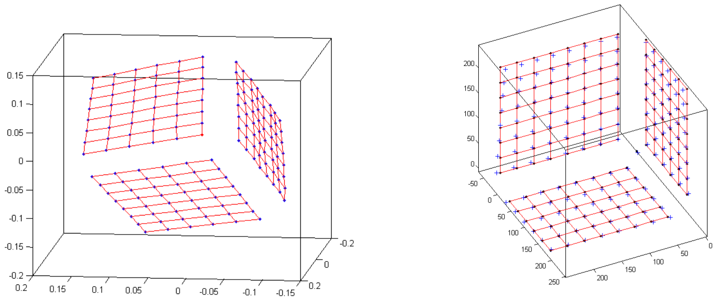


Figure 4.6: *3D Affine reconstruction (left) and Euclidean reconstruction (right). Reconstructed points are marked with '+'.*

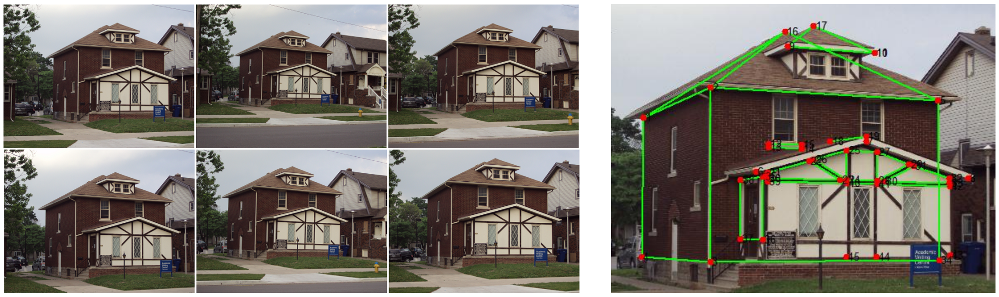
Table 4.2: 3D RMS errors (in mm) using three different pairs of zooming images.

Used Camera	EasyShare × DSC-S930	EasyShare × DSC-W560	DSC-S930 × DSC-W560
3D RMS error	4.4 mm	3.5 mm	4.8 mm

wireframe connecting the reconstructed points is shown on Figure 4.6 (left). These 3D affine reconstruction were aligned with the known geometry via an affine Direct Linear Transformation (DLT [43]). Figure. 4.6 (right) shows the reconstructed 3D model after the DLT alignment. The 3D RMS errors obtained for the 3 different pairs of cameras, given in Table 4.2, were comparable and did not exceed a mean RMS error of 5 mm. The good quality of these results obtained by different low-end cameras validates the correctness of our assumptions and efficiency of our method with real cameras. When considering the fact that the scene was at about 2000 mm from the cameras and that feature points were not extracted with a sub-pixel accuracy, these results are excellent as the mean errors ranged from 3.5 mm to 4.8 mm.

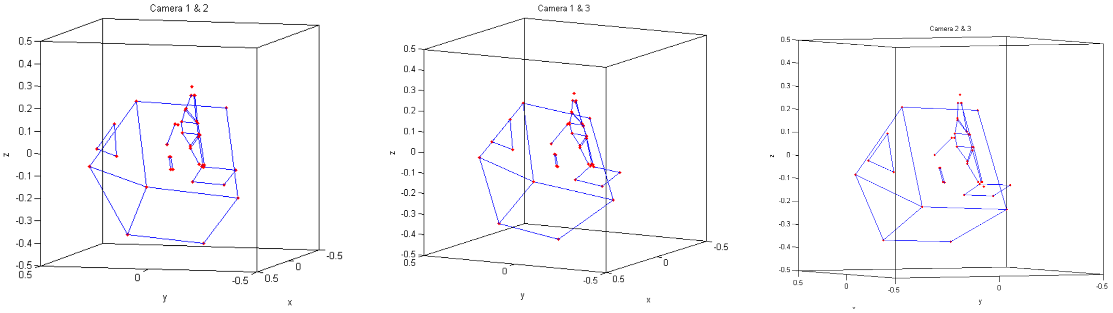
4.3.3 An outdoor scene

In a third experiment, we applied our method to an outdoor scene of a house. Our camera, a *Sony DSC-S930*, was placed at approximately 25 meters off a house then we have captured 2 images at different zoom settings. This process was repeated 3 times for 3 different positions, with a few meters between them. A total of 45 points were extracted and matched across the different images of the scene as shown in Figure 4.7.



(a) (b)

Figure 4.7: *Outdoor scene. (a) 3 pairs of images taken from 3 different positions where each pair of images is taken with different zoom settings. (b) Sample selected feature points*



(a) (b) (c)

Figure 4.8: *Affine reconstruction of the "house" scene obtained by the different pairs of cameras*

Using 2 pairs of zoom images (total of 3 different cases), we have applied our method in the minimum case and have computed the 3D affine reconstruction of the points. Figure 4.8 shows the affine reconstruction obtained from each pair of cameras from different viewpoints. To better interpret the results, the reconstructed affine 3D points were translated to the origin and anisotropicaly scaled by making the 3 dimensions of the scene approximately round. In addition, a wireframe connecting the reconstructed 3D points were drawn to better visualize it. As clearly shown on Figure 4.8, the obtained affine reconstructions are very good and are similar to each other, regardless of the pair of cameras used.

4.3.4 Comparison with other methods

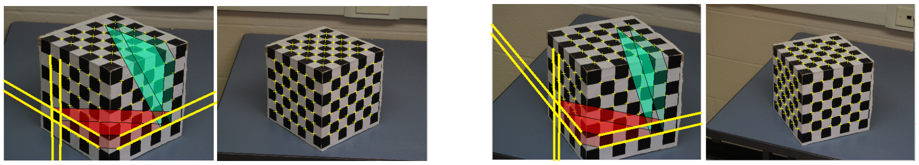
In these experiments we confront our linear affine self-calibration method with other methods employing scene constraints. Altogether, we have tested three methods using the same images obtained from the same scene and cameras. The first method relies on the use of 3 vanishing points from 3 orthogonal pairs of parallel lines that make it possible to calculate the plane at infinity and hence the affine structure. The second method employs 2 pairs of parallel planes from the scene to locate the plane at infinity and recover the affine structure. The third method is our zoom-based affine reconstruction in the minimum case of 2 cameras and 2 zoom images. In order to provide such scene constraints in a single model, i.e., parallel planes and parallel lines, along with ground truth data, we have used our calibration cube. Figure 4.9 shows the calibration cube along with the parallel lines and planes that have been used as scene constraints. Note that unlike our method, the methods employing scene constraints do not work for scenes where no parallel lines or planes exist.

The Canon PowerShot camera was used to capture images from 2 viewpoints. For each viewpoint, the camera was again mounted on a tripod and captured 2 images at different settings of its zoom. Figure 4.9 shows the images captured from each

Table 4.3: 3D RMS errors (in %) using three different methods.

Method	Parallel lines	Parallel planes	Zooming Cameras
3D RMS error(%)	0.7967 %	0.7493%	1.0277%
Scene Constraints	3 pairs of parallel lines	2 pairs of parallel planes	no scene constraints

of the viewpoints. The two viewpoints were roughly 70 cm apart. A total of 108 points, located on 3 mutually orthogonal faces of the cube shown in Figure 4.9, were extracted and matched across all 4 images.

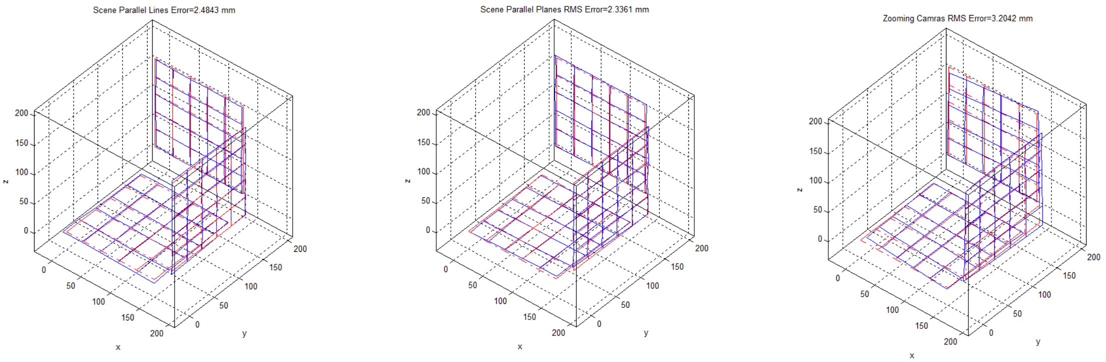


(a) Viewpoint # 1

(b) Viewpoint # 2

Figure 4.9: *Four images taken with two stationary cameras at different zoom settings showing the manually selected parallel planes and lines.*

The affine 3D scene structures were obtained using each method independently, then aligned with the ground truth data using the affine DLT method. A 3D reconstruction of the aligned data along with ground truth data is presented in Figure 4.10. Table 4.3 shows the results of the affine reconstruction where the RMS errors obtained with scene constraints were about 0.75% while the zoom-based method provided a reconstruction with 1.03% error. The latter is clearly very good considering that our method does not employ any a priori knowledge about the scene as it relies on point correspondences only. Note that, the used scene constraints in these experiments, parallel lines and planes, are perfect cases for such methods. Such perfect lines and planes constraints rarely exist in real scenarios. Moreover, and more importantly, our method is not limited to scenes exhibiting such constraints giving it an advantageous



(a) Parallel Lines Method (b) Parallel Planes Method (c) Zooming Method

Figure 4.10: *Obtained 3D reconstruction after upgrading the affine reconstruction to Euclidean*

flexibility.

4.4 Conclusion

This chapter focused on the problem of obtaining affine reconstruction and camera matrices from a pair of stationary non-rotating zooming cameras. This is a problem which has not been specifically addressed in the literature previously, where only non-linear general solution are available. A simple linear method for locating the plane at infinity is proposed, allowing to upgrade the initial projective reconstruction to affine. The method retrieves the plane at infinity - directly from the projective projection matrices of the cameras - by exploiting the displacement of their principal planes under the effect of zooming. Different than all other existing approaches, the proposed method for locating the plane at infinity does not rely on restricted intrinsic camera parameters, nor does it depend on special camera motions or scene constraints. The proposed method is based on the valid observation that the principle planes of a stationary camera at two distinct zoom settings are parallel. This observation does not impose any restriction on the intrinsic camera parameters as all parameters are allowed to vary. Obtaining the affine reconstruction simplifies the process of upgrading it to metric, the topic of the next chapter. Besides the simplicity of our method, the obtained results using low-end zooming cameras yielded good accuracy in comparison to other methods, which rely on scene's constraints.

Chapter 5

Metric Auto-Calibration and 3D-Reconstruction For Stationary Zooming Cameras

5.1 Introduction

The auto-calibration problem of a system of zooming cameras is nonlinear and challenging to solve [13, 29, 48, 72, 81, 99, 104]. In this chapter, a linear stratified auto-calibration method for stationary zooming cameras is proposed and evaluated. An affine upgrade of the scene and cameras is first calculated. Then, the intrinsic parameters and metric structure can be linearly obtained. The affine calibration is achieved by using an improved version of the method for locating the plane at infinity from a stereo pair of zooming cameras presented in the previous chapter (chapter 4). This enhancement is desirable for cases where more than two cameras are available and when each camera may capture more than two zoom images. Employing all cameras and images at hand helps to cope with image noise and critical motions.

Once the plane at infinity is retrieved from parallel principal planes, the no-skew and/or known aspect ratio constraints can be used to linearly estimate the so-called Image of the Absolute Conic (IAC) and hence all the intrinsic parameters. It is also well-known that estimating the camera's intrinsic parameters is very sensitive to the localization of the plane at infinity. Hence, we have investigated two methods for linearly calculating an estimate of the IAC (and as a consequence the camera parameters) from the linearly estimated plane at infinity: (a) the well-known linear least-squares through Singular Value Decomposition (SVD) [41], and (b) a Linear Matrix Inequality formulation which allows to enforce the requirement of a positive-definite IAC [59]. Our extensive experiments both on simulated and real images (using a variable number of cameras, zoom settings and image noise) show that the estimate of the intrinsic parameters, obtained by both methods along with the zoom-based candidate plane at infinity, allow for a simple nonlinear least-squares optimization procedure to converge towards the optimal parameters.

This chapter is organized as follows. Section 5.2 describes our self-calibrating method for zooming stationary cameras. First, the linear method for estimating the plane at infinity, presented in the previous chapter, is reformulated to robustly incorporate more zoom images and cameras. Next, under the assumption of square pixels, linear method for estimating the IAC and hence the intrinsic parameters is described. Our experiments and the results we have obtained are described and discussed in Section 5.3. Section 5.4 concludes our work.

5.2 Zoom-based camera auto-calibration

In this section, we present and describe the stratified camera auto-calibration method for a set of two or more stationary zooming cameras. Assuming a consistent set of projective camera matrices has already been recovered from image correspondences, the full camera auto-calibration is carried out using the following steps.

1. Affine upgrade: linear estimation of the plane at infinity using the zoom based method discussed in chapter 4, exploiting the assumption that the principal planes corresponding to distinct zoom levels of the same camera are parallel to one another. This method is *extended* and *reformulated* to support the general case of an arbitrary number of zoom images and cameras.

2. Metric Upgrade: under the valid assumption of zero-skew and unit aspect ratio, the intrinsic parameters of all cameras can be linearly calculated. We investigate two linear methods for the computation of the IAC including the singular Value Decomposition (SVD) and Semi-Definite Programming (SDP).

3. Refinement(optional): the initial linear estimation can be refined to obtain the optimal intrinsic parameters and coordinates of the plane at infinity through a nonlinear least-squares optimization procedure.

5.2.1 Estimating the plane at infinity: revisited

It is shown in the previous chapter that two images of a zooming camera allows to identify a line at infinity constituted from the intersection of its (parallel) principal planes using the equation:

$$\alpha_{\mathsf{i,s,s'}}\mathsf{\Phi}_{\mathsf{i,s}} + \alpha_{\mathsf{i,s',s}}\mathsf{\Phi}_{\mathsf{i,s'}} = \mathsf{\Pi}_{\infty}. \tag{5.1}$$

where $\mathsf{\Phi}_{\mathsf{i,s}}$ and $\mathsf{\Phi}_{\mathsf{i,s'}}$ are the homogeneous coordinate vectors of the principal planes at two distinct zoom settings of a camera i, and $\alpha_{\mathsf{i,s,s'}}$ and $\alpha_{\mathsf{i,s',s}}$ are non-zero scalars.

Two or more zooming cameras ($n \geqslant 2$), two of which not pointing in the same direction, allow to recover the coordinates of the plane at infinity by solving a linear system of equations (5.1), involving all cameras and zoom images.

However, when relying on (5.1), each pair of parallel principal planes introduces two new unknowns $\alpha_{\mathsf{i,s,s'}}$ and $\alpha_{\mathsf{i,s',s}}$ for every given camera. For instance, two cameras, each capturing two zoom images, yield eight linear equations in eight unknowns and suffice in theory to retrieve the plane at infinity. In practice, more than two cameras may be available and each camera may very well capture more than two images at distinct zoom settings of its lens. In such case, in order to cope with image noise, it is highly desirable to employ all cameras and images at hand. However, $n \geqslant 2$ cameras and $m_i \geqslant 2$ zoom images captured by camera i, give rise to a system of $4n$ linear equations (5.1) in $4 + \sum_{i=1}^{n} m_i(m_i - 1)$ unknowns. Although such linear system can be solved (typically using SVD), the increase in the number of unknowns, in the presence of noisy image measurements, may affect negatively the accuracy of the results.

Incorporating more cameras and images

Fortunately, (5.1) can be brought to a system of equations solely involving the coordinates of the plane at infinity. This can be achieved by considering that neither the plane at infinity nor any of the principal planes contain the origin of the reference frame. Note that this is always possible either by arbitrarily choosing the world reference frame within the scene or by discarding the principal plane containing the origin of the frame, should the latter be attached to one of the cameras. Under this assumption, the coordinate vector of the plane at infinity and that of any given principal plane are of the form $\mathsf{\Pi}_\infty^\mathsf{T} = (\pi_\infty^\mathsf{T} 1)$ and $\mathsf{\Phi}_{\mathsf{i,s}}^\mathsf{T} = (\phi_{\mathsf{i,s}}^\mathsf{T}\ 1)$ where π_∞ and $\phi_{\mathsf{i,s}}$ are

3-vectors. Equation (5.1) becomes

$$(\pi_\infty^\mathsf{T}\ 1) = \alpha_{\mathsf{i,s,s'}}(\phi_{\mathsf{i,s}}^\mathsf{T}\ 1) + \alpha_{\mathsf{i,s',s}}(\phi_{\mathsf{i,s'}}^\mathsf{T}\ 1) \tag{5.2}$$

from which one can easily deduce that $\alpha_{\mathsf{i,s,s'}} + \alpha_{\mathsf{i,s',s}} = 1$. Note that $\alpha_{\mathsf{i,s,s'}}$ and $\alpha_{\mathsf{i,s',s}}$ can neither be zero nor one, since the plane at infinity is distinct from any of the principal planes. As a consequence, one of the unknown scalars, say $\alpha_{\mathsf{i,s',s}}$, can be eliminated by substitution which simplifies the equation to

$$\pi_\infty = \alpha_{\mathsf{i,s,s'}}(\phi_{\mathsf{i,s}} - \phi_{\mathsf{i,s'}}) + \phi_{\mathsf{i,s'}}. \tag{5.3}$$

Let $[v]_\times$ denote the skew-symmetric matrix induced by the cross-product of some 3-vector v. Since $[v]_\times v = (0, 0, 0)^\mathsf{T}$, the remaining unknown scalar $\alpha_{\mathsf{i,s,s'}}$ can be eliminated by multiplying both sides of (5.3) by $[\phi_{\mathsf{i,s}} - \phi_{\mathsf{i,s'}}]_\times$ which leads to

$$[\phi_{i,s} - \phi_{i,s'}]_\times \pi_\infty = [\phi_{i,s}]_\times \phi_{i,s'},$$

or equivalently, using the notation of the 3×4 matrix $\mathsf{M}_{\mathsf{i,s,s'}}$ to describe the relation between pairs of parallel planes for camera i at two distinct zoom setting s and s', by

$$\mathsf{M}_{\mathsf{i,s,s'}}\Pi_\infty = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \text{ where } \mathsf{M}_{\mathsf{i,s,s'}} = [\ [\phi_{i,s} - \phi_{i,s'}]_\times \quad [\phi_{i,s}]_\times^\mathsf{T}\phi_{i,s'}\ ]. \tag{5.4}$$

For a given camera i and a pair of zoom settings, the rows of the 3×4 matrix $\mathsf{M}_{\mathsf{i,s,s'}}$ are the coordinate vectors of points lying on the plane at infinity. Note, however, that only two rows are linearly independent and more cameras are required to identify the plane at infinity. Let M_i be the $\frac{m_i(m_i-1)}{2} \times 4$ matrix obtained by stacking all $\mathsf{M}_{\mathsf{i,s,s'}}$

matrices obtained from all pairs of zooming images of camera i. Considering n such cameras, the plane at infinity can be recovered by solving a homogeneous linear system of equations involving all cameras and zoom images:

$$\mathsf{\Pi}_{\infty}^{\mathsf{T}}[\ \mathsf{M}_1^{\mathsf{T}}\ \mathsf{M}_2^{\mathsf{T}} ... \mathsf{M}_n^{\mathsf{T}}\ ] = (0, 0, 0, ..., 0). \tag{5.5}$$

Retrieving the plane at infinity from (5.5) may work well for low levels of image noise. However, when using (5.5), cameras with more zoom images would carry more weight than the rest of the cameras and thus have more influence on the calculation of the plane at infinity. This may be a source of failure if the images obtained from such dominant cameras turn out to be particularly affected by noise. Furthermore, solving (5.5) allows to retrieve the plane whose distance to all 3D points (given by the rows of all the M_i matrices) is minimal. However, in the presence of noise, such solution does not take into account the fact that the rows of each matrix M_i must define a line and that the sought plane ought to contain all such lines. Therefore, a more geometrically meaningful solution is to first define the line that best fits the points at infinity defined by each zooming camera (i.e. the rows of the associated M_i) before fitting a plane to those lines. Finding the line that best fits a set of 3D points is an orthogonal regression problem. The best fitting line can be perimetrically defined as a set of points $\bar{\mathsf{M}}_i + \lambda \mathsf{D}_i$ containing the centroid $\bar{\mathsf{M}}_i^{\mathsf{T}} = (\bar{\mathsf{m}}_i^{\mathsf{T}}\ 1)$ and following a direction $\mathsf{D}_i^{\mathsf{T}} = (\mathsf{d}_i^{\mathsf{T}}\ 0)$ [94]. The centroid can be obtained by re-scaling each row of M_i so its last entry is 1 and averaging the entries in each column of the resulting matrix. Denoting by $(\mathsf{m}_{i,r}^{\mathsf{T}}\ 1)$ the r^{th} row of the re-scaled matrix $\mathsf{M}_i^{\mathsf{T}}$, the d_i component of the direction of the line corresponds to the first principal component (i.e. the right singular vector associated with the largest singular value) of the matrix formed by stacking the vectors $\mathsf{m}_{\mathsf{i,r}}^{\mathsf{T}} - \bar{\mathsf{m}}^{\mathsf{T}}$ from all rows of $\mathsf{M_i}$. Since $\Pi_\infty^{\mathsf{T}}(\bar{\mathsf{M}}_{\mathsf{i}} + \lambda \mathsf{D_i}) = 0$ for all values of the parameter λ, the plane at infinity can then be obtained by solving the linear system of equations

$$\Pi_\infty^{\mathsf{T}} \mathsf{L} = (0, 0, 0, 0, ..., 0) \text{ where } \mathsf{L} = [\ \bar{\mathsf{M}}_1\ \mathsf{D}_1\ \bar{\mathsf{M}}_2\ \mathsf{D}_2 ... \bar{\mathsf{M}}_{\mathsf{n}}\ \mathsf{D}_{\mathsf{n}}\ ]. \tag{5.6}$$

The plane at infinity corresponds to the right singular vector of the $4 \times 2n$ matrix L associated with its smallest singular value. Retrieving the plane at infinity through (5.6) has proven more accurate in practice and less sensitive to noise than when using (5.5).

affine upgrade

Once the plane at infinity is located, the projective ambiguity that affects the scene structure and the cameras can be reduced to an affine one by means of an adequate transformation T represented by a 4×4 regular matrix of the form

$$\mathsf{T} \sim \begin{bmatrix} \mathsf{P} \\ \Pi_\infty^{\mathsf{T}} \end{bmatrix}. \tag{5.7}$$

The transformation matrix T is obtained by stacking a 3×4 matrix, arbitrarily chosen in the set of camera matrices $\mathsf{P_{i,s}}$, and the homogeneous coordinate vector Π_∞^{T} of the plane at infinity. While every scene point Q is mapped by T to its new location TQ, the camera matrices in the affine frame are given by $\mathsf{P_{i,s}T^{-1}}$.

5.2.2 Estimating the intrinsic parameters

Let the 3×3 matrix $\mathsf{H_{i,s}} = \mathsf{P_{i,s}T^{-1}}[\ \mathtt{I}\ |\ \mathsf{0}\ ]^{\mathsf{T}}$ represents the inter-image homography induced by the plane at infinity and relating the reference image (whose projective camera matrix is P) and the image captured by the i^{th} camera at the setting s of its zoom. Note that $\mathtt{I}$ and $\mathsf{0}$ respectively denote the 3×3 identity matrix and the null 3-vector. When known, the matrices $\mathsf{H}_{\mathsf{i,s}}$ allow to self-calibrate the imaging system and hence to upgrade the scene's structure and cameras into a metric frame. Indeed, these matrices satisfy the relationship

$$\mathsf{H}_{\mathsf{i,s}}^{-\mathsf{T}} \omega \mathsf{H}_{\mathsf{i,s}}^{-1} \sim \omega_{\mathsf{i,s}} \tag{5.8}$$

between the Image of the Absolute Conic (IAC) ω in the reference image and its corresponding IAC $\omega_{i,s}$ in the image captured by camera i under the s^{th} zoom setting. The IAC in each image, including the reference image, is solely dependent upon the intrinsic parameters of the imaging camera. It is represented by a 3×3 symmetric positive-definite matrix that can be factored into $\omega_{\mathsf{i,s}} \sim \mathsf{K}_{\mathsf{i,s}}^{-\mathsf{T}} \mathsf{K}_{\mathsf{i,s}}^{-1}$ and whose inverse allows to recover the 3×3 upper-triangular intrinsic parameters matrix $\mathsf{K}_{\mathsf{i,s}}$,

$$\mathsf{K}_{\mathsf{i,s}} = \begin{bmatrix} \tau \mathsf{f}_{\mathsf{i,s}} & \gamma & \mathsf{u}_{\mathsf{i,s}} \\ 0 & \mathsf{f}_{\mathsf{i,s}} & \mathsf{v}_{\mathsf{i,s}} \\ 0 & 0 & 1 \end{bmatrix}, \tag{5.9}$$

through Cholesky factorization. While the focal length, denoted here $\mathsf{f}_{\mathsf{i,s}}$, and the pixel coordinates $(\mathsf{u}_{\mathsf{i,s}}, \mathsf{v}_{\mathsf{i,s}})$ of the principal point may vary with every new camera or zoom change, all cameras will be assumed to have a known (unit) aspect ratio τ and zero skew γ. Note that as long as the aspect ratio is known for a camera, $\mathsf{K}_{\mathsf{i,s}}$ can always be transformed to make $\tau = 1$.

In order to upgrade the scene and cameras to a metric frame, it only suffices to recover the intrinsic parameters matrix K of the reference camera or, equivalently, ω's entries. Under the zero skew and unit aspect ratio assumptions, each image provides two linear equations,

$$\mathsf{e}_1^\mathsf{T}\mathsf{H}_{\mathsf{i,s}}^{-\mathsf{T}}\omega\mathsf{H}_{\mathsf{i,s}}^{-1}\mathsf{e}_1 - \mathsf{e}_2^\mathsf{T}\mathsf{H}_{\mathsf{i,s}}^{-\mathsf{T}}\omega\mathsf{H}_{\mathsf{i,s}}^{-1}\mathsf{e}_2 = 0 \quad \text{and} \quad \mathsf{e}_1^\mathsf{T}\mathsf{H}_{\mathsf{i,s}}^{-\mathsf{T}}\omega\mathsf{H}_{\mathsf{i,s}}^{-1}\mathsf{e}_2 = 0, \tag{5.10}$$

in the unknown entries of ω. It is assumed that the element at the last row and last column of ω is fixed and set to 1. The vectors e_1 and e_2 are the canonical basis vectors $\mathsf{e}_1 = (1,0,0)^\mathsf{T}$ and $\mathsf{e}_2 = (0,1,0)^\mathsf{T}$. At least three images (including the reference image) captured from distinct viewpoints are needed to recover all five unknown entries of ω.

Note that some classes of motion sequences between cameras are critical for camera auto-calibration and lead to its failure [56]. For instance, under the no-skew, known aspect ratio and known plane at infinity assumptions, camera sequences containing at most two viewing directions are critical and the underlying reconstruction ambiguity is affine. Cameras with parallel (or anti-parallel) optical axes share the same viewing direction. Hence, at least three distinct viewing directions throughout the sequence of cameras are necessary for the recovery of the intrinsic parameters when solving (5.10).

The intrinsic parameters of the reference camera can be calculated by solving (5.10) either using SVD [41] or through Semi-Definite Programming (SDP), employing a Linear Matrix Inequality (LMI) formulation [59]. The advantage of solving (5.10) as an SDP problem is that, unlike when using SVD, the positive-definiteness of the sought IAC matrix ω can be enforced. Indeed, when using SVD in the presence of image noise, the retrieved ω may not be positive-definite, rendering the calculation of the camera parameters impossible. In the experiments presented in this chapter, we have tested both the SVD and the LMI-based SDP approaches.

Through solving the linear system of equations (5.10), one would like to calculate all five entries of ω under the zero skew and unit aspect ratio assumptions for all cameras. Although the recovery of the plane at infinity requires two cameras, each capturing at least two images at different zoom settings, the calculation of the IAC requires at least three images captured however from distinct viewpoints. Note that the images obtained from the same camera at different zoom settings all share a unique viewing direction. Therefore, after the plane at infinity is retrieved, at least three images captured from cameras at different locations in space, and having different viewing directions, are required for the recovery of the reference IAC. In practice, because at least two of the cameras would provide two zoom images for the affine upgrade, at least five images will be available all of which will be used for calculating ω.

Because solving (5.10) using SVD is straightforward and well-known [43], we only recall here, rather briefly, the LMI-based approach. For instance, assuming $n \geqslant 3$ cameras are available, of which at least two are zooming ($m_i \geqslant 2$ for at least two instances of i), the IAC ω of the reference image can be obtained by solving the following SDP:

$$
\begin{aligned}
\min_{\omega,\lambda_{i,s}} \quad & \textstyle\sum_{i=1}^{n}\sum_{s=1}^{m_i}\lambda_{i,s} \\
s.t. \quad & \omega \succ 0, \\
& \begin{bmatrix} \lambda_{i,s} & \mathsf{e}_1^\mathsf{T}\mathsf{H}_{\mathsf{i,s}}^{-\mathsf{T}}\omega\mathsf{H}_{\mathsf{i,s}}^{-1}\mathsf{e}_2 \\ \mathsf{e}_1^\mathsf{T}\mathsf{H}_{\mathsf{i,s}}^{-\mathsf{T}}\omega\mathsf{H}_{\mathsf{i,s}}^{-1}\mathsf{e}_2 & \lambda_{i,s} \end{bmatrix} \succ 0, \\
& \begin{bmatrix} \lambda_{i,s} & \mathsf{e}_1^\mathsf{T}\mathsf{H}_{\mathsf{i,s}}^{-\mathsf{T}}\omega\mathsf{H}_{\mathsf{i,s}}^{-1}\mathsf{e}_1 - \mathsf{e}_2^\mathsf{T}\mathsf{H}_{\mathsf{i,s}}^{-\mathsf{T}}\omega\mathsf{H}_{\mathsf{i,s}}^{-1}\mathsf{e}_2 \\ \mathsf{e}_1^\mathsf{T}\mathsf{H}_{\mathsf{i,s}}^{-\mathsf{T}}\omega\mathsf{H}_{\mathsf{i,s}}^{-1}\mathsf{e}_1 - \mathsf{e}_2^\mathsf{T}\mathsf{H}_{\mathsf{i,s}}^{-\mathsf{T}}\omega\mathsf{H}_{\mathsf{i,s}}^{-1}\mathsf{e}_2 & \lambda_{i,s} \end{bmatrix} \succ 0.
\end{aligned}
\tag{5.11}
$$

The symbol $\succ 0$ means that the symmetric matrix on the left-hand side is positive definite. Problem (5.11) is a quasi-convex one that can be solved very efficiently using interior-point methods [11]. From a practical point of view, several solvers, such as SeDuMi (`http://sedumi.ie.lehigh.edu/`) and Matlab's LMI Control Toolbox, are available. The reader may refer to [59] for more details about this SDP formulation of the problem of retrieving the IAC.

5.2.3 Refinement

As in all camera auto-calibration methods, the initial estimate of the plane at infinity and that of the intrinsic parameters need to be refined through a nonlinear optimization procedure in order to retrieve the optimal parameters. While several cost functions have been proposed in the literature, the optimal results reported in the present chapter have been obtained by minimizing the following objective function

$$\mathcal{C}(\mathsf{K}, \pi_\infty) = \sum_{i=1}^{n} \sum_{s=1}^{m_i} \frac{(\mathsf{e}_1^\mathsf{T} \mathsf{H}_{\mathsf{i,s}}^{*\mathsf{T}} \omega \mathsf{H}_{\mathsf{i,s}}^{*} \mathsf{e}_2)^2 + (\mathsf{e}_1^\mathsf{T} \mathsf{H}_{\mathsf{i,s}}^{*\mathsf{T}} \omega \mathsf{H}_{\mathsf{i,s}}^{*} \mathsf{e}_1 - \mathsf{e}_2^\mathsf{T} \mathsf{H}_{\mathsf{i,s}}^{*\mathsf{T}} \omega \mathsf{H}_{\mathsf{i,s}}^{*} \mathsf{e}_2)^2}{\| \mathsf{H}_{\mathsf{i,s}}^{*\mathsf{T}} \omega \mathsf{H}_{\mathsf{i,s}}^{*} \|_F^2} \tag{5.12}$$

where $\|.\|_F$ refers to the Frobenius norm of a matrix. Again, although at least three images captured from different viewpoints are needed, all available images are to be used in this optimization procedure. Note that, in (5.12), the matrix inverse $\mathsf{H}_{\mathsf{i,s}}^{-1}$ has been replaced by its equivalent adjoint matrix $\mathsf{H}_{\mathsf{i,s}}^{*}$ as to avoid inverting matrices during optimization and to make π_∞ appear explicitly. We recall that the inverse of a matrix and its adjoint are related by

$$\mathsf{H}^{*}{}_{\mathsf{i,s}} = det(\mathsf{H}_{\mathsf{i,s}}) \mathsf{H}_{\mathsf{i,s}}^{-1}. \tag{5.13}$$

The adjoint matrix $\mathsf{H}_{\mathsf{i,s}}^{*}$ is defined as the transpose of the matrix of co-factors of $\mathsf{H}_{\mathsf{i,s}}$. It can thus be expressed numerically as well as symbolically. In particular, it has been recently demonstrated in [30] that $\mathsf{H}^{*}{}_{\mathsf{i,s}}$ entries are affine functions of π_∞ and is of the form

$$\mathsf{H}_{\mathsf{i,s}}^{*} = (\mathsf{P}_{\mathsf{i,s}} [\ \mathtt{I} \mid 0\ ]^\mathsf{T})^* + [\ \pi_\infty\ ]_\times [\ \mathtt{I} \mid 0\ ] \mathsf{P}_{\mathsf{i,s}}{}^\mathsf{T} [\ \mathsf{p}_{\mathsf{i,s}}\ ]_\times^\mathsf{T} \tag{5.14}$$

where $\mathsf{p}_{\mathsf{i,s}}$ is the last column of $\mathsf{P}_{\mathsf{i,s}}$. It is this expression of $\mathsf{H}_{\mathsf{i,s}}^{*}$ that we have employed in our cost function (5.12).

5.3 Experiments

In order to validate and assess our auto-calibration method for stationary non-rotating zooming cameras, we have conducted several experiments using both synthetic and real images. The experiments with real images have been carried out both in a laboratory setup and with a real scene.

In all our experiments, the projective camera matrices were calculated using the method described in [82]. As customary, data normalization has been used throughout. In all our experiments, a linear estimate of the plane at infinity was obtained by solving (5.6). Initial estimates of the intrinsic parameters were obtained by solving (5.10) using SVD as well as by solving the SDP problem (5.11). Matlab LMI Control Toolbox has consistently been used throughout the experiments to solve SDPs. An estimate of the intrisic parameters have been extracted from the linearly calculated IAC. Then, the optimal intrinsic parameters of the camera have been obtained by minimizing (5.12) using the Levenberg-Marquardt algorithm. Errors on the 3D reconstruction have been recorded and reported following each step of the algorithm. In the case of real images, we also provide the resulting intrinsic parameters for the sake of comparison with those obtained by the pattern-based calibration procedure. In all the results reported here:

- "LMI linear" refers to the results obtained after solving SDP problem (5.11) without any further refinement of the results;
- "SVD linear" refers to the results obtained by solving the linear system of equations (5.10) without refinement;
- "Focal only linear" are the results obtained by assuming all the parameters of the camera, but the focal length, to be known;
- "Affine DLT" are the results obtained by aligning the affine reconstruction (directly calculated from parallel principal planes by solving (5.6)) with the Euclidean ground truth via the best affine Direct-Linear-Transform (DLT) [43].

The results obtained after refinement (by minimizing (5.12) and reconstructing the scene) are referred to as ”LMI nonlinear”, ”SVD nonlinear” and ”Focal only nonlinear”, each taking as input the results returned respectively by ”LMI linear”, ”SVD linear” and ”Focal only linear”. Apart from ”Affine DLT” which uses the best transformation, all 3D reconstructions have been carried out in the same frame as the ground truth data and re-scaled accordingly.

Note that, in the estimation of the camera parameters, although our working assumption is the absence of skew and known (unit) aspect ratio, none of the 5 sought entries of ω were fixed (apart from the element at the third row and third column which was fixed to 1). This choice was made so the IAC on the reference image and the IACs on all other views are treated equally.

5.3.1 Simulations

We have conducted extensive experiments with simulated data. In each simulation, we randomly generated a 3D point cloud consisting of 200 points confined within 1m radius sphere. The experiments were conducted using a variable number of cameras each of which randomly generated at a mean distance of 2 m from the center of the sphere with a 0.4 m standard deviation. Each camera was oriented in such a way it roughly pointed towards the center of the sphere. The generated cameras were created to simulate a zooming camera with realistic zoom length capabilities of 12.5 - 35 mm, a CCD array of 8×8 mm and 64 pixels per millimeter. Using each generated camera, we captured a number of images at different zoom settings by projecting the scene points onto 512×512 pixels images. The first image captured by each camera was obtained using a 800 pixels (12.5 mm) focal length, zero-skew, unit aspect ratio, and the principal point located at the center of the image. For the subsequent images taken by each camera, and in order to simulate zooming, the focal length was randomly increased to a length within the range of 15-35 mm (960-2240 pixels). In this way, the optical center of the camera is translated by a relative amount within the range of 2.5 and 22.5 mm along the optical axis from its initial position before zooming. For every fixed number of cameras and number of zoom images, each experiment was repeated by progressively corrupting pixel coordinates by a zero-mean Gaussian noise with standard deviation in the range 0 to 2 pixels (with a 0.25 pix step). Furthermore, each experiment was repeated for 1000 independent trials for every number of cameras, number of zoom images and noise level. The experiments were conducted using 3 to 6 cameras each capturing between 2 and 7 images at different zoom settings. For each trial, we have recorded the relative 3D RMS error (in percent) of the reconstructed Euclidean structure. Both the mean (over 1000 trials) and the median 3D RMS errors are reported in our figures.

Figure 5.1 shows the results obtained with the minimum number of viewpoints ($n = 3$), here represented by 3 distinct physical cameras, each however capturing 2 zoom images. A total of 6 images have thus been employed. Note that the linear step, whether using the LMI formulation or SVD, yielded large 3D errors (about 50% on average when using the LMIs and high levels of noise). Yet, the linearly recovered plane at infinity and camera parameters allowed for the refinement step to converge to a fair result (35% using the LMIs method and 2 pixels of noise) considering only 3 viewpoints have been used. While the refined results are comparable regardless of which linear method is initially used (”Focal only linear”, ”SVD linear” or ”LMI linear”), the method relying on solving the LMIs provided the best results. The median errors reported on the right-hand side of this figure suggest that at least half of the trials have led to a very good reconstruction with a 3D RMS error of not more than 5%.

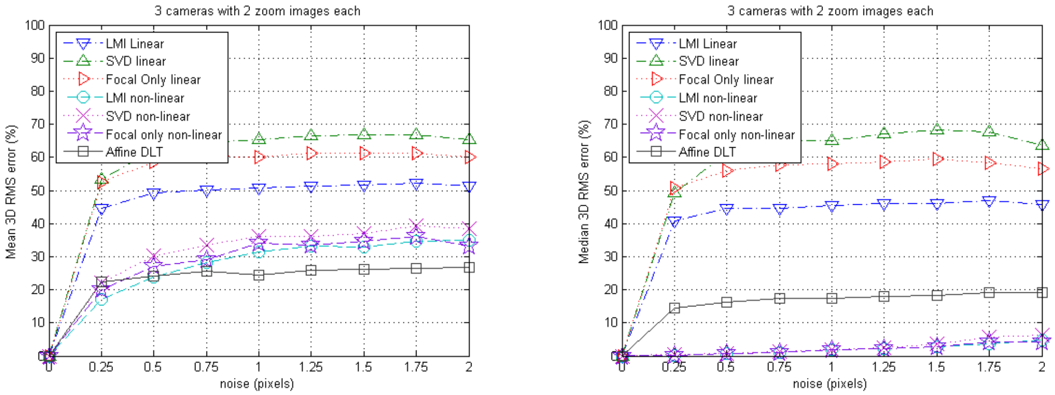


Figure 5.1: *The mean (left) and median (right) 3D RMS error (in percent) of the reconstructed scene versus different noise levels for $n = 3$ cameras each capturing 2 zoom images.*

It is clear however that more cameras and/or more zoom images are to be considered for better results. For instance, Figure 5.2 summarizes the results obtained with the same number of cameras (i.e. $n = 3$), employing, however, 7 zoom images captured by each. In the refined results, the mean 3D RMS error is half (about 18%) what it was when only 2 zoom images per camera were used (about 35%). The errors obtained via the LMI-based and SVD-based linear steps have significantly dropped to 40% (with 2 pixels of noise) from respectively 50% and 65% on average. Again, the median 3D RMS errors on the right-hand side of Figure 5.2 show that most trials have led to excellent results with at most 1% error.

As a realistic pixel localization error is generally within a single pixel, we provide in Figure 5.3 the 3D errors obtained when using 3 cameras, each capturing a variable number of zoom images (from 2 to 7) and a 1-pixel image noise.

Figure 5.3 shows that both the median and mean 3D RMS errors decrease with the increasing number of zoom images. This is particularly true when calculating

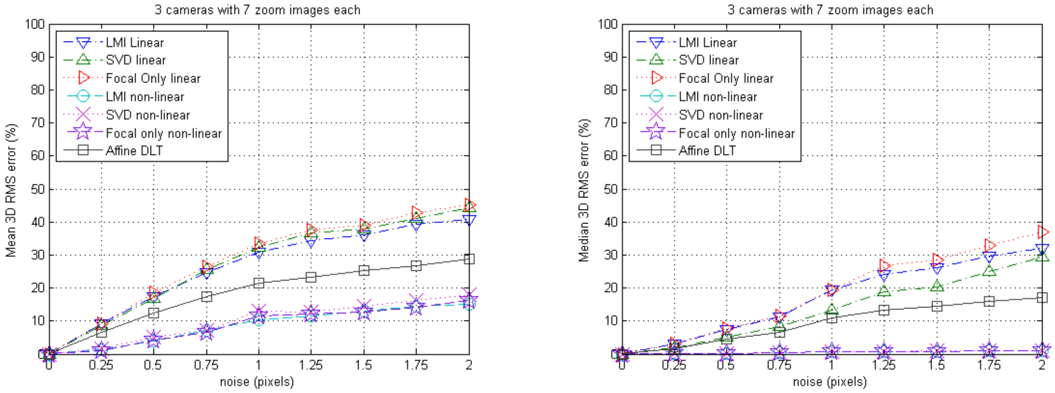


Figure 5.2: *The mean (left) and median (right) 3D RMS error (in percent) of the reconstructed scene versus different noise levels for $n = 3$ cameras each capturing 7 zoom images.*

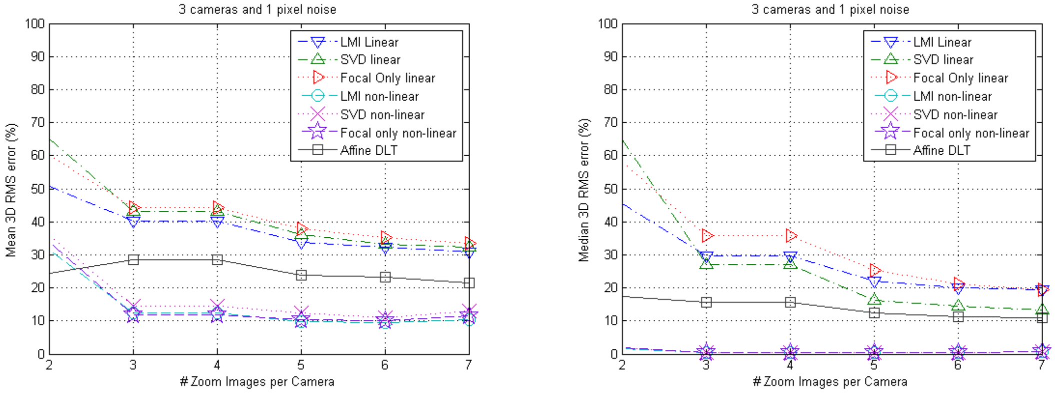


Figure 5.3: *The mean (left) and median (right) 3D RMS error (in percent) of the reconstructed scene for $n = 3$ cameras versus a variable number of zoom images per camera and 1 pixel of noise.*

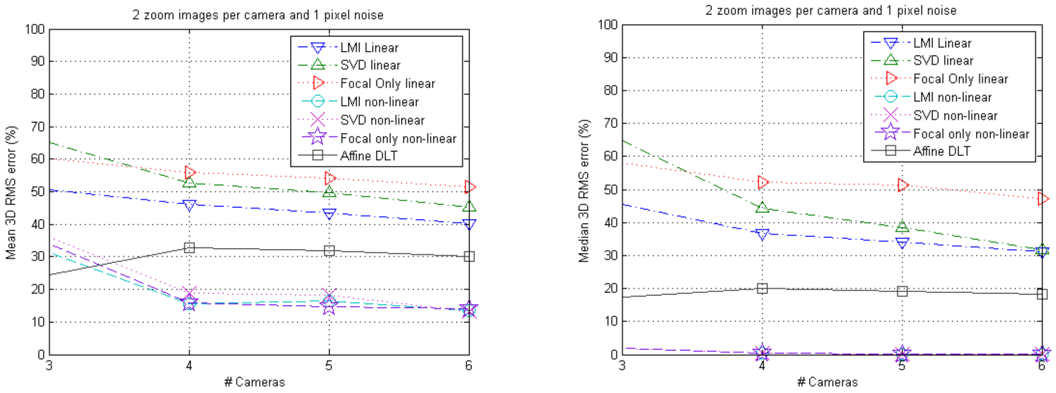


Figure 5.4: *The mean (left) and median (right) 3D RMS error (in percent) of the reconstructed scene for 1 pixel of noise, 2 zoom images per camera and a variable number of cameras.*

the camera parameters linearly (LMI linear, SVD linear and Focal only). The figure, however, shows also that the refined result remains rather stable as early as when 3 zoom images per camera are used. This suggests that, apart from noise reduction, further improvement of the quality of reconstruction cannot be only achieved by adding more zoom images but also adding more viewpoints, and hence more distinctly located stationary cameras. In this respect, we report in Figure 5.4 the reconstruction results obtained, with 1 pixel of noise, by keeping the number of zoom images per camera (only 2 zoom images) unchanged while varying the number of cameras from 3 to 6.

One can only deduce from Figure 5.4 that using more viewpoints contributes to obtaining a more accurate reconstruction. Typically, using more viewpoints and more zoom images may allow the linear step for calculating the camera parameters to provide better results than those obtained after refinement from fewer viewpoints and zooms. This is for instance the case when using 6 cameras, each capturing 7 zoom images, as depicted in Figure 5.5. The results reported therein clearly show

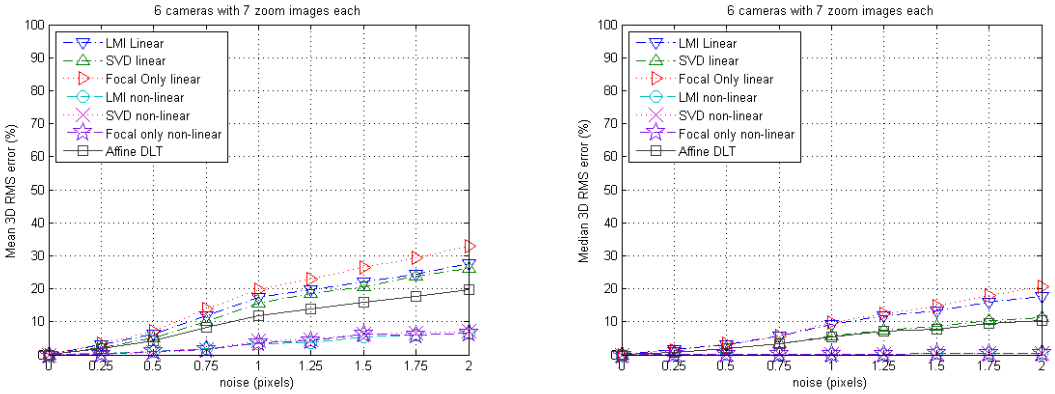


Figure 5.5: *The mean (left) and median (right) 3D RMS error (in percent) of the reconstructed scene versus different noise levels for $n = 6$ cameras each capturing 7 zoom images.*

that excellent results can be achieved in this manner. Indeed, with only 25% mean error (with 2 pixels noise), the quality of the reconstruction obtained with any of the linear methods (LMI, SVD or Focal only) exceeds that of the reconstruction obtained after refinement when using only 3 camera with 2 zoom images each (between 35% and 40% depending on the method).

We conclude this section on simulations with some remarks and comments. In all our experiments, the results obtained linearly, when considering only the focal length to be unknown, were generally worse (except when using SVD with 3 cameras and 2 zoom images each) than those obtained without such knowledge. This is because the ground truth camera parameters do not correspond to the best parameters that can be obtained in the presence of noise. It is thus recommended to leave all the parameters free and to use the LMI-based method for the linear estimate of these parameters as this method has consistently provided the best results. Furthermore, note that the errors calculated after applying the affine DLT always fall between those obtained linearly and those refined. In fact, the errors obtained via the DLT provide the best indication with regard to the quality of the linearly calculated affine reconstruction from parallel principal planes. The errors obtained following the recovery of the intrinsic parameters, whether from the LMIs or from SVD, are likely to be undermined by any proximity to a critical viewing configuration (critical motion between cameras). It is worth mentioning that the nonlinear least-squares optimization step shows quick time convergence. In our experiments, using Matlab optimization toolbox, the average time for the nonlinear optimization step convergence is less than 50 milliseconds.

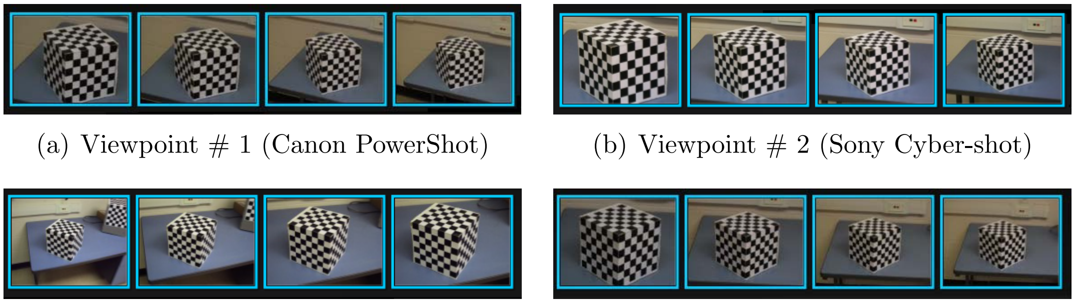
5.3.2 Laboratory experiments

In order to validate the proposed auto-calibration method, we carried out various experiments using real images in a laboratory setup. Three low-end consumer digital cameras with motorized zoom lenses have been used: a *Kodak EasyShare*, a *Sony Cyber-shot DSC-S930* and a *Canon PowerShot SX150 IS*. All three cameras were assumed to have zero skew and unit aspect ratio. Because the same physical camera has sometimes been used to capture a scene from more than a single location, in the present section and the next one (Section 5.3.3 dealing with real scenes), we often use the word ”viewpoint” instead of ”camera” to refer to a stationary camera placed at some location in space and possibly capturing several zoom images from that same location.

In these experiments, instead of using synthetic data, we presented our EasyShare, Cyber-shot and PowerShot cameras with a scene consisting of a $21 \times 21 \times 21$ cm cube-shaped calibration object exhibiting 30×30 mm black and white squares on each face. We conducted a number of experiments as to assess the quality of the reconstruction and the effect of the relative orientation between pairs of viewpoints (turntable experiments). We also compare the results of our method against those obtained by relying on scene constraints (namely parallel lines and planes). Note that, although the Euclidean structure of the calibration cube was known to us, the cube was treated as an unknown scene when applying our auto-calibration method. Knowledge about the cube has only been used to measure the resulting 3D reconstruction errors and in the method employing scene constraints.

3D metric reconstruction

The experiments we have conducted here are similar to those we carried out with simulated data. The calibration cube has been imaged from 4 distinct viewpoints by placing each camera on a tripod, roughly 1.25 to 1.5 meters from the scene. Each of the EasyShare and Cyber-shot cameras captured images from one viewpoint while the CyberShot camera imaged the scene from two viewpoints. From every viewpoint, each camera captured 4 images at different settings of its zoom lens. This has resulted in the 4 sequences of 4 zoom images given in Figure 5.6, each sequence being captured from a different viewpoint by one of the cameras.



(a) Viewpoint # 1 (Canon PowerShot)

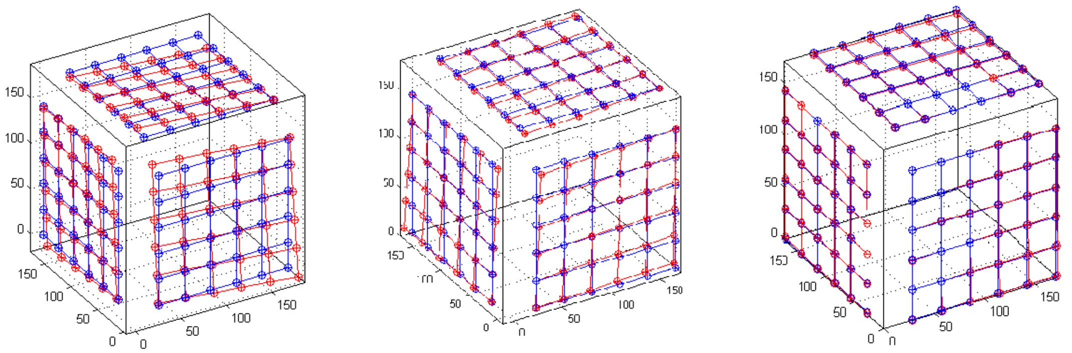
(b) Viewpoint # 2 (Sony Cyber-shot)

(c) Viewpoint # 3 (Kodak EasyShare)

(d) Viewpoint # 4 (Canon PowerShot)

Figure 5.6: *Four sequences of zoom images: each sequence was captured from a different viewpoint by a stationary camera mounted on a tripod.*

A total of 108 points, located on 3 mutually orthogonal faces of the cube, were matched across the images obtained by all cameras at all considered zoom settings.



(a) Linear LMI: 3 viewpoints (b) Linear SVD: 3 viewpoints (c) Linear SVD: 4 viewpoints

Figure 5.7: *3D Euclidean reconstruction obtained from linearly estimated parameters: the plane at infinity was calculated using only 2 zoom images per viewpoint.*

However, we have first conducted our experiments by considering only two viewpoints (Viewpoints # 1 and # 2 in Figure 5.6) starting with two zoom images from each of these two cameras and repeating the experiment with 3 and 4 zoom images. Then, the same experiment was conducted by considering three viewpoints (Viewpoints # 1, # 2 and # 3) followed by using all four viewpoints and by varying the number of zoom images each time. Table 5.1 provides the 3D RMS error (in %) - relative to the cube's diagonal - of the 3D reconstruction calculated from the LMI-based (5.11) and SVD-based (5.10) linearly estimated parameters along with the errors obtained after the nonlinear refinement (5.12) of the intrinsic parameters and plane at infinity. We have also reported in Table 5.1 the 3D errors obtained after aligning the affine structure (calculated via parallel principal planes) and the ground truth data. The ground truth data are the measurements we obtained from the cube using an office ruler. Note that, for the experiments involving only two viewpoints, only the errors obtained using the affine DLT are reported since the method requires at least three distinct viewpoints for upgrading the scene to metric. For the reader's convenience, we provide in Figure 5.7 the 3D structures obtained (red) after linearly estimating the parameters either by solving (5.10) through LMIs or SVD. Note that no nonlinear refinement was used. Each estimated structure is superimposed on the ground truth data (blue). All the estimated 3D structures in this figure have been obtained using only two zoom images for the 3-viewpoint and 4-viewpoint cases.

Table 5.1: 3D RMS errors % .

# of	# of	**Linear**			**Non-linear**	
viewpoints	zooms images	Affine DLT	SVD	LMI	SVD	LMI
2	2	10.57 %	N/A	N/A	N/A	N/A
3	2	3.22 %	100.98 %	25.33 %	9.16 %	9.16 %
4	2	1.17 %	12.12 %	8.14 %	17.57 %	17.48 %
2	3	2.22 %	N/A	N/A	N/A	N/A
3	3	0.31 %	1.63 %	2.9 %	0.48 %	0.488 %
4	3	0.46 %	1.49 %	5.74 %	0.81 %	0.81 %
2	4	7.16 %	N/A	N/A	N/A	N/A
3	4	0.54 %	6.88 %	7.24 %	0.56 %	0.56 %
4	4	0.26 %	0.76 %	2.84 %	0.86 %	0.86 %

These results show that adding more zoom images and more cameras often allows to obtain a better quality reconstruction. When using 3 viewpoints, the reconstruction error obtained through the affine DLT does not exceed 3.22% when only 2 zoom images (per viewpoint) are used. This error drops, with every additional zoom image and viewpoint, down to 0.26% with 4 viewpoints and 4 zoom images each. One can only notice that in the 3-viewpoint case with 2 zoom images, the linear step failed when SVD was used to recover the intrinsic parameters while the LMI method has allowed to recover the 3D scene with 25% error. We would like to stress the fact here that the non-linear refinement reduced the RMS % error close to an acceptable 9% (given the few viewpoints and zoom images used) for both the LMI and SVD methods. This is something we have often observed in all our experiments, including those with simulated data. This suggests that the coordinates of the plane at infinity obtained from the parallel principal planes assumption, along with the linearly estimated intrinsic parameters, have always provided initial estimates that fall within the basin of convergence of our nonlinear refinement cost function. Furthermore, although the LMI method has provided slightly less accurate results than the SVD method for the intrinsic parameters calculation, this method has consistently led to a 3D reconstruction that is within acceptable bounds from the true scene. More importantly, the nonlinear refinement step has always yielded excellent results (less than 1%) with all experiments conducted with 3 and more zoom images per viewpoint.

The intrinsic parameters of the cameras, corresponding to the first image (leftmost image in each sub-figure of Figure 5.6) captured at each viewpoint, are reported in Table 5.2. The two last rows of each table provide the parameters obtained linearly using SVD (by solving (5.10)) preceded by our linear affine auto-calibration method (”SVD”) as well as the refined parameters after nonlinear optimization (”Refined”). Note that the parameters reported here have been obtained using 4 viewpoints with 3 zoom images each. We also report in this table the intrinsic parameters obtained when calibrating the camera using the cube object as a calibration pattern with 108 known 3D points to estimate the Euclidean camera matrix from which the parameters are extracted (”Pattern”). It can be seen that the intrinsic parameters obtained after refinement are always close to those obtained though the pattern-based calibration. The focal length obtained linearly via ”SVD” is also closed to both the refined value and the one estimated through calibration. The errors on the ”SVD” parameters are mostly concentrated in one of the image coordinates of the principal point. However, these errors remain acceptable in the sense that the parameters linearly retrieved still allow the refinement step to converge as desired. Although the true principal point does not necessarily coincide with the image center, the results reported here show that, for all the cameras we have used, this point consistently falls within a reasonable distance from it.

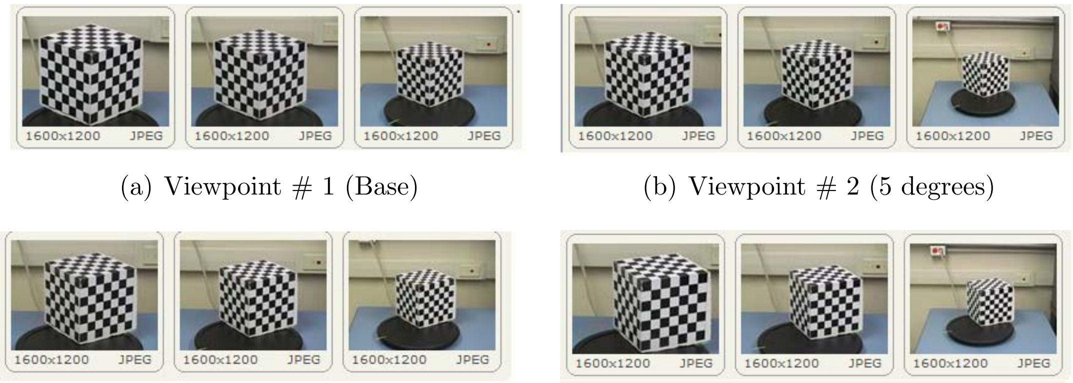
Table 5.2: Estimated intrinsic parameters corresponding to the first image of each viewpoint.

	Viewpoint #1 PowerShot 1600 × 1200 pixel					**Camera #2** Cyber-shot 3648 × 2736 pixel				
	τf	f	γ	u	v	τf	f	γ	u	v
Pattern	3436	3419	3	897	553	6537	6514	-18	1614	1298
SVD	3828	3836	3	749	1118	6872	6872	3	1881	1897
Refined	3461	3464	14	818	611	6643	6655	-61	1818	1402

	Viewpoint #3 EasyShare 2592 × 1944 pixel					**Camera #4** PowerShot 1600 × 1200 pixel				
	τf	f	γ	u	v	τf	f	γ	u	v
Pattern	8688	8656	-2	1381	872	5357	5334	-8	787	521
SVD	10580	10500	-2	1256	3014	6065	6077	-18	901	1364
Refined	8761	8765	13	1275	988	5456	5471	-6	816	574

Turntable experiments

The experiments described here have been conducted to test the relative orientation requirement between cameras (or viewpoints) for obtaining a satisfactory 3D affine reconstruction. Indeed, our method for linearly estimating the plane at infinity using the parallel principal planes assumption fails when the image planes of the physical cameras are parallel to one another. This includes the case of a pure translational motion between viewpoints. It is hence necessary to conduct experiments in a setup



(a) Viewpoint # 1 (Base)

(b) Viewpoint # 2 (5 degrees)

(c) Viewpoint # 3 (10 degrees)

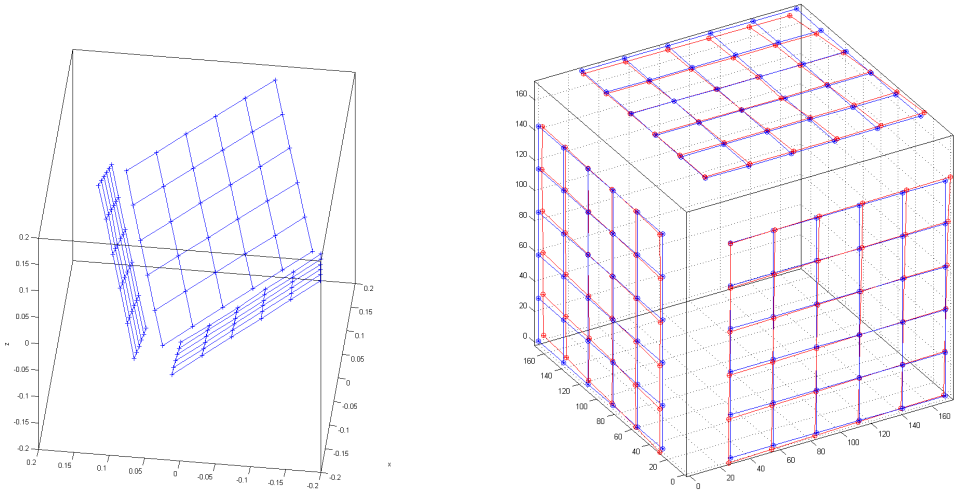
(d) Viewpoint # 4 (20 degrees)

Figure 5.8: *Each triplet of images represents 3 (out of the 6) zoom images captured by the PowerShot camera from different viewpoints.*

that allows testing the impact of an increasingly changing orientation between the zooming cameras. In these experiments, we have used the *Canon PowerShot SX150 IS* camera only.

The camera was mounted on a tripod about 2 meters from the calibration cube which, in turn, was placed on a rotating tray as shown in Figure 5.8. Six 1600×1200 pixels images have been captured at different zoom settings of the camera while the latter was kept stationary. The rotating tray was then rotated by 5, 10, and 20 degrees from its initial position. For every rotation of the tray, again six images were captured at different settings of our PowerShot's zoom lens. Figure 5.8 shows 3 (out of the 6) images captured from each of the 4 viewpoints. As in the previous experiments involving the calibration cube, a total of 108 points were extracted and matched across the images.

We have repeatedly used our linear method for retrieving the plane at infinity, reconstructed the scene in an affine frame and aligned the latter with the ground truth structure via the best affine DLT. Only two viewpoints were considered each time:



(a)

(b)

Figure 5.9: *Affine reconstruction from parallel principal planes (left) and its affine DLT alignment with the true Euclidean structure (right): Viewpoints 1 & 3 (10 degrees rotation) with 3 zoom images per viewpoint.*

viewpoint # 1 (as per Figure 5.8) and each of the other viewpoints. Furthermore, we have carried out these experiments using from 2 to 6 zoom images per viewpoint. Table 5.3 summarizes the relative 3D RMS errors (in percent) obtained in each case. A view of the affine reconstruction and another one of its alignment (through the best affine DLT) with the true structure of the cube are given in Figure 5.9 for visual assessment.

Table 5.3: 3D RMS errors (%) .

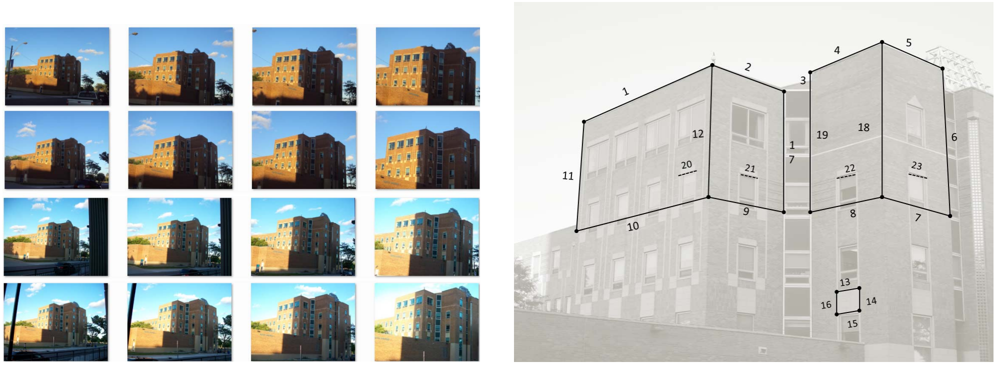
Viewpoints	Rotation	**Zoom sequence length**				
(pair #)	(degrees)	2 images	3 images	4 images	5 images	6 images
1 & 2	5	12.397 %	6.298 %	5.12 %	4.348 %	2.168 %
1 & 3	10	3.136 %	0.782 %	0.810 %	0.631 %	0.836 %
1 & 4	20	3.922 %	0.985 %	1.147 %	0.593 %	0.705 %

In particular, the results reported in Table 5.3 show that the error on the affine structure obtained with our method is rather significant (12.4%) in the minimal case of 2 viewpoints with 2 zoom images each for a small rotation of 5 degrees between the viewpoints. However, increasing the number of zoom images, even for such a small rotation, allows to improve the quality of the reconstruction (typically to 2.16% with 6 zoom images). The reconstruction quality also improves when considering wider viewpoints. These experiments show that with only 10 degrees rotation between viewpoints, a reconstruction with less than 1% error has been obtained when using 3 and more zoom images. Using wider rotation angles (as in the 20 degrees cases reported) does not necessarily improve the quality of the reconstruction. Hence, the proposed method may provide very satisfactory results even with small rotations thus allowing for the point correspondence across images to be carried out in the usual image-proximity conditions.

5.3.3 Real scene experiments

We have conducted additional experiments on a real scene consisting of a large building. Two different cameras, the Sony Cyber-shot and the Kodak EasyShare, were used to capture the images for these experiments. Images have been captured from 4 distinct viewpoints by placing each camera on a tripod at 2 different locations at about 30 meters from the scene. The viewpoints were a few meters apart from each other. At each viewpoint, the considered camera captured 4 images at different settings of its zoom lens. Each row in Figure 5.10(a) shows the 4 images captured from every one given viewpoint.

A total of 71 feature points were extracted and manually matched across all 20 images using mouse clicks. Certain line segments have been chosen to provide a wireframe model and simplify the 3D visualization of the reconstructed structures.



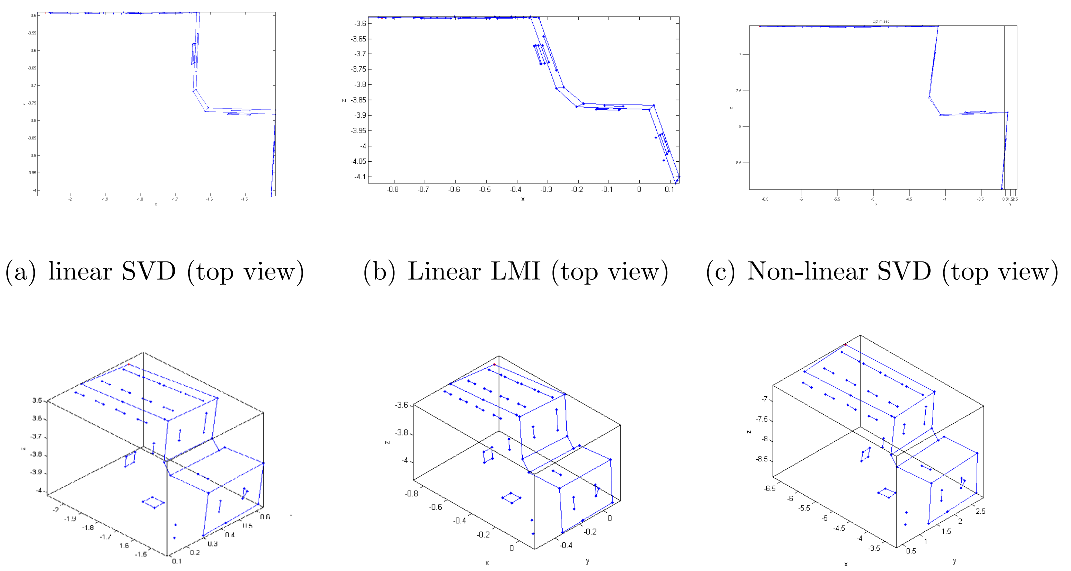
(a) (b)

Figure 5.10: *The building scene: (a) the first 4 zoom images captured from 4 different viewpoints, (b) line segments used to compute angles and distance ratios.*

A total of 23 line segments, of approximately known Euclidean geometry, have been selected as shown in Figure 5.10(b). The end-points of these segments are feature points that have been extracted and matched across the images. Selected pairs of these segments have been used to estimate distance ratios and angles and compare them against the approximately known true values. In particular, known segments of equal length, orthogonal, and parallel segments are estimated and the error from ground truth is then calculated.

Our camera auto-calibration method was applied in different scenarios each using different numbers of viewpoints and zoom images. The affine calibration computed using our method was upgraded to metric using the SVD and LMI methods for calculating the intrinsic parameters as well as the results refined through nonlinear optimization.

For this particular experiment, we report the following observations. In the case in which only 3 viewpoints were considered, regardless of the number of zoom images employed, the linear metric reconstruction was of bad quality for both the SVD and LMI methods. The recovered structure had acceptable depth in the different direction



(a) linear SVD (top view) (b) Linear LMI (top view) (c) Non-linear SVD (top view)

(d) linear SVD (side view) (e) Linear LMI (side view) (f) Non-linear SVD (side view)

Figure 5.11: *Sample wireframe models of metric 3D reconstructed scene model obtained linearly and after the refinement of the camera parameters.*

but the orthogonal angles were quite skewed. The resulting 3D structures were closer to an affine structure than to a metric one. However, the nonlinear optimization was successful for all cases and provided good metric 3D reconstructions. For the experiments where all 4 viewpoints were employed, the metric 3D reconstruction obtained linearly was good with all numbers of zoom images. The reconstruction obtained from linearly estimating the camera parameters using SVD was better than the one obtained by the LMI method. The quality of the 3D was best when all zoom images from all 4 viewpoints were used. Figure 5.11 shows top and side views of the wireframe models of the scene obtained from 4 viewpoints with 4 zoom images per viewpoint. These wireframe models were obtained using the parameters calculated linearly using SVD and those obtained after refinement. The structure obtained from the LMI-based linear estimation of the parameters is also provided in this figure.

In the case in which 4 viewpoints and 4 zoom images are considered, Table 5.4 lists the angle and ratio errors on the computed metric properties of all pairs of line segments spanning the full 3D metric structure of the recovered scene model. In this table, the average errors of all estimated values along with the standard deviations are given for the results obtained with the linearly calculated camera parameters (using both SVD and LMI methods) and for those obtained using the refined parameters. As the refinement step for both SVD and LMI produced essentially the same results, only the refined result of SVD is presented in each table.

Table 5.4: Average angle errors ± Standard deviation (degree) over all pairwise orthogonal lines and pairwise parallel lines. Average relative errors ± Standard deviation (%) over all pairs of line segments.

	Linear SVD	Linear LMI	Optimized SVD
Orthogonal lines	$8.94^\circ \pm 3.31^\circ$	$14.57^\circ \pm 7.88^\circ$	$5.28^\circ \pm 3.94^\circ$
Parallel lines	$3.53^\circ \pm 1.25^\circ$	$3.55^\circ \pm 1.37^\circ$	$2.62^\circ \pm 1.46^\circ$
Line segment ratios	2.94 ± 2.17%	3.60 ± 2.26%	8.57 ± 5.37%

After refinement, the parallel lines were found within 2.62 ± 1.46 degrees error while the orthogonal lines exhibited 5.28 ± 3.94 degrees error. The average relative error on line segments ratios is roughly about 9%. However, we recall that the exact ground truth data are not available and we have used as a ground truth only an estimate based on visual assessment of all angles and distances. The results we have obtained can be considered as fair, in particular because only 4 viewpoints have been used to image this large building that would normally require to be captured from more viewpoints.

5.4 Conclusion

In this chapter, a stratified linear method for self-calibrating a set of stationary non-rotating zooming cameras is described. The linear zoom-based method for locating the plane at infinity is extended to incorporate more zoom cameras and images. This helps to cope with higher image noise and provide more robust estimation of the affine camera matrices. The affine cameras and structure can be upgraded to metric ones by imposing some restriction on the camera intrinsic. In this work, the intrinsic parameters were estimated using the widely accepted valid assumption of square pixels (i.e. zero-skew and known aspect ratio). The results of our experiments whether using simulations, laboratory setups or an outdoor scene generally show satisfactory results using our method in the minimum case of two cameras each acquiring a pair of zooming images. Our experiments have also shown that these results are further enhanced in situations where more cameras and/or zoom images are incorporated in the process. The full auto-calibration method includes the linear recovery of the intrinsic parameters and their refinement along with the coordinates of the plane at infinity. We have tested two linear methods for calculating the intrinsic parameters: using SVD and another method employing a LMI formulation of the problem allowing to enforce the positive-definiteness of the IAC. Note that both methods have provided camera parameters that allowed the nonlinear refinement step to readily converge to the desired optimal values. Although the SVD method does not enforce the positive-definiteness of the IAC, it has consistently provided a positive definite solution from which we have always been able to extract an initial estimate of the camera parameters.

Chapter 6

Automatic Identification of Scene Parallel Planes

6.1 Introduction

Images of man-made structures are rich sources of geometrical constraints that can be exploited to aid accomplishing various computer vision tasks. Geometric properties such as parallelism, orthogonality, equality of line segments and angles, flat surfaces, squares, and cuboid are only few examples of such constraints. The latter have been successfully employed in solving several central problems in computer vision, such as camera self-calibration, recognition, 3D reconstruction, augmented reality, etc. Although these constraints are abundant in man-made environments, their usage remains limited in practice. This is mainly due to the need of certain amount of user intervention before employing such scene knowledge. Automating the identification of such geometrical scene constraints will eventually leverage the field of their usage for many other unattended computer vision applications.

This chapter investigates the problem of automatic identification of parallel planes, e.g. walls of hallways, buildings on both sides of streets, etc, from uncalibrated views.

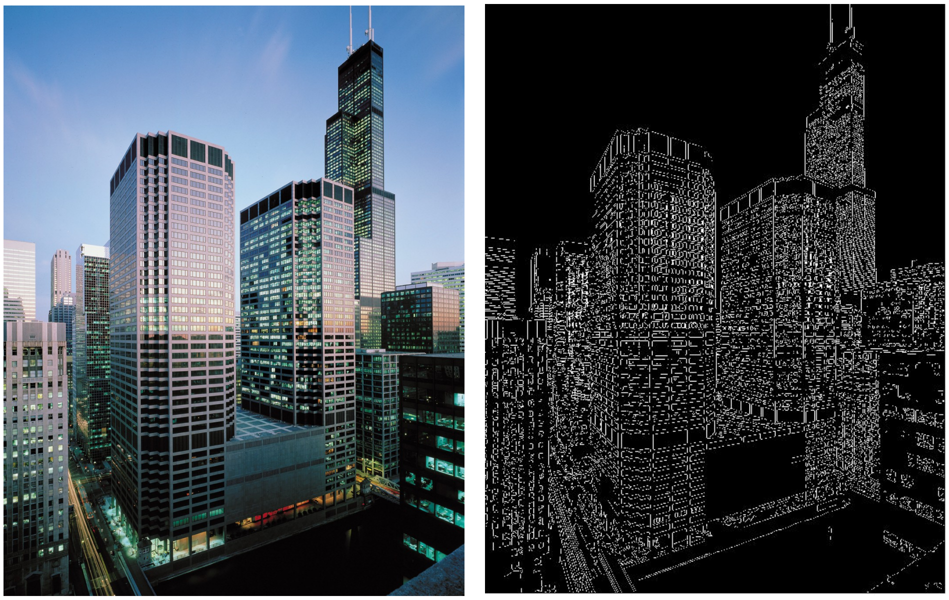


Figure 6.1: *An arial view of city buildings (left), Extracted line edges (right)*

To motivate the approach, some advantages of utilizing parallel planes in comparison to parallel lines are discussed. Consider the group of *parallel lines* with vertical direction to the ground, appearing on the different buildings' walls in Figure 6.1. All these parallel lines meet at a single point, *the point at infinity*, representing the orthogonal direction to the ground plane. Recall that any two parallel lines define a plane. Now, consider the set of different planes which may be formed by the same previous set of vertical lines. Several subgroups of these planes are parallel. Each pair or more of those parallel planes, with different orientation, uniquely identify a *line at infinity*. In this particular example, at least two lines at infinity can be identified which is enough to uniquely identify the infinity plane and, hence, the affine structure of the scene. This is a clear example that shows the superiority of parallel planes to parallel lines, in terms of number of scene constraints they provide and geometrical invariants they preserve.

A 3D reconstruction of the scene from calibrated images makes it possible to identify parallel geometrical primitives. However, in the absence of calibration parameters, the scene can be reconstructed only up to a projective ambiguity where parallelism is no longer preserved (e.g. parallel lines may appear intersecting). Automatic detection of line parallelism from uncalibrated images has been thoroughly studied in the literature through the detection of vanishing points (see for example [87,103]). In general, automatic detection of vanishing points relies on identifying and clustering dominant directions of line segments present in the scene. Despite that a vanishing point can be identified from images of two parallel lines, reliable detection of vanishing points mandates the presence of many parallel lines sharing the same direction.

Conversely, the identification of the scene's parallel planes from uncalibrated images has not gained adequate attention from researchers in the past. Perhaps the main reason is the nature of these geometrical primitives where, and in contrast to points and line segments which can be identified from a single image, the identification of a scene plane requires in general a set of matched points, or lines, between two or more images. Nevertheless, identifying planes from images of a scene is a very feasible task as several automatic and robust plane extraction methods has been proposed in the past. For example, a fully automatic method for detecting the planes present in a scene using a set of matched point and line features across a pair of images has been proposed in [62]. In this method, an iterative voting scheme, based on pairs of point and line features, searches for planar homographies. In another work, Vincent and Laganieŕe developed in [100] a robust homography-based method which can be used as a first step in an image analysis process (e.g. aid in weak calibration). More recently, Amintabar and Boufama [2] proposed a method not only for detecting planes from uncalibrated images but also distinguishing physical and virtual ones with certain level of confidence. Note that regardless of the robustness or the accuracy of the above mentioned algorithms, any three 3D points in projective space can be used to define a plane. The 3D projective reconstruction of matched image points is always possible from 8 or more point matches across two or more uncalibrated images.

The importance of parallel planes usage has been justified by their aid to solving a wide number of Computer Vision and scene analysis problems. Tebaldini et al. in [95] proposed a method for generating synthesised views under the knowledge of at least one pair of parallel planes. In the people tracking problem, Khan and Shah [57] showed that the usage of parallel planes can provide great aid in situation of occlusion. Another important application for parallel planes is their aid in camera calibration. For instance, in [31], the authors utilized planes parallelism to locate the plane at infinity and hence affinely calibrate a camera. Furthermore, a calibration method for optical triangular profilometry, proposed in [15], requires parallel planes for testing.

It seems that the only automatic parallel plane identification method has been proposed by [32]. The authors show that there exists a linear relationship between the coordinates of parallel planes and those belonging to the plane at infinity. Consequently, an image-based parallel planes identification method is proposed that combines the latter relationship and the so-called modulus constraint [74] on the homography matrix of the infinity plane. It is worth mentioning that this method is limited to images taken by a camera with fixed parameters and requires at least 3 images taken at different orientations.

In this chapter, an automatic method for identifying scene parallel planes from uncalibrated images of a scene is presented. The images need not to be taken with constant camera parameters. The identification method is based on a sufficient constraint relating the coordinates of two pairs of parallel planes. If the coordinate of a pair of parallel planes are known ***a priori***, it is possible to identify other parallel planes in the scene. In our zooming camera case, such a priori can be provided automatically from a pair of zoom images and thus the whole detection process can be performed without user intervention.

This chapter is organized as follows. Section 6.2 describes the method for identifying parallel planes from uncalibrated images with varying camera parameters. Our experiments and the results we have obtained on both synthesized and real images are presented and discussed in Section6.3 and 6.4. Section 6.5 concludes this chapter.

6.2 Parallel planes identification

In this section, a method for identifying parallel planes from two or more images taken by different and unknown camera's interior is described. We assume a consistent set of projection matrices has been calculated and a set of scene planes has been identified from these images. This can be achieved using any of the different existing methods such as [2, 62, 100]. The proposed parallel planes detection method requires the knowledge of the coordinates of two scene parallel planes as a priori. This assumption is always valid if we consider the intrinsic planes of a stationary zooming camera. We recall that the third rows of the projection matrices of a zooming non-rotating camera are the coordinates of planes that are parallel to each other as discussed in the previous chapters. Note that this assumption is only required to provide valid parallel planes for any scene. The presented detection method in this paper works with any other camera configuration, i.e. without the use of zooming camera, if two parallel planes, or equivalently a line at infinity or two vanishing points, where provided as a priori. First, we describe a necessary condition constraining two pairs of parallel planes. Then, we discuss degenerate configuration cases and how they can be avoided. Finally, we present a practical use of the constraint for parallel planes detection from noisy images.

6.2.1 The parallel planes identification constraints

Let Φ_1 and Φ_2 be two 4-vectors representing the projective coordinates of two parallel scene planes in $\mathbb{P}^3$. Consider also Π_1 and Π_2 as the projective coordinates of two other distinct scene planes, that are possibly parallel.

When the projective coordinates of those four planes' vectors are stacked as rows in a 4 × 4 square "detection matrix" D such that

$$\mathsf{D}_{4\times 4} \sim \begin{bmatrix} \mathsf{\Phi}_1^\mathsf{T} \\ \mathsf{\Phi}_2^\mathsf{T} \\ \mathsf{\Pi}_1^\mathsf{T} \\ \mathsf{\Pi}_2^\mathsf{T} \end{bmatrix}. \qquad (6.1)$$

With the a priori knowledge that $\mathsf{\Phi}_1$ and $\mathsf{\Phi}_2$ correspond to two scene parallel planes, a hypothesis on the detection matrix D can be used to accept or reject the parallelism of the pair of planes, $\mathsf{\Pi}_1$ and $\mathsf{\Pi}_2$. This can be achieved by evaluating the detection matrix D's determinant and rank. In case Π_1 and Π_2 are not parallel, the matrix D will have a non-zero determinant and a full rank of 4. Whereas, if the two tested planes are parallel, the square matrix D will have a zero determinant and rank of 3.

This can be illustrated geometrically. Recall that a line in $3D$ space may be represented by a (2×4) matrix containing the coordinates of two planes[1], as row vectors (see Paragraph 2.1.3). An intuitive geometric notation of this is to consider the $3D$ line as the axes of a pencil of planes, and thus is defined by the intersection of any two planes from the pencil. Consider our two pairs of planes and let the two lines L_ϕ and L_π be their intersections given by

$$\mathsf{L}_\phi \sim \begin{bmatrix} \Phi_1^\mathsf{T} \\ \Phi_2^\mathsf{T} \end{bmatrix}, \mathsf{L}_\pi \sim \begin{bmatrix} \Pi_1^\mathsf{T} \\ \Pi_2^\mathsf{T} \end{bmatrix}$$

The line L_ϕ is the intersection of two parallel plane and thus lies at the infinity plane, Π_∞. Now consider the following two cases:

Parallel pair : In the event that the pair of planes Π_1 and Π_2 are parallel, their intersection line L_π must also lie in the infinity plane Π_∞. Hence, both lines, L_ϕ and L_π, are coplanar. Because coplanar lines are linearly dependant and must intersect in a single point, the detection matrix D will have only three independent equations out of four and thus it is singular and of rank 3. The one-dimensional right null-space of the detection matrix D is the homogenous point resulting from the two lines intersection.

Non-parallel pair: when the two tested planes, Π_1 and Π_2, are not parallel, their corresponding line of intersection L_π does not lie at the infinity plane, but rather intersects the plane at infinity in a single point. This point, in general, is not on the line L_ϕ. As the two lines do not intersect, they are not coplanar and they are linearly independent. Algebraically, the matrix D in this case is not singular and has a full rank of 4. Another way to picture this, as the two planes are not parallel, each of

[1] As planes and points are dual in $3D$, dual representation of a line can be formed with two $3D$ points coordinates as well.

them will intersect the line L_ϕ in a distinct point. In rare cases, discussed in the next paragraph, the point of intersection may be the same, even though the two planes are not parallel, this is a degenerate case.

6.2.2 Degenerate cases

Beside the obvious situation in which a plane is crossing any of the cameras centers, there are some degenerate cases which may occur. If the two tested planes $\mathsf{\Pi}_1$ or $\mathsf{\Pi}_2$ are parallel with the planes $\mathsf{\Phi}_1$ & $\mathsf{\Phi}_2$, the detection matrix D will be of rank 2. This is true as all of the 4 planes are parallel and will intersect at the same line at infinity (i.e. a pencil of planes). In the event that only one of the tested planes, $\mathsf{\Pi}_1$ or Π_2, is parallel with $\mathsf{\Phi}_1$ and $\mathsf{\Phi}_2$, the detection matrix D will have rank 3. This could lead to a false indication that the two tested planes are parallel. However, such a case can be distinguished and be avoided easily by testing the rank of a 3×4 matrix consisting of the line at infinity L_ϕ with each of $\mathsf{\Pi}_1$ and $\mathsf{\Pi}_2$, respectively. If one of the tested planes is parallel to $\mathsf{\Phi}_1$ and $\mathsf{\Phi}_2$, the rank of such a matrix will be 2. This is clear since the three planes in such a case will intersect in the same line at infinity L_ϕ.

The last degenerate case occurs when the line of intersection of the scene planes under testing L_π is parallel to planes $\mathsf{\Phi}_1$ and $\mathsf{\Phi}_2$. In this case, the line L_π intersects our infinity line L_ϕ in a point, and thus both lines L_π and L_ϕ are considered coplanar, but not with respect to the plane at infinity. Such configuration may occur only when all the four planes differ from each other by *a rotation* or *a translation and rotation* around a common single axes (i.e. planar motion). For example, consider any 4 scene planes with different orientation but all are perpendicular to the ground plane, see Figure 6.2. Each arbitrary, but non-parallel, pair of these planes will intersect each other in a line perpendicular to the ground, forming a prism or a pencil of planes, and

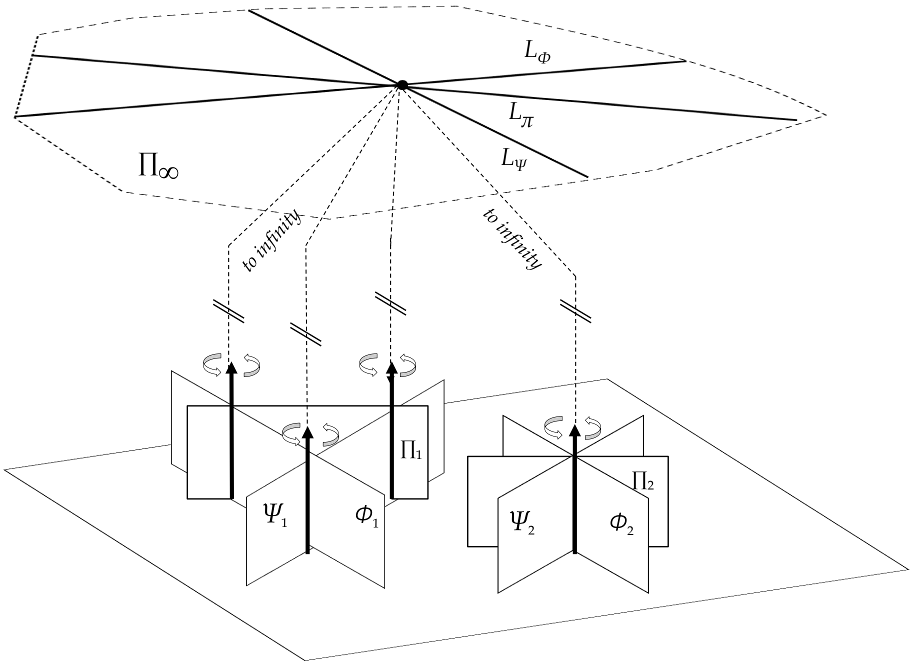


Figure 6.2: *Degenerate case*

thus all of these lines of intersection are parallel and intersect at the same vertical vanishing point at infinity. In Figure 6.2, each pair of planes $\mathsf{\Phi_i}$, $\mathsf{\Pi_i}$, and $\mathsf{\Psi_i}$ are parallel, and thus intersect at the plane at infinity in the infinity lines L_ϕ, L_π and L_ψ respectively. Moreover, these lines L_ϕ, L_π and L_ψ intersect in a single common point. This point represents the direction of the parallel rotational axes between the different planes at the infinity plane. It is important to emphasize that this degeneracy occurs only when all pairs (the pair of known parallel planes and the pair of planes under testing) share a common rotational axis direction. This degenerate case rarely happens in our case when using the pair of parallel planes resulting from the stationary zooming camera.

6.2.3 Dealing with noise

In real scenarios, relying on the algebraical characteristic of the detection matrix, i.e. determinant and rank, turned out to be not practical. This is due to the severe effect of noise on the detection matrix. In addition, approximating the correct rank of a noisy 4×4 matrix is not a reliable solution too, due to its small size.

However, a practical solution is possible by following a geometrical approach as described below. In this approach, each scene plane is represented by a single $3D$ point, $\mathsf{V}_{\pi_\mathsf{i}}$, representing the intersection point of the scene plane Π_i with the known pair of parallel planes. Each point $\mathsf{V}_{\pi_\mathsf{i}}$ can be computed, using SVD for example, as the right null space of the 3×4 matrices formed by the three planes coordinated as row vectors. More specifically, if the two parallel planes coordinates are Φ_1 and Φ_2, the point $\mathsf{V}_{\pi_\mathsf{i}}$ satisfies

$$\begin{bmatrix} \mathsf{\Phi}_1^\mathsf{T} \\ \mathsf{\Phi}_2^\mathsf{T} \\ \mathsf{\Pi}_\mathsf{i}^\mathsf{T} \end{bmatrix} \mathsf{V}_{\pi_\mathsf{i}} = \begin{bmatrix} \mathsf{L}_\phi \\ \mathsf{\Pi}_\mathsf{i}{}^T \end{bmatrix} \mathsf{V}_{\pi_\mathsf{i}} = \ 0 \tag{6.2}$$

Note that the obtained ideal points $\mathsf{V}_{\pi_\mathsf{i}}$ are all collinear as each point lies on the infinity line L_ϕ.

The constraint for a pair of parallel planes can be reformulated as follow. Two planes Π_i and Π_j are parallel if their corresponding computed $\mathsf{V}_{\pi_\mathsf{i}}$ and $\mathsf{V}_{\pi_\mathsf{j}}$ are equal. A geometric interpretation of this is that when the two tested planes are parallel, they intersect in a line at infinity which must intersect L_ϕ in this single point and thus our two obtained points $\mathsf{V}_{\pi_\mathsf{i}}$ and $\mathsf{V}_{\pi_\mathsf{j}}$ are nothing but the same point.

In practice and because of noise, two ideal points,$\mathsf{V}_{\pi_\mathsf{i}}$ and $\mathsf{V}_{\pi_\mathsf{j}}$, corresponding to a pair of parallel planes will not be perfectly identical but should be very close to eacah other. Because the distance between points in projective space cannot be quantified in the usual sense, these $3D$ points are first projected on an image plane then, their pixel distance is estimated.

By doing this way, the projection of the image points (pixels) of two or more ideal points can be considered equal if the distance between them is below a certain threshold. In this case, their corresponding planes will be considered parallel. Such a threshold depends on the image size, camera orientation, level of noise, as well as the image location of the vanishing points. To use a uniform threshold that works for all situations, it is important to use normalized image coordinates, which can be easily done with a simple transformation of the image coordinates [38].

In our case, by using such normalization, a threshold of 10^{-3} was a good choice for detecting parallel planes in all of our experiments on both simulated and real data.

6.2.4 Automatic a priori obtainment

The automatic identification method discussed above requires a pair of parallel planes as an a priori. In the case of stationary zooming camera, such an a priori can be obtained from a pair of principal planes of two images at different zoom settings, and thus the whole method can be carried on automatically.

However, it is worth mentioning that the proposed method is not restricted to the case of zooming cameras only. For example, it is possible to obtain a pair of parallel planes in the case of a translating camera. A translating camera preserves its orientation, and thus the principal planes of a translating camera are also a pair of parallel planes, regardless of the constancy of the camera intrinsic parameters.

Another practical case is the case of vanishing points. The knowledge of two vanishing points can be translated to knowledge of two parallel planes. It is possible to *automatically* detect vanishing points or vanishing lines from perspective images. In many situations, however, only two vanishing points or a single vanishing line can be identified in these images. Such knowledge alone is not enough to allow affine upgrade of the projectively reconstructed scene. The proposed method can complement this by detecting possibly parallel planes in the scene and hence provide enough constraints for recovering the affine reconstruction.

The knowledge of two vanishing points or a single vanishing line can be mapped easily into the coordinates of a pair of virtual parallel planes. This can be achieved by reconstructing the two vanishing points, by triangulation from two images, to obtain their 3D coordinates. The two reconstructed vanishing points define a $3D$ line L_∞ contained within the infinity plane of the projective space $\mathbb{P}^3$. Consider the family (pencil) of planes passing through the axis line L_∞. Any arbitrary, but distinct, pair of planes from this pencil must correspond to two parallel world planes in the Euclidian space $\mathbb{E}^3$. Consequently, we may obtain the coordinates of two parallel planes by picking any two arbitrary $3D$ points, e.g. from the reconstructed scene points set, such that these two points together with the two vanishing points are not all coplanar. Each plane can be computed from the triplet of points constituted by both vanishing points and each of the selected scene points respectively.

6.3 Simulation

Experiments with simulated data were conducted to evaluate the proposed method with different level of pixel noise. Each generated scene consists of 10 planes each formed by 50 randomly generated scene points, distributed randomly on a disc of radius 1. The first and second generated planes are parallel with a mean distance of 0.5 units between them and a 0.15 standard deviation. In all the experiments, the center of the disc on each of the remaining 8 planes was placed at random orientation in front of the randomly generated cameras (see Figure 6.3).

In each simulation, two cameras were generated where each one was roughly pointing to the center of the scene. The cameras were randomly located at a mean distance of 3 units, from the scene center, with a 0.25 standard deviation. The first images were captured with camera parameters of a 12.5mm focal length, zero skew, unit aspect ratio, and the principal point located at the center of the image. The zooming cameras were created to simulate a camera with a zoom range of 12.5mm - 35mm, a CCD array of 8×8 mm and a 64 pixels per millimeter. In each simulation, the zooming camera takes a second image at different zoom setting, where the focal length is randomly increased to a value within the range of 15mm-35mm. The optical center of the camera is translated by a relative amount within the range of 2.5mm - 22.5mm along the optical axis from its initial position before zooming.

For each scene and camera, we progressively corrupt the pixel coordinates by a zero-mean Gaussian noise with standard deviation in the range 0 to 2 pixels (with a 0.25 pixel step). In each trial, each possible pair of planes is evaluated by the proposed method. The principal planes of the zoom cameras at different zoom settings were used to provide the a priori pair of known parallel planes. Using the principal planes from the projection matrices of a single zooming camera, we compute the ideal point V_{π_i} for each scene plane as described in equation (6.2). These computed 3D points are then projected on the reference image plane as vanishing points. Using normalized image coordinates in all calculations, a threshold of 10^{-3} is used to distinguish parallel from non-parallel planes, as described in Paragraph 6.2.3.

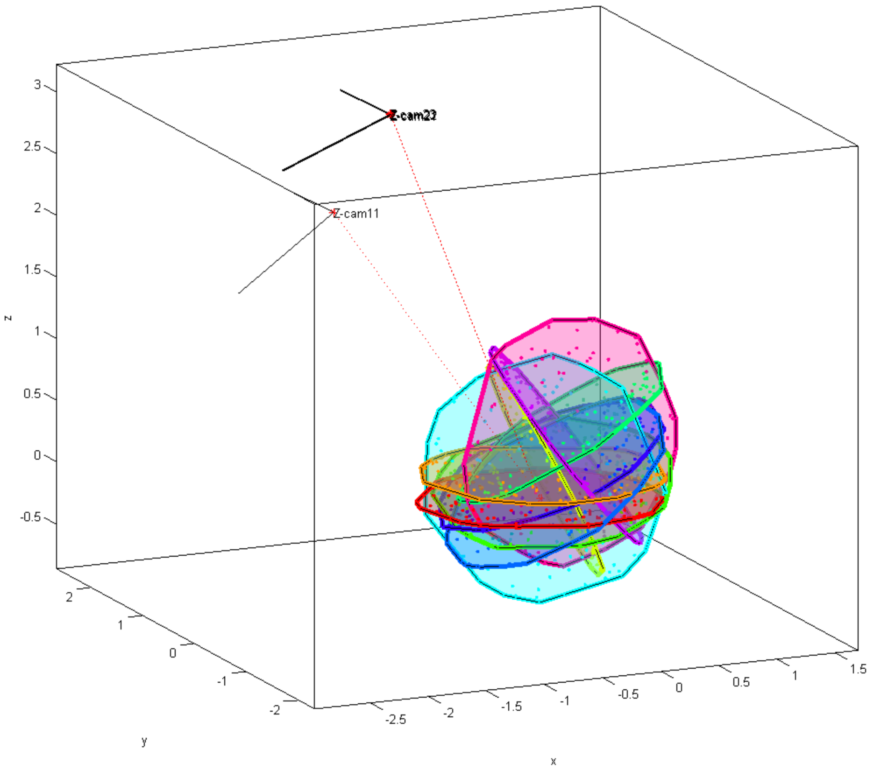


Figure 6.3: *Simulations setup showing the generated planes and cameras*

Table 6.1 summarizes the obtained results of our simulations, with a 1000 trials for each noise level, and the corresponding value in each raw represents the success rate (%) in identifying the sought pair of parallel planes. The results show that the success rate slightly decreases with the increase of pixel noise. However, the success rate for a realistic 1-pixel noise is above 85%. Even when the noise level reaches 2 pixels, the success rate is still satisfactory at 70%.

Table 6.1: Success rate (%) for the detection of parallel planes

Noise(pixels)	0.0	0.25	0.5	0.75	1.0	1.25	1.5	1.75	2.0
Success (%)	100.00 %	99.00 %	95.50 %	93.25 %	85.25 %	82.00 %	82.00 %	75.50 %	68.25%

6.4 Experiments using real scenes

The proposed method has been tested for both indoor laboratory and outdoor scenes. We have used off-the-shelf low-end cameras with motorized zoom lenses to capture our images. Common among all of the conducted tests, 3 images were taken with two different orientations, i.e., motion and rotation. The first image is taken by a camera oriented toward the scene, followed by two images taken from a single stationary zooming camera fixed on a tripod at two different zoom settings. In all our experiments, we have used the linear method reported in [82] to compute a consistent set of projection matrices for all acquired images. From these projective matrices, we use the principal plane coordinates, i.e., the third row, of the matrices of the 2 zoom images as our a priori known parallel planes.

For each test, a set of parallel and non-parallel planes in the scene have been manually collected and used as ground truth to compare against our method result. Matched pixels across the images have a realistic noise level around 0.5-pixel on average. Using the computed parallel planes deduced from the 2 zooming camera projection matrices, we compute the ideal points V_{π_i} for each scene plane, as described in Equation 6.2. These computed 3D ideal points are then projected on the image plane. Using normalized image coordinates, a threshold of 10^{-3} in all of our experiments has been used to distinguish parallel from non-parallel planes.

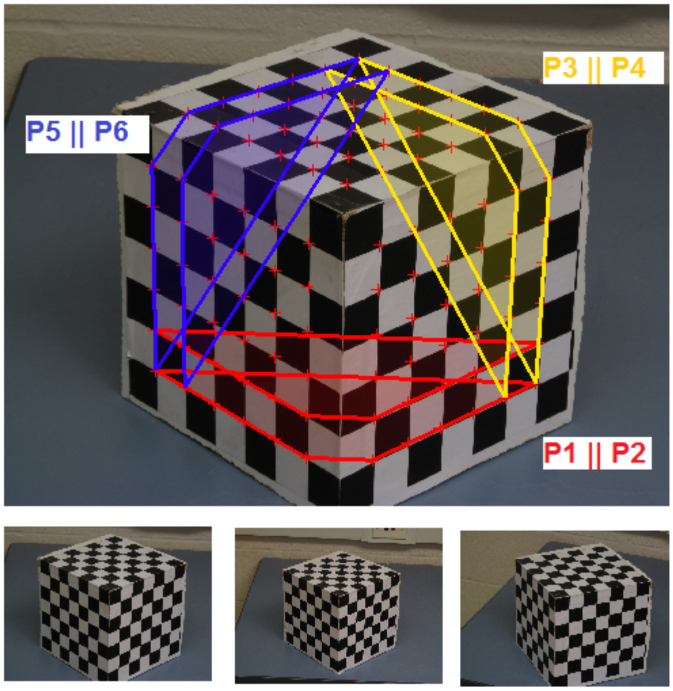


Figure 6.4: *Cube object images and selected planes.*

6.4.1 Indoor tests

The Cube Model:

In this laboratory experiment, we have used a calibration cube scene model shown in Figure 6.4. All the three images in this test were taken using a low-end camera, Canon PowerShot SX150, fixed on a tripod. A total of 108 feature points scattered on the three different sides of the cube were extracted and matched across the 3 views. Points on three orthogonal pairs of virtual planes are chosen as illustrated in Figure 6.4 (top), where we have three pairs of parallel planes, $P_1 \| P2$, $P3 \| P4$ and $P5 \| P6$. Each of these planes consists of 12 points spanning two sides of the cube. Our method successfully detected all the parallel pairs. The computed distances between pairs of vanishing points belonging to each pair of the parallel planes $P1 \| P2$, $P3 \| P4$ and $P5 \| P6$ were 0.000849 , 0.000125 and 0.0002, respectively. On the other hand, the distances between vanishing points of non-parallel pairs were very high. For example, for the non parallel pairs $P1 \nparallel P3$, $P1 \nparallel P5$ and $P3 \nparallel P5$, these values were 0.26, 0.488 and 0.2333 respectively.

The same experiments were repeated but instead of using the parallel planes deduced from the stationary zooming camera images, we used randomly 2 parallel scene planes to test the other remaining planes. All parallel and non-parallel pairs were successfully detected, regardless of which pair of parallel planes used as the a priori known pair.

Cylinder Model: In this set of experiments, 3 images were taken of a cylindrical scene model as shown in Figure 6.5. All images were taken by the same digital camera *(Canon PowerShot SX150 IS)* fixed on a tripod were the last 2 images are taken from the same position but with different zoom settings. A total of 35 points were extracted and matched across the different images as shown in the same figure. As there are

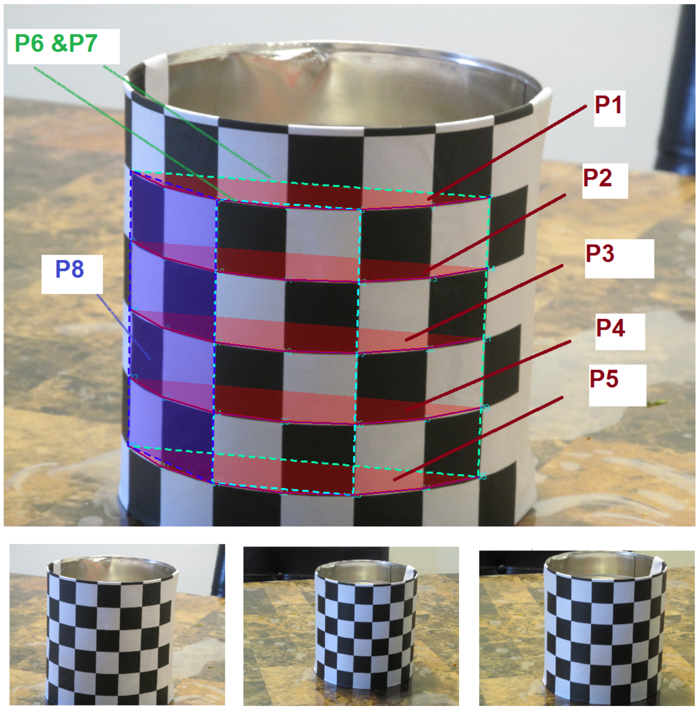


Figure 6.5: *The used cylindrical object and selected virtual planes.*

no physical planes in the scene, several virtual planes has been manually identified and used for testing (see Figure 6.5). Each of the labeled planes $P1$, $P2$,..., $P5$ are mutually parallel to each other, and consist of 6 points. Furthermore, Planes $P6$ & $P7$ are parallel while P8 is not parallel with any of the other planes. The result of detecting every two consecutive planes of $P1 \ldots P5$ were successfully detected as parallel with an error distance in the range 0.0002-0.0004, well below our set threshold of 10^{-3}. However, in two cases of non-consecutive planes, $P1$ with $P5$ and $P2$ with $P5$, the error distance exceeded slightly the threshold at 0.003 and 0.005, and failed to detect their parallelism. This is acceptable as these virtual planes consist of 6 points only, making the estimation of the plane sensitive to noise. For the parallel planes $P6$ and $P7$ the difference were close to zero 0.00000032. The difference between $P8$

and all the other planes exceeded the threshold and did not yield any false positive with any other plane.

6.4.2 Outdoor tests

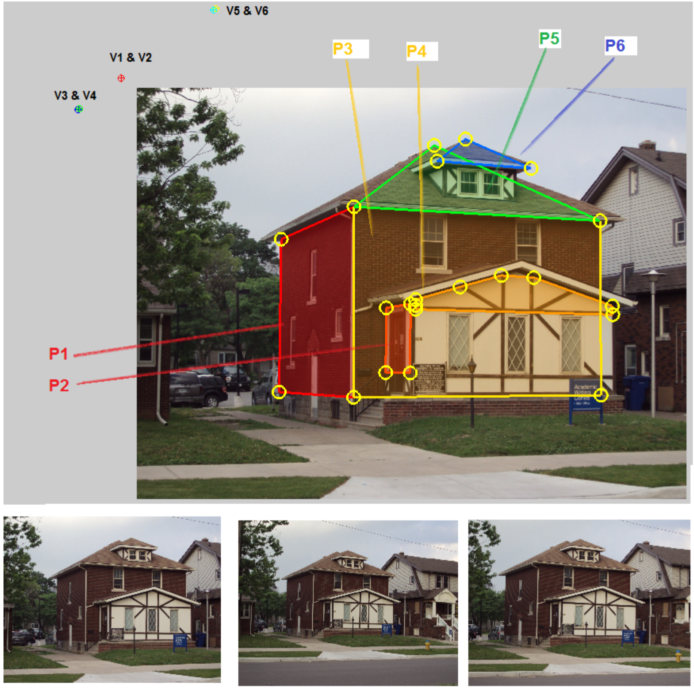


Figure 6.6: *House images and chosen planes.*

House Scene : In this set of experiments, our camera, a *(Sony DSC-S930)*, was placed at approximately 25 meters off a house. We have captured 2 images at different zoom settings and a third image was taken from a different locations (see Figure 6.6). A total of 45 points were extracted and matched across the different images and 6 different planes has been identified and labeled (see Figure 6.6-top). The computed distances between the vanishing points corresponding to the parallel pairs $P1||P2$, $P3||P4$, and $P5||P6$ were 0.0002, 0.000698 and 0.000036, respectively. On the other hand, the distance between vanishing points of non-parallel pairs were very high. For example, the distance between the vanishing points of the non-parallel pairs $P1 \nparallel P4$, $P1 \nparallel P6$ and $P4 \nparallel P6$ were 0.841, 1.2 and 0.416, respectively, which clearly exceed our set threshold of 10^{-3}. As an example, the vanishing points of each plane are plotted on the top left corner of Figure 6.6. Note that these points are collinear, as they all belong to the same line at infinity.

Round building Scene :

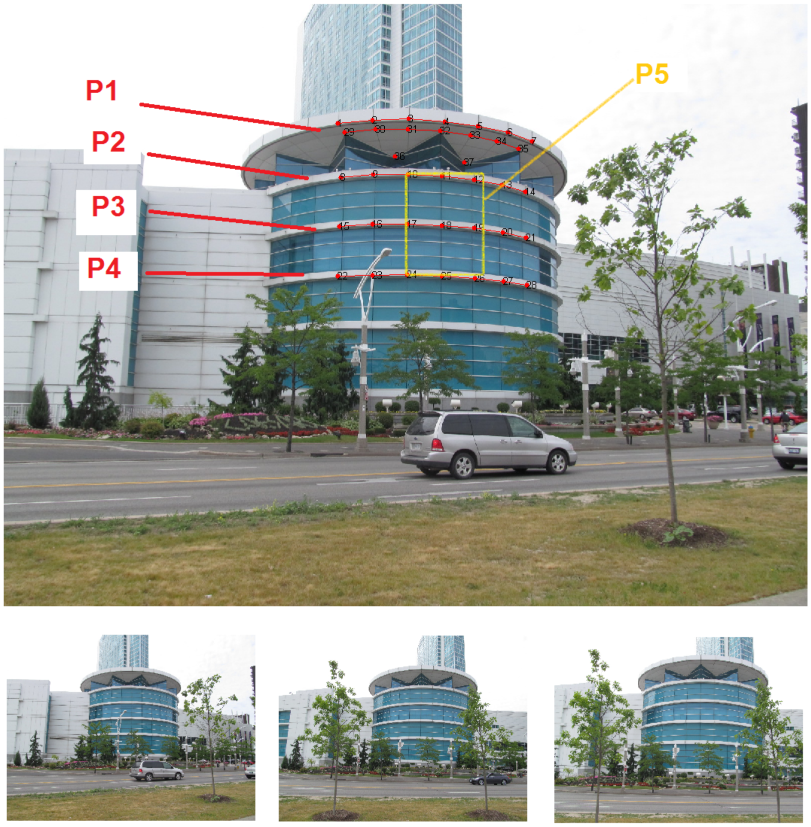


Figure 6.7: *Images of a round building used in our experiments.*

This is a cylindrical building that was photographed by the *(Canon PowerShot SX150 IS)* camera. A total of 37 points were extracted and matched across the 3 images and 5 virtual planes where selected and labeled (see Figure 6.7). Clearly, planes $P1$, $P2$, $P3$ and $P4$ are parallel while plane $P5$ is not parallel to any of the others. The computed difference of any two vanishing points among the parallel planes, i.e. $P1$,...,$P4$, were between 0.0007 and 0.00002 and thus below the threshold and the method detected them correctly as parallel. Even the non-consecutive planes have been correctly identified parallel. On the other hand, the difference among these planes and the non-parallel plane $P5$ were very high, 1.79 on average, which strongly indicates non-parallelism with the other planes.

6.5 Conclusion

A novel automatic method for identifying parallel planes from uncalibrated views of a scene was developed. In particular, a sufficient condition for identifying parallel planes from the projective reconstruction of the scene was devised. The method utilizes a priori knowledge of a pair of parallel planes in order to automatically detect the other scene parallel planes. Such an a priori pair of known parallel planes is always possible when using zoom images of a stationary non-rotating camera, making the method fully automatic. Moreover, the method is flexible to work in other situations such as, when two vanishing points or a vanishing line can be provided or detected automatically. We have also provided a practical way to deal with image noise. Our extensive experiments on simulated data and on indoor and outdoor real scenes have yielded excellent results despite the use of low-end cameras and noisy images.

Chapter 7

Conclusion

The work presented in this dissertation is primarily concerned with the problem of auto-calibration and 3D reconstruction for a vision system consisting of stationary non-rotating zooming cameras. This is a common configuration which is often encountered in stereo camera systems such as, surveillance networks and monitoring of events. In such image capture systems, each camera is physically attached to a static structure (wall, ceiling or tripod) and is only allowed to zoom.

While several approaches have been developed in the past for vision systems with fixed settings, little work has been done in the context of active vision systems with zoom lenses. Active vision systems with zoom capabilities allow to adjust the captured images or videos to perform tasks which are not possible for vision systems with fixed lenses. For example, a zoom-out allows analysis of a wide scene, while a zoom-in helps taking a closer look at an object of interest. However, integrating zoom lenses into vision systems introduces many difficulties. On the top of these difficulties comes the need to auto-calibrate the vision system as all reconstruction techniques require some form of calibration. The metric auto-calibration problem of a system with cameras of different settings is known to be a hard non-linear problem that may often fail. A simple and linear solution for such problem would be of great importance.

In order to linearize the solution, a stratified auto-calibration approach has been adopted. A stratified approach can bridge the gap between projective and metric or Euclidean structure, by obtaining an affine calibration in a first stage. This is equivalent to locating the plane at infinity for a given projective structure. Existing techniques for locating the plane at infinity are either based on restricted camera motion or depends on explicit scene constraints. When the cameras are supposed to be stationary and non-rotating, the special camera motion case cannot be applied. In addition, scene constraints may or may not exist in the scenes and require an automatic and reliable identification methods.

In this work, a new theoretical insight on the zooming process is developed, enabling approaches which simplify the auto-calibration and three-dimensional reconstruction problem. In particular, we have shown that a stationary zooming camera allows the identification of parallel planes. We have translated this observation in to the following practical methods:

- A *linear* method to compute the affine 3D structure, using a *stereo* zooming camera system, was developed. Based on the valid observation that, the principal planes, before and after zooming, provide a pair of parallel planes, we were able to extract linear constraints on the plane at infinity. Two such pairs of parallel planes, from the stereo pair of cameras, are enough to identify the plane at infinity, making it possible for the scene's projective reconstruction to be upgraded to affine structure. Typically, affine calibration from a stereo pair of stationary cameras with unknown orientation is not possible without scene or camera motion constraints, even for cameras with fixed parameters (i.e. non-zooming). Note that affine camera calibration and structure may be enough for many computer vision tasks. This research work has resulted into a refereed conference article at the IEEE International Conference on Image Processing (ICIP).

- Based on the assumption that typical cameras have rectangular or even square pixels, a stratified auto-calibration approach was proposed. This has allowed the recovery of the intrinsic camera parameters, and thus metric measurements and structure of the scene have become possible. The previous method for locating the plane at infinity was extended from its minimal case of a pair of cameras and zoom images to the case of more cameras and more zoom images. Such extension was necessary to obtain successful results under noisy conditions. Two linear methods, based on SVD and LMI, were investigated and tested for estimating the camera parameters. The obtained results were very good on both synthetic and on real images. This research work has resulted into a refereed journal paper at Image and Vision Computing (Elsevier).

- A method for automatically identifying parallel planes in a scene, using zooming cameras, was developed. Given a priori knowledge about a single pair of parallel planes, a sufficient linear condition was devised and successfully applied to identify other scene's parallel planes. In this method, we have used the pair of parallel planes, resulting from two zoom images, as the required a priori, to automatically identify parallel planes in the scene. This research work has resulted into a refereed conference article at the IEEE Canadian Conference on Computer and Robot Vision (CRV).

In our future work, we may investigate lens radial distortion and and their impact on the obtained results. Lens radial distortion estimation can be done by assuming a constant radial distortion model that can be estimated at a particular zoom level.

Then, its variation can be modeled using a magnification factor. Taking lens distortion into account may improve both the success of the method and the quality of the results.

Another important area for future research is the integration of these methods with other auto-calibration techniques. Our proposed methods can fit naturally within the Pan-Tilt-Zoom (PTZ) camera networks. Auto-calibration of a stationary and rotating camera around its center are attractive because of their linear solutions. However, moving cameras in a pure rotational motion is not feasible in practice, due to rotation misalignment, and such an assumption is only plausible when the camera is far from the scene. The integration of the proposed zoom-based auto-calibration methods with the existing camera PTZ rotational based auto-calibration techniques may provide more flexible and reliable solutions.

References

[1] Y. I. Abdel-Aziz and H.M. Karara. Direct linear transformation from comparator coordinates into object space coordinates in close-range photogrammetry. In *Proceedings of the Symposium on Close-Range photogrammetry*, pages 1–18. American Society of Photogrammetry, 1971.

[2] A. Amintabar and B. Boufama. The distinction between virtual and physical planes using homography. *International Conference on Image Analysis and Recognition*, pages 727–736, 2009.

[3] M. Armstrong, A. Zisserman, and P. Beardsley. Euclidean structure from uncalibrated images. In *Proceedings of the Conference on British Machine Vision*, pages 509–518, 1994.

[4] M. Armstrong, A. Zisserman, and R.I. Hartley. Self-calibration from image triplets. In *Proceeedings of the Fourth European Conference on Computer Vision, ECCV'96*, Lecture Notes in Computer Science, pages 1–16. Springer-Verlag, 1996.

[5] A. Azarbayejani and A. P. Pentland. Recursive estimation of motion, structure, and focal length. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(6):562–575, 1995.

[6] P. A. Beardsley, A. Zisserman, and D. W. Murray. Navigation using affine structure from motion. In *Proceedings of the Third European Conference on Computer Vision, ECCV'94*, Lecture Notes in Computer Science, pages 85–96. Springer, 1994.

[7] D. Bondyfalat, T. Papdopoulo, and B. Mourrain. Using scene constraints during the calibration procedure. In *Proceedings of the Eighth IEEE International Conference on Computer Vision, ICCV'01*, pages 124–130, 2001.

[8] B. Boufama and S. Bouakaz. The use of constraints for calibration-free 3d metric reconstruction: From theory to applications. *Emerging Topics in Computer Vision and Its Applications*, 1:337, 2012.

[9] B. Boufama and R. Mohr. Epipole and fundamental matrix estimation using virtual parallax. In *Proceedings of the Fifth International Conference on Computer Vision, ICCV'95*, pages 1030–1036. IEEE, 1995.

[10] S. Bougnoux. From projective to euclidean space under any practical situation, a criticism of self-calibration. In *Proceedings of the Sixth International Conference on Computer Vision, ICCV'98*, pages 790–798, 1998.

[11] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[12] B. Caprile and V. Torre. Using vanishing points for camera calibration. *International Journal of Computer Vision*, 4(2):127–140, 1990.

[13] M. Chandraker, S. Agarwal, F. Kahl, D. Kriegman, and D. Nistér. Autocalibration via rank-constrained estimation of the absolute quadric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR'07*, pages 1–8, 2007.

[14] A. Criminisi, I. Reid, and A. Zisserman. Single view metrology. *International Journal of Computer Vision*, 40(2):123–148, 2000.

[15] Suochao Cui and Xiao Zhu. A generalized reference-plane-based calibration method in optical triangular profilometry. *Optics Express*, 17(23):20735–20746, 2009.

[16] L. De Agapito, E. Hayman, and R.I. Hartley. Linear self-calibration of a rotating and zooming camera. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR'99*, pages 15–21, 1999.

[17] L. De Agapito, E. Hayman, and I. Reid. Self-calibration of a rotating camera with varying intrinsic parameters. In *Proceedings of the Ninth British Machine and Vision Conferance, BMVC'98*, pages 105–114, 1998.

[18] L. De Agapito, E. Hayman, and I. Reid. Self-calibration of rotating and zooming cameras. *International Journal of Computer Vision, IJCV'01*, 45(2):107–127, 2001.

[19] T. Elamsy, B. Boufama, and A Habed. Parallel planes identification using uncalibrated zooming cameras. In *Proceedings of the International Conference on Computer and Robot Vision, CRV'13*, pages 174–180, 2013.

[20] T. Elamsy, A. Habed, and B. Boufama. A new method for linear affine self-calibration of stationary zooming stereo cameras. In *Proceeding of 19th IEEE International Conference on Image Processing, ICIP'12*, pages 353–356, 2012.

[21] T. Elamsy, A. Habed, and B. Boufama. Self-calibration of stationary non-rotating zooming cameras. *Image and Vision Computing*, 32(3):212–226, 2014.

[22] F. Espuny. A new linear method for camera self-calibration with planar motion. *Journal of Mathematical Imaging and Vision*, 27(1):81–88, 2007.

[23] F. Espuny, J. Aranda, and J. I. Burgos Gil. Camera self-calibration with parallel screw axis motion by intersecting imaged horopters. In *Proceedings of the Seventeenth Scandinavian Conference on Image Analysis*, pages 1–12. Springer-Verlag, 2011.

[24] O.D. Faugeras, Q.-T. Luong, and S.J. Maybank. Camera self-calibration: Theory and experiments. In *Proceedings of Second European Conference on Computer Vision, ECCV'92*, pages 321–334. Springer, 1992.

[25] O.D. Faugeras and T. Papadopoulo. A nonlinear method for estimating the projective geometry of 3 views. In *Proceedings of the Sixth International Conference on Computer Vision, ICCV'98.*, pages 477–484, 1998.

[26] A. Fitzgibbon and A. Zisserman. Automatic camera recovery for closed or open image sequences. In *Proceedings of the Fifth European Conference on Computer Vision*, pages 311–326. Springer-Verlag, 1998.

[27] S. Ganapathy. Decomposition of transformation matrices for robot vision. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 130–139, 1984.

[28] A. Gardel, J. L. Lázaro, and J. M. Lavest. Influence of mechanical errors in a zoom camera. *Image Analysis Stereol*, 22:21–25, 2003.

[29] R. Gherardi and A. Fusiello. Practical autocalibration. In *Proceedings of the European Conference on Computer Vision*, Lecture Notes in Computer Science, pages 790–801. Springer-verlag, 2010.

[30] A. Habed, K. Al Ismaeil, and D. Fofi. A new set of quartic trivariate polynomial equations for stratified camera self-calibration under zero-skew and constant parameters assumptions. In *Proceedings of the European Conference on Computer Vision, ECCV'12*, Lecture Notes in Computer Science, pages 710–723. Springer-Verlag, 2012.

[31] A. Habed, A. Amintabar, and B. Boufama. Affine camera calibration from homographies of parallel planes. In *Proceeding of the IEEE Seventeenth International Conference on Image Processing, ICIP'10*, pages 4249–4252. IEEE, 2010.

[32] A. Habed, A. Amintabar, and B. Boufama. Reconstruction-free parallel planes identification from uncalibrated images. In *Proceedings of the 20th International Conference on Pattern Recognition, ICPR'10*, pages 1828–1831, 2010.

[33] A. Habed and B. Boufama. Camera self-calibration from bivariate polynomials derived from kruppa's equations. *Pattern Recognition*, 41(8):2484–2492, 2008.

[34] Pär Hammarstedt, F. Kahl, and A. Heyden. Affine reconstruction from translational motion under various autocalibration constraints. *Journal of Mathematical Imaging and Vision*, 24(2):245–257, 2006.

[35] R.I. Hartley. Estimation of relative camera positions for uncalibrated cameras. In G. Sandini, editor, *Computer Vision ECCV'92*, volume 588 of *Lecture Notes in Computer Science*, pages 579–587. Springer Berlin Heidelberg, 1992.

[36] R.I. Hartley. Cheirality invariants. In *Proceedings DARPA Image Understanding Workshop*, pages 745–753, 1993.

[37] R.I. Hartley. Euclidean reconstruction from uncalibrated views. In *Applications of Invariance in Computer Vision*, pages 237–256. Springer-Verlag, 1993.

[38] R.I. Hartley. In defense of the eight-point algorithm. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 19(6):580–593, 1997.

[39] R.I. Hartley. Kruppa's equations derived from the fundamental matrix. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(2):133–135, 1997.

[40] R.I. Hartley. Self-calibration of stationary cameras. *International Journal of Computer Vision*, 22:5–23, 1997.

[41] R.I. Hartley, E. Hayman, L. de Agapito, and I. Reid. Camera calibration and the search for infinity. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 510–517, 1999.

[42] R.I. Hartley and R. Kaucic. Sensitivity of calibration to principal point position. In *Proceedings of the European Conference on Computer Vision*, pages 433–446. Springer-Verlag, 2002.

[43] R.I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision.* Cambridge University Press, second edition, 2004.

[44] E. Hayman and D.W. Murray. The effects of translational misalignment when self-calibrating rotating and zooming cameras. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25:1015–1020, 2003.

[45] M. Hebert, C. Zeller, O.D. Faugeras, and L. Robert. Applications of non-metric vision to some visually guided robotics tasks. Technical Report RR-2584, INRIA, 1995.

[46] J. Heikkila. Geometric camera calibration using circular control points. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(10):1066–1077, 2000.

[47] A. Heyden and K. Åström. Euclidean reconstruction from constant intrinsic parameters. In *Proceedings of the Thirteenth International Conference on Pattern Recognition, ICPR'96*, pages 339–343, 1996.

[48] A. Heyden and K. Åström. Euclidean reconstruction from image sequences with varying and unknown focal length and principal point. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR'97*, pages 438–443, 1997.

[49] A. Heyden and K. Åström. Minimal conditions on intrinsic parameters for euclidean reconstruction. In Roland Chin and Ting-Chuen Pong, editors, *Computer Vision ACCV'98*, volume 1352 of *Lecture Notes in Computer Science*, pages 169–176. Springer Berlin Heidelberg, 1997.

[50] A. Heyden and K. Åström. Flexible calibration: Minimal cases for autocalibration. In *Proceedings of the Seventh IEEE International Conference on Computer Vision, ICCV'99*, pages 350–355. IEEE, 1999.

[51] Z. Hu and Z. Tan. Depth recovery and affine reconstruction under camera pure translation. *Pattern Recogniton*, 40(10):2826–2836, 2007.

[52] Z. Hu, F. Wu, and G. Wang. The impossibility of affine reconstruction from perspective image pairs obtained by a translating camera with varying parameters. *Pattern Recognition Letters*, 24(16):2909–2911, 2003.

[53] D. Q. Huynh and A. Heyden. Scene point constraints in camera autocalibration: an implementational perspective. *Image and Vision Computing*, 23(8):747–760, 2005.

[54] Q. Ji and S. Dai. Self-calibration of a rotating camera with a translational offset. *IEEE Transactions on Robotics and Automation*, 20(1):1–14, 2004.

[55] Z. Jiang, S. Guo, and L. Jia. Sequences images based camera self-calibration method. In Y. Zhang, Z.-H. Zhou, C. Zhang, and Y. Li, editors, *Intelligent Science and Intelligent Data Engineering*, volume 7202 of *Lecture Notes in Computer Science*, pages 538–545. Springer Berlin Heidelberg, 2012.

[56] F. Kahl, B. Triggs, and K. Åström. Critical motions for auto-calibration when some intrinsic parameters can vary. *Journal of Mathematical Imaging and Vision*, 13(2):131–146, 2000.

[57] S. M. Khan and M. Shah. Tracking multiple occluding people by localizing on multiple scene planes source. *IEEE transactions on pattern analysis and machine intelligence*, 31(3):505–519, 2009.

[58] J. Knight, A. Zisserman, and I. Reid. Linear auto-calibration for ground plane motion. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR'03*, pages 503–510. IEEE Computer Society, 2003.

[59] H. Li and C. Shen. An LMI approach for reliable PTZ camera self-calibration. In *Proceedings of the IEEE International Conference on Video and Signal Based Surveillance, AVSS'06*. IEEE Computer Society, 2006.

[60] D. Liebowitz and A. Zisserman. Metric rectification for perspective images of planes. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR'98*, pages 482–488, 1998.

[61] HC Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. In *Readings in computer vision: issues, problems, principles, and paradigms.*, pages 61–62. Morgan Kaufmann Publishers Inc., 1987.

[62] M.I.A. Lourakis, A.A. Argyros, and S.C. Orphanoudakis. Detecting planes in an uncalibrated image pair. In *Proceedings of the British Machine Vision Conferance, BMVC'02*, pages 1–10, 2002.

[63] M.I.A. Lourakis and R. Deriche. Camera self-calibration using the kruppa equations and the svd of the fundamental matrix: The case of varying intrinsic parameters. Technical Report RR-3911, INRIA, 2000.

[64] Q.-T. Luong and O.D. Faugeras. Self-calibration of a moving camera from point correspondences and fundamental matrices. *International Journal of Computer Vision*, 22(3):261–289, 1997.

[65] Q.-T. Luong and T. Viville. Canonical representations for the geometries of multiple projective views. *Computer Vision and Image Understanding*, 64(2):193–229, 1996.

[66] S.J. Maybank and O.D. Faugeras. A theory of self-calibration of a moving camera. *International Journal of Computer Vision*, 8(2):123–151, 1992.

[67] T. Moons, L. Van Gool, M. Proesmans, and E. Pauwels. Affine reconstruction from perspective image pairs with a relative object-camera translation in between. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 18(1):77–83, 1996.

[68] T. Moons, L. Van Gool, M. Van Diest, and E. Pauwels. Affine reconstruction from perspective image pairs obtained by a translating camera. In *Applications of invariance in computer vision*, pages 297–316. Springer, 1994.

[69] D. Nistér. Untwisting a projective reconstruction. *International Journal of Computer Vision*, 60(2):165–183, 2004.

[70] T. Pajdla and V. Hlavac. Camera calibration and euclidean reconstruction from known observer translations. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR'98*, pages 421–426, 1998.

[71] O.A. Pellejero, C. Sagiies, and J. J. Guerrero. Automatic computation of the fundamental matrix from matched lines. In *Proceedings of the 10th Conference on Current Topics in Artificial Intelligence: of the Spanish Association for Artificial Intelligence, CAEPIA'03, and the 5th Conference on Technology Transfer, TTIA'03, San Sebastian, Spain, November 12-14, 2003. Revised Selected Papers*, pages 197–206. Springer, 2004.

[72] M. Pollefeys, R. Koch, and L. Van Gool. Self-calibration and metric reconstruction in spite of varying and unknown internal camera parameters. *International Journal Of Computer Vision*, pages 7–25, 1999.

[73] M. Pollefeys and L. Van Gool. A stratified approach to metric self-calibration. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR'97*, pages 407–412, 1997.

[74] M. Pollefeys and L. Van Gool. Stratified self-calibration with the modulus constraint. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(8), 1999.

[75] M. Pollefeys, L. Van Gool, and A. Oosterlinck. The modulus constraint: a new constraint self-calibration. In *Proceedings of the Thirteenth International Conference on Pattern Recognition, ICPR'96*, pages 349–353, 1996.

[76] M. Pollefeys, L. Van Gool, and M. Proesmans. Euclidean 3d reconstruction from image sequences with variable focal lenghts. In *Proceedings of the European Conference on Computer Vision*, pages 31–42. Springer-Verlag, 1996.

[77] J. Ponce, K. McHenry, T. Papadopoulo, M. Teillaud, and B. Triggs. On the absolute quadratic complex and its application to autocalibration. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR'05*, pages 780–787. IEEE, 2005.

[78] W.H. Press. *Numerical recipes Third edition: The art of scientific computing.* Cambridge University Press, 2007.

[79] F. Rameau, A. Habed, C. Demonceaux, D. Sidib, and D. Fofi. Self-calibration of a ptz camera using new lmi constraints. In K. Lee, Y. Matsushita, J. Rehg, and Z. Hu, editors, *Computer Vision ACCV 2012*, volume 7727 of *Lecture Notes in Computer Science*, pages 297–308. Springer Berlin Heidelberg, 2013.

[80] J. Ronda, A. Valds, and G. Gallego. Line geometry and camera autocalibration. *Journal of Mathematical Imaging and Vision*, 32(2):193–214, 2008.

[81] J. Ronda, A. Valds, and G. Gallego. Autocalibration with the minimum number of cameras with known pixel shape. *Journal of Mathematical Imaging and Vision*, pages 1–20, 2014.

[82] C. A. Rothwell, O.D. Faugeras, and G. Csurka. Different paths towards projective reconstruction. In *Proceedings of the Europe-China Workshop on Geometrical Modelling and Invariants for Computer Vision*. Xidan University Press, 1995.

[83] A. Ruf, G. Csurka, and R. Horaud. Projective translations and affine stereo calibration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR'98*, pages 475–481. IEEE Computer Society Press, 1998.

[84] F. Schaffalitzky. Direct solution of modulus constraints. In *Proceedings of the Indian Conference on Computer Vision, ICCV'00*, pages 314–321, 2000.

[85] F. Schaffalitzky and A. Zisserman. Geometric grouping of repeated elements within images. In *Shape, Contour and Grouping in Computer Vision*, pages 165–181. Springer, 1999.

[86] F. Schaffalitzky and A. Zisserman. Planar grouping for automatic detection of vanishing lines and points. *Image and Vision Computing*, 18(9):647–658, 2000.

[87] K.S. Seo, J.H. Lee, and H.M. Choi. An efficient detection of vanishing points using inverted coordinates image space. *Pattern Recognition Letters*, 27(2):102–108, 2006.

[88] Y. Seo, A. Heyden, and R. Cipolla. A linear iterative method for auto-calibration using the dac equation. In *Proceedings of the twentieth IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR'01*, pages 880–885. IEEE, 2001.

[89] Y. Seo and K. S. Hong. Auto-calibration of a rotating and zooming camera. In *Proceedings Of Iapr Workshop On Machine Vision Applications*, pages 17–19, 1998.

[90] Y. Seo and K.S. Hong. About the self-calibration of a rotating and zooming camera: Theory and practice. In *Proceeding IEEE International Conference on Computer Vision, ICCV'99*, pages 183–188, 1999.

[91] P. Sturm. Self-calibration of a moving zoom-lens camera by pre-calibration. *Image and Vision Computing*, 15(8):583–589, 1997.

[92] P. Sturm and L. Quan. Affine stereo calibration. In *Proceedings of International Conference on Computer Analysis of Images and Patterns*, pages 838–843, 1995.

[93] P. Sturm and B. Triggs. A factorization based algorithm for multi-image projective structure and motion. In B. Buxton and R. Cipolla, editors, *Computer Vision, ECCV '96*, volume 1065 of *Lecture Notes in Computer Science*, pages 709–720. Springer Berlin Heidelberg, 1996.

[94] R. Szeliski. *Computer vision: algorithms and applications.* Springer, 2010.

[95] S. Tebaldini, M. Marcon, A Sarti, and S Tubaro. Uncalibrated view synthesis from relative affine structure based on planes parallelism. *In Proceedings of the fifteenth IEEE International Conference on Image Processing, ICIP'08*, pages 317–320, 2008.

[96] P. Torr and A. Zisserman. Mlesac: A new robust estimator with application to estimating image geometry. *Computer Vision and Image Understanding*, 78(1):138–156, 2000.

[97] B. Triggs. Autocalibration and the absolute quadric. In *Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition*, CVPR '97, pages 609–614. IEEE Computer Society, 1997.

[98] R.Y. Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *Robotics and Automation, IEEE Journal of*, 3(4):323–344, 1987.

[99] A. Valdés, J. I. Ronda, and G. Gallego. The absolute line quadric and camera autocalibration. *International Journal of Computer Vision*, 66(3):283–303, 2006.

[100] E. Vincent and R. Laganiere. Detecting planar homographies in an image pair. In *Proceedings of the Second International Symposium on Image and Signal Processing and Analysis, ISPA'01.*, pages 182–187, 2001.

[101] Y. Wan, Z. Miao, and Z. Tang. Robust and accurate fundamental matrix estimation with propagated matches. *Optical Engineering*, 49(10):107002–107009, 2010.

[102] R. G. Willson. *Modeling and Calibration of Automated Zoom Lenses.* Phd thesis, Carnegie Mellon University, 1994.

[103] Yiliang Xu, Sangmin Oh, and A Hoogs. A minimum error vanishing point detection approach for uncalibrated monocular images of man-made environments. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR'13*, pages 1376–1383, 2013.

[104] A. Zaheer, I. Akhter, M.H. Baig, S. Marzban, and S. Khan. Multiview structure from motion in trajectory space. In *Proceedings of the IEEE International Conference on Computer Vision, ICCV'11*, pages 2447–2453, 2011.

[105] Z. Zhang. Flexible camera calibration by viewing a plane from unknown orientations. In *Proceedings of the Seventh IEEE International Conference on Computer Vision, ICCV'99*, pages 666–673, 1999.

[106] Z. Zhang. A flexible new technique for camera calibration. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, 2000.

VITA AUCTORIS

Tarik A. Elamsy was born in 1974 in Kuwait City, the capital city of Kuwait. He received his Bachelor degree in Computer Science from Ajman University of Science and Technology, U.A.E, in 1996 and a Masters of Science degree in Computer Science from the University of Windsor, Canada, in 2009. He received his PhD degree in Computer Science from the University of Windsor, Canada, in 2014. His research interest is Computer Vision and Multimedia Networking.