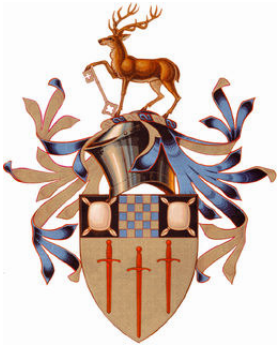
Contributions to image-based object reconstruction: geometric and photometric aspects

Jean-Yves Guillemaut

Submitted for the degree of
Doctor of Philosophy
from the
University of Surrey



Centre for Vision, Speech and Signal Processing
School of Electronics and Physical Sciences
University of Surrey
Guildford, Surrey, GU2 7XH, U.K.

September 2005

Abstract

This thesis treats one fundamental problem in computer vision which is image-based object reconstruction. It concentrates on the problem of improving the geometric accuracy of the reconstructed three-dimensional (3D) models. We define two principal lines of research which are: i) improving camera calibration accuracy, and ii) improving reconstruction accuracy based on Helmholtz Stereopsis (HS). Starting by improving the accuracy of camera calibration is a natural idea, because it is a preliminary stage to most reconstruction techniques. HS is a relatively recent reconstruction technique (2002), based on the principle of Helmholtz reciprocity, and which is remarkable for its ability to reconstruct a wide range of surfaces, regardless of their surface properties.

In camera calibration, we present a collection of methods based on invariants, which can be used to improve calibration accuracy of the camera. Two main classes of methods are presented. The first one is based on Points at Infinity (PI), and applies to a translating camera. The second one is based on a novel entity called the Normalised Image of the Absolute Conic (NIAC). The NIAC generalises the invariance properties of the Image of the Absolute Conic (IAC), and we demonstrate its application for zooming camera calibration. In both situations, experiments with synthetic and real data showed some improvement over standard camera calibration methods which do not consider such invariance properties.

In object reconstruction using HS, we present two main contributions. Firstly, we improve the intrinsic accuracy of the standard HS technique, by formulating an optimum normal reconstruction method, which gives a Maximum Likelihood (ML) estimate under standard Gaussian noise assumption. Secondly, we look at HS in a broader perspective, and observe that the standard pixel based implementation is biased in the case of rough and/or strongly textured surfaces. We propose a novel formulation, supported by recent research in the field of Physics, which does not suffer from such limitations. Results are given with a variety of objects presenting diverse surface properties and whose reconstruction with conventional reconstruction techniques is challenging. We show that HS is able to produce realistic and visually accurate 3D models.

Keywords: Computer vision, camera calibration, Vanishing Points, Image of the Absolute Conic, Normalised Image of the Absolute Conic, image-based object reconstruction, Helmholtz Stereopsis.

Acknowledgements

My first thanks go to my supervisor Prof. John Illingworth for guiding me during the four years of preparation of my PhD thesis. He taught me how to carry out rigorous scientific research. His expertise in the field of computer vision contributed greatly to shaping this thesis, and to the dissemination of the ideas into the scientific community.

In addition to his supervision, I had the chance to benefit from the insights of two other computer vision experts. The first one is Dr. Alberto Aguado, who supervised me during the first eight months of my PhD thesis. I am deeply grateful for his thorough theoretical as well as technical guidance, and also for his enthusiasm, which allowed me to progress rapidly in the field of computer vision. He has been a great source of inspiration for the development of the camera calibration work presented in the first part of this thesis.

The second computer vision expert to whom I would like to express my gratitude is Dr. Ondřej Drbohlav. He supervised me during the six months of my stay as a visiting researcher at the Center for Machine Perception (CMP) in Prague. After having concentrated on the geometric aspect of computer vision for the first years of my PhD, he guided me in the exploration of the photometric aspect of computer vision. He taught me a lot on this topic, always with the voice of a friend, and he contributed enormously to the work on object reconstruction using Helmholtz Stereopsis, which is presented in the second part of this thesis.

I am very grateful to Prof. Josef Kittler for giving me the opportunity to do a PhD at the Centre for Vision Speech and Signal Processing (CVSSP), and for employing me as a research fellow while I was completing the writing-up of my thesis. I am also very thankful to Prof. Václav Hlaváč for receiving me as a visiting researcher at the CMP in Prague.

I would like to thank Dr. Emanuele Trucco and Prof. Adrian Hilton for examining this thesis, and for their comments which improved greatly the quality of this document.

I would like to acknowledge all my colleagues at CVSSP and CMP, and also all my friends, for their help and for making the PhD an enjoyable experience. Some special thanks go to Ondřej Chum for his help and willingness to always make me feel at home in Czech Republic.

Finally I would like to thank my parents Jean-Michel and Odile, my brother Julien, and my girlfriend Afrodita for their constant help, support and encouragement.

This work was supported by EPSRC (project grant number GR/R08629/01), and by the Socrates exchange programme during the six months of my stay in Prague.

Contents

Notations

We adopt the following main mathematical typesetting conventions.

Scalar values are represented in italic, for example a or λ.

Vectors are represented in boldface italic, for example $\boldsymbol{v}$. All vectors are assumed to be column vectors by default. When a row vector is considered, this is indicated explicitely by using the transpose symbol $^\top$. For example $\boldsymbol{p}$ denotes a column vector, while $\boldsymbol{p}^\top$ denotes a row vector.

By abuse of notation, $\mathsf{A}^{-\top}$ denotes $(\mathsf{A}^{-1})^\top$ or $(\mathsf{A}^\top)^{-1}$, where A is an invertible matrix.

Matrices are represented in a sans serif font, for example M or T. Block notations are used when appropriate. For example $[\mathsf{R}|\boldsymbol{t}]$ denotes the matrix which is the result of the concatenation of the matrix R and the vector $\boldsymbol{t}$ (obviously they must have the same number of rows). When a block consists only of zeros, it is usually omitted for clarity, for example $\begin{bmatrix} f & & \\ & f & \\ & & 1 \end{bmatrix}$ stands for $\begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix}$.

In projective geometry, entities are usually defined up to an arbitrary non-zero scale factor. The $\sim$ notation is used to represent equality up to the arbitrary non-zero scale factor.

Unless specified otherwise, $\|\boldsymbol{v}\|$ denotes the L_2 norm of the vector $\boldsymbol{v}$, which is defined as the square root of the sum of its squared components.

The bar symbol over a variable, such as $\bar{L}$, is sometimes used to represent the mean value of the variable.

Acronyms

2D	two-dimensional
3D	three-dimensional
BRDF	Bidirectional Reflectance Distribution Function
DIAC	Dual Image of the Absolute Conic
DLT	Direct Linear Transform
HS	Helmholtz Stereopsis
IAC	Image of the Absolute Conic
LM	Levenberg-Marquardt
ML	Maximum Likelihood
NIAC	Normalised Image of the Absolute Conic
PDF	Probability Density Function
PI	Point at Infinity
RAC	Radial Alignment Constraint
RANSAC	Random Sample Consensus
RMS	Root Mean Squared
SSD	Sum of Squared Differences
SVD	Singular Value Decomposition
VP	Vanishing Point

Chapter 1

Introduction

Image-based object reconstruction consists in inferring three-dimensional (3D) information from two-dimensional (2D) images. If we consider the human vision system, this task is performed almost effortlessly. For example, our two eyes allow us to locate objects in the 3D space and interact with them with an amazing simplicity; looking at an object for a brief period of time allows us to appreciate its speed in addition to its trajectory; also, we can guess the shape of an object from the way it reflects light, and we can anticipate the surface properties of an object and even its shape from visual observation of the texture, even before touching it. We have become so accustomed to reasoning in 3D that we tend to forget that our eyes provide us only 2D information about our environment.

In computer vision, the sensor used to infer 3D information is the camera. Like the human eye, the camera provides 2D information about the scene but in this case in the form of images, which are collections of finite elements called pixels. The challenge of image-based object reconstruction consists of recovering the 3D geometry of the scene from a set of such images. Although the task appears almost trivial in the case of the human vision system, the translation into automatic and accurate computer vision algorithms is still an area of active research.

The ability to build 3D models from images is of broad interest and finds applications in many aspects of science as well as everyday life. Fields of application include for example 3D measurements in manufacturing industry, where traditional metrology applications usually have a high cost which can be reduced by using automatic artificial vision inspection techniques. The

entertainment industry is also a sector where computer vision is widely applied, in particular in the production of special effects in films, and in the generation of virtual worlds for video games. In robotics, 3D vision is of primary interest for the development of autonomous systems such as planetary land rovers used for the exploration of distant planets, or more generally for the development of robots aimed at exploring hostile environments that cannot be accessed directly by humans. In other applications such as augmented reality, computer vision is used to supplement human vision. An example of this type of application is in medicine, where 3D models of organs or tissues can be overlaid onto live images of a camera in order to assist the surgeon during an operation.

Whether they are intended to replace or supplement the human vision system, accuracy is usually important. There are several ways of improving the accuracy of the reconstructed 3D models. One strategy is to improve the knowledge of the sensor used to inspect the environment - this is done through camera calibration. The other strategy consists in improving the reconstruction method used. In most situations, camera calibration is a preliminary stage to reconstruction.

Let us illustrate the problem with a simple example which is not related to computer vision. Suppose that we are given a ruler and would like to measure a 3D object as accurately as possible. The first thing we would like to ensure before carrying out any measurement is that the ruler is accurate. We may want to make sure that the graduations are reliable, and even try to generate more graduations on the ruler if this is possible. This is what we call calibrating the sensor. Once this is done, we can concentrate on the measurement itself. At this stage, the object may be very irregular, we may not be able to access it from all possible angles because of some spatial constraints, or we may simply have time constraints that prevent us from taking as many measurements as we would like. For all these reasons, we may have to make some strategic choices on the parts we are going to measure, and we may have to extrapolate the dimensions of occluded areas by applying some arithmetic to the visible parts or by making some assumptions on the surface. Some methodologies for reconstructing the shape of the object may be more accurate than others. They are usually independent of the calibration accuracy of the ruler.

Let us now come back to computer vision. Calibrating a sensor means building a model of

the way it perceives its environment. In the case of a camera, we can think of each pixel as a directional sensor: each pixel represents a line of sight on which must lie the 3D point viewed by the camera. This is a geometric description of the camera. In addition, each pixel can take a range of intensity values depending on the light reflected by the object surfaces that it captures, the light received by a particular pixel depending on many factors such as the scene lighting, the colours of the objects in the scene, the object surface properties and also their shape and relative spacial arrangement. This is a photometric (or radiometric) description of the camera.

Object reconstruction involves transforming the low level 2D cues contained in the images into high-level 3D models. Naturally both the geometric and the photometric properties of the camera become useful at this stage. For example, the pixel position provides information about the position in space of the imaged surface point and the pixel intensity provides information about the local surface orientation at this point. There exists a multitude of reconstruction methods each of them capitalising on a particular cue. In this thesis, the reconstruction technique chosen is called Helmholtz Stereopsis (HS). It is based on a physical principle called *Helmholtz reciprocity* and has been chosen for its wide range of application. Contrary to most reconstruction methods, HS does not rely on any assumption regarding the surface properties.

1.1 Objectives

The main objective of this thesis is to investigate methods of improving the geometric accuracy of the reconstructed 3D models. We distinguish two main problems: camera calibration and object reconstruction. Both problems contribute to the general accuracy of the 3D model reconstructed. For example, the best reconstruction method would perform very poorly if the camera is inaccurately calibrated, and equally, calibrating accurately a camera is of no use if it is not followed by an accurate reconstruction method. Both problems can usually be treated sequentially; first the camera is calibrated and then the object is reconstructed. It should be mentioned here that there exists more sophisticated technique where the two task cannot be so clearly separated (see Section 2.3.6).

Improving the geometric accuracy of 3D reconstruction is too general a problem, which goes beyond the scope of a single PhD thesis. For this reason we have identified some more specific

objectives. In the case of camera calibration, we concentrate on geometric calibration, therefore leaving the radiometric calibration problem as a separate issue. In the case of reconstruction, we focus on improving the accuracy of the reconstruction based on HS. Our objectives can therefore be restated as follows in the light of the two main problems defined:

1. Improve the accuracy of geometric camera calibration,
2. Improve the accuracy of the reconstruction based on HS.

The essential issues addressed in this thesis are:

1. In camera calibration, the constraints used are provided by the observation of a specific calibration object. The more images we take of the calibration object, the more constraints we have for the calibration of the camera. Multiplying the information available by taking multiple images at different positions or with different lens settings is *a priori* a plausible strategy for increasing the number of constraints and thereby the calibration accuracy. However, everytime the camera moves or changes its settings, this also introduces new parameters to calibrate. Can we significantly increase the number of views, and thereby the number of calibration constraints available, without increasing arbitrarily the dimensionality of the problem and affecting the calibration accuracy?
2. For object reconstruction, HS has been shown to be a powerful method for a large variety of objects. Can we improve further the intrinsic accuracy of the method for such objects? What are the limitations of the current HS algorithm? In particular, can we extend the applicability of the method to a wider class of surfaces?

1.2 Contributions

Our contributions are at two levels. The first one is in camera calibration, the second one in HS. They are clearly related by sharing the same goal: improving the geometric accuracy of the 3D models reconstructed. But they are also distinct and independent. The reader interested only in camera calibration will benefit from our work on camera calibration and is free to

implement it followed by the reconstruction method of their choice. Similarly, our work on HS is independent of the technique chosen to calibrate the camera.

In the case of camera calibration, our main contributions are the following:

- Investigation of the use of invariants to increase calibration accuracy.
- Proposition of a novel method based on Points at Infinity (PI) for calibrating a translating camera. Contrary to similar methods which make use of the invariance to translation, our method does not require the observation of sets of parallel lines in the scene and is therefore more flexible. The novel method results in an improvement in the calibration accuracy compared to standard calibration methods which do not exploit the invariance property.
- Definition of a novel entity called the Normalised Image of the Absolute Conic (NIAC), which is the extension of the Image of the Absolute Conic (IAC) to zooming invariance. The NIAC is a geometric abstraction which encapsulates all the camera parameters invariant to zooming.
- Proposition of a novel method for calibrating a zooming camera based on the NIAC. The method requires only to take several images of a plane and it has been shown to be more accurate than other plane-based calibration methods.

In the case of image-based object reconstruction using HS, the following contributions have been made:

- Definition of a radiometric distance for optimum normal estimation. The novel distance introduced is a Maximum Likelihood (ML) estimate under standard Gaussian noise assumption. This guarantees an optimum surface normal estimation.
- Observation that the standard HS constraint is biased in the case of rough and strongly textured surfaces.
- Formulation of a novel HS constraint applicable to rough and/or strongly textured surfaces and demonstration of its success on a variety of challenging real objects.

1.3 Structure of the thesis

The thesis is structured as follows. This chapter was a general introduction with the aim of motivating the work presented in this thesis and stating the main objectives and contributions. The rest of the thesis is divided into three parts. Part I is dedicated to camera calibration. It starts with Chapter 2 in which we review the main camera calibration methods. In that chapter, we also introduce some general concepts such as the camera models and some notations which will be useful in the rest of the thesis. In Chapter 3, we propose a novel camera calibration method for a translating camera. In Chapter 4, we continue our exploration of invariants and define a novel invariant to translation, rotation, and zoom, called the NIAC. We show how this invariant can be used to calibrate a zooming camera. Part II concentrates on the reconstruction of 3D models from images. It starts with a broad review of the topic in Chapter 5. We consider the applicability of the different techniques in terms of types of object surfaces to which they apply. This motivates the choice of HS for reconstruction in this thesis. In Chapter 6, we tackle the normal estimation problem with HS and come up with an optimum solution to the problem. In Chapter 7, we pursue reconstruction of surfaces using HS, but this time extending the method to a wider class of surfaces, which could not be reconstructed efficiently by previous implementations of the method. Part III, which consists only of Chapter 8, closes the discussion on improving the geometric accuracy of the 3D models reconstructed. It concludes and proposes some avenues for future work. Some additional material and proofs are given in the appendices at the end of the thesis.

1.4 List of publications

The results from this research have been reported in a number of publications.

Conferences:

- J.-Y. Guillemaut, A.S. Aguado, and J. Illingworth. Using Points at Infinity for parameter decoupling in camera calibration. In *Proc. British Machine Vision Conference*, pages 263–272, volume 1, September 2002.

- J.-Y. Guillemaut, A.S. Aguado, and J. Illingworth. Calibration of a zooming camera using the Normalized Image of the Absolute Conic. In *Proc. International Conference on 3-D Digital Imaging and Modeling*, pages 225–232, October 2003.

- J.-Y. Guillemaut, O. Drbohlav, R. Šára, and J. Illingworth. Helmholtz Stereopsis on rough and strongly textured surfaces. In *Proc. International Symposium on 3D Data Processing, Visualization and Transmission*, pages 10–17, September 2004.

Journals:

- J.-Y. Guillemaut, A.S. Aguado, and J. Illingworth. Using Points at Infinity for parameter decoupling in camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 27(2):265–270, February 2005.

Part I

Geometric aspect:

Camera calibration using invariants

Chapter 2

Background

2.1 Introduction

The main sensor used in computer vision is the camera. It provides information about the physical world surrounding us in the form of 2D images. Before being able to extract 3D information from such images, it is important to be able to model the phenomenon taking place in the camera during image formation. The estimation of the parameters of the model of the image formation process is the aim of camera calibration. This is of major importance in computer vision, as it is a preliminary stage to most vision based object reconstruction techniques. As such, camera calibration has been a topic of interest in computer vision and photogrammetry for nearly half a century, and there exists a very extensive literature on the topic. It is obviously not possible to mention all the methods here, however this survey is intended to give a good overview of the diversity of the existing approaches, including the most commonly used methods.

The chapter is structured as follows. First, the different camera models are described in Section 2.2. Then a taxonomy of the main approaches for the estimation of the parameters of the camera model chosen is proposed in Section 2.3. In this chapter, we also introduce the notations which will be employed in the rest of the thesis.

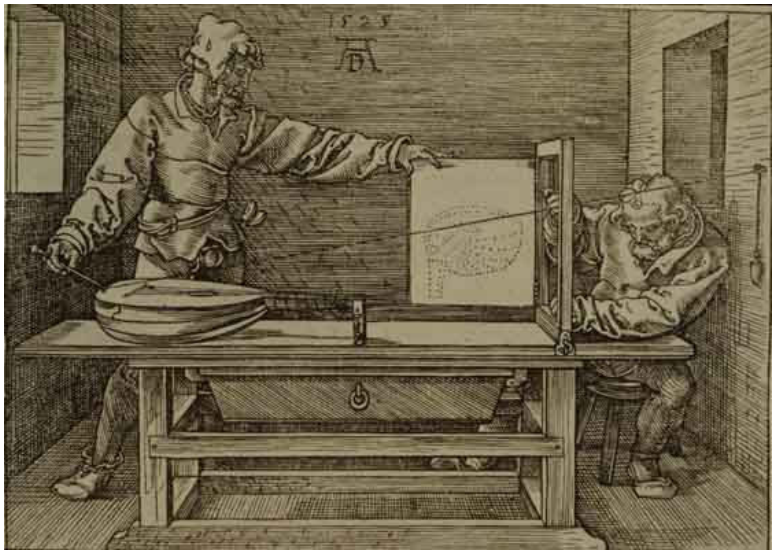


Figure 2.1: Man Drawing a Lute, *Albrecht Dürer, 1525. The artist illustrates here an example of device which can be used to draw perspective images of objects.*

2.2 Geometric camera model

The camera model characterises the mapping (perspective projection) from 3D world points to 2D image points taking place in the camera during image formation. Historically, the first perspective pictures appeared early in the fifteenth century with Renaissance painters who introduced the primary concepts of perspective and projective geometry (see Fig. 2.1 for an illustration of the principle). In particular, they designed tools such as the *camera obscura* (or dark room), which is the ancestor of todays camera, in order to generate realistic rendering of scenes. Some of the material contained in this section is now fairly standard, for this reason references are sometimes omitted. The reader interested is referred to standard textbooks on the topic [46, 154, 72].

2.2.1 Basic pinhole model

The *pinhole* camera model is the most commonly used geometric camera model in computer vision. It is illustrated in Fig. 2.2. It consists of an image plane $\boldsymbol{\pi}$ and a point $\boldsymbol{C}$ called the optical centre or the camera centre. The plane $\boldsymbol{F}$ passing through the optical centre and parallel to the image plane is called focal plane. Focal plane and image plane are separated by

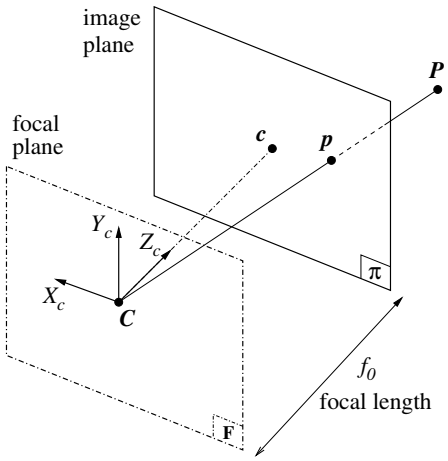


Figure 2.2: *The pinhole model. Note that the image plane is placed in front of the focal plane even though physically it is located behind; this convention is equivalent, it is preferred because it allows to work with non-inverted images.*

a distance f_0 that is called the effective focal length. The line passing through the optical centre and perpendicular to the image plane is called the optical axis; it intersects the image plane in a point called principal point $\boldsymbol{c}$.

A ray of light emitted by a scene point $\boldsymbol{P}$ travels through the optical centre and intersects the image plane in an image point $\boldsymbol{p}$. In the *camera reference frame*, centred at $\boldsymbol{C}$ with the Z axis pointing along the optical axis (see Fig. 2.3), the relation linking a 3D point $\boldsymbol{P}_c = [X_c, Y_c, Z_c, 1]^\top$ and its projection $\boldsymbol{p}_c = [x_c, y_c, w_c]^\top$ in the image plane[1] is expressed in homogeneous coordinates by

$$\boldsymbol{p}_c \sim \begin{bmatrix} f_0 & & & 0 \\ & f_0 & & 0 \\ & & 1 & 0 \end{bmatrix} \boldsymbol{P}_c \,, \tag{2.1}$$

where the symbol $\sim$ denotes the equality up to a non-zero scale factor.

2.2.2 Extrinsic and intrinsic parameters

In the previous section, the camera reference frame was introduced because it was mathematically the most appropriate for expressing the perspective projection in a simple form (see

[1]Note that by abuse of notation we have dropped the Z component in the expression of $\boldsymbol{p}_c$. Image points being located in the image plane, the Z component is always equal to f_0.

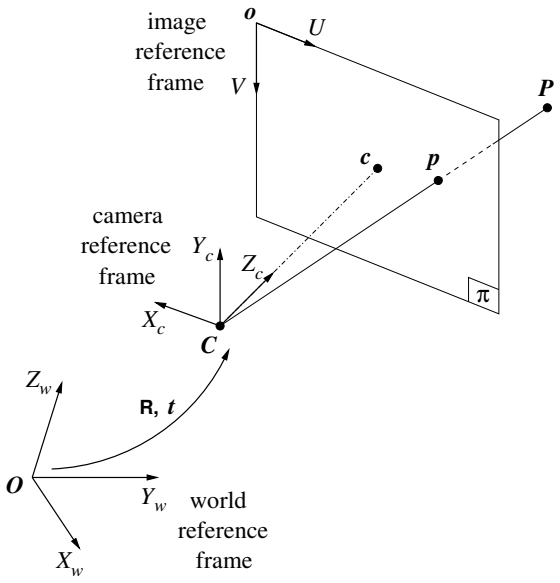


Figure 2.3: *The different reference frames used to model the image formation process. The intrinsic parameters represent the transformation from the image to the camera reference frame, while the extrinsic parameters represent the transformation from the camera to the world reference frame.*

Eq. (2.1)). In practice, however, the camera reference frame is not directly accessible to the operator, because it is not attached to any visible reference object (the optical centre for example is located somewhere inside the camera). For this reason, the *image reference frame* and the *world reference frame* are defined; they are linked respectively to the image and some easily recognisable features from the environment (or world). Two sets of parameters called *intrinsic* and *extrinsic* parameters are introduced; they characterise the transformations between reference frames.

The intrinsic parameters

These parameters define the projective transformation between a 3D point $\boldsymbol{P_c}$ expressed in the camera reference frame and its image $\boldsymbol{p} = [u, v, w]^\top$ expressed in pixel coordinates. It has already been seen in the previous section that the focal length models the central projection within the camera reference frame. The other intrinsic parameters represent the 2D transfor-

mation required to convert camera coordinates into image coordinates:

$$\boldsymbol{p} \sim \begin{bmatrix} m_u & -m_u \cot\theta & u_0 \\ & m_v/\sin\theta & v_0 \\ & & 1 \end{bmatrix} \boldsymbol{p_c}\,. \tag{2.2}$$

(u_0, v_0) are the coordinates in pixels of the principal point, they represent the offset between the origins of the two frames. m_u and m_v are the number of pixels per unit distance. Finally, for generality, θ represents the angle between the axes u and v of the image reference frame. For most normal cameras $\theta = \pi/2$ rad, however in some rare instances this parameter can take different values. In practice, it can also be convenient, for linearity of the equations, to compute a general model with all the parameters. Combining Eq. (2.1) and (2.2), we obtain

$$\boldsymbol{p} \sim [K \mid \mathbf{0}]\, \boldsymbol{P_c}\,, \tag{2.3}$$

where K is called the *calibration matrix* and is defined by

$$K = \begin{bmatrix} f_0 m_u & -f_0 m_u \cot\theta & u_0 \\ & f_0 m_v/\sin\theta & v_0 \\ & & 1 \end{bmatrix}. \tag{2.4}$$

The parameters f_0, m_u and m_v are redundant, they can be grouped into two new parameters, the focal length $f = f_0 m_u$ in pixel units along the u axis, and the aspect ratio $r = m_v/m_u$. The aspect ratio can be different from 1 in the case of CCD cameras, and it is usually necessary to estimate it for an accurate calibration. Note also that the term $s = -f\cot\theta$ is called the *skew parameter*. In summary, K is parametrised by five intrinsic parameters:

$$K = \begin{bmatrix} f & -f\cot\theta & u_0 \\ & fr/\sin\theta & v_0 \\ & & 1 \end{bmatrix}. \tag{2.5}$$

The extrinsic parameters

They define the transformation from the camera reference frame into the world reference frame. This transformation models the camera orientation (rotation matrix R) and location (translation vector $\boldsymbol{t}$) with respect to the world reference frame. A world point $\boldsymbol{P} = [X, Y, Z, 1]^\top$ and its

coordinates $\boldsymbol{P}_c$ in the camera reference frame are related by

$$\boldsymbol{P_c} \sim \begin{bmatrix} R & \boldsymbol{t} \\ & 1 \end{bmatrix} \boldsymbol{P} \,. \tag{2.6}$$

There are six extrinsic parameters: three for the rotation and three for the translation.

Eq. (2.3) and (2.6) can be grouped into a single relation linking 3D points $\boldsymbol{P}$ in world coordinates and their projection $\boldsymbol{p}$ in pixel coordinates:

$$\boldsymbol{p} \sim M\boldsymbol{P} \,, \quad \text{with} \quad M = K\,[R \mid \boldsymbol{t}] \,, \tag{2.7}$$

where M is called the *projection matrix*. The projection matrix has eleven degrees of freedom and is fully characterised by intrinsic and extrinsic parameters. The block form $[R \mid \boldsymbol{t}]$ represents the 3 by 4 matrix obtained from the concatenation of R and $\boldsymbol{t}$. This concise notation is commonly used in the rest of this thesis.

2.2.3 Zooming camera model

So far, only camera models with static parameters have been considered. Zooming camera models, however, must incorporate variable parameters in order to accommodate variations in the lens' zoom. This is typically a complex problem, because of the variations in the optical alignment of the lens' components, and the displacement of these elements along the optical axis which occur during zooming. The choice of the zoom model is usually dictated by the accuracy required, and also by the individual specifications of each camera.

The primary effect of zooming is to change the focal length of the camera. To model this, it is convenient to separate the focal length from the other intrinsic parameters, by defining the following matrices:

$$K_1 = \begin{bmatrix} 1 & -\cot\theta & u_0 \\ & r/\sin\theta & v_0 \\ & & 1 \end{bmatrix} \quad \text{and} \quad F = \begin{bmatrix} f & & \\ & f & \\ & & 1 \end{bmatrix} .$$

An ideal zooming camera model is obtained, with F encapsulating the zooming properties and K_1 containing the other intrinsic parameters:

$$M = K_1 F\,[R \mid \boldsymbol{t}] \,. \tag{2.8}$$

It has been observed in [165, 166] that zooming can also affect the field of view of the camera, which can be approximated by considering a variable principal point. A more accurate model [166] considers a variable position of the optical centre along the optical axis (Z axis of the camera reference frame), in addition to the three other variable parameters. In [163], a general methodology for building models of cameras with variable parameters is presented and applied to the case of variable zoom and focus lenses. The main idea is to describe each camera parameter by a polynomial function of the lens control settings. More details can be found in [164].

2.2.4 Lens distortion model

The pinhole model is generally a good approximation of the image formation process taking place in most cameras. However, in reality, a number of deviations called *aberrations* are observed. There are many types of deviations (see [127] for a detailed description). Typically the *radial distortion* is the most significant one. It consists of a displacement of the image points radially towards or away from the centre of radial distortion. Usually it is sufficient to claim that the centre of radial distortion and the principal point are the same, but this is not necessarily the case (see [166]). This effect is generally relatively well modelled if two distortion coefficients k_1 and k_2 are introduced to warp distorted image points $\boldsymbol{p}_d = [u_d, v_d, 1]^\top$ to undistorted ones $\boldsymbol{p} = [u, v, 1]^\top$ by the relation

$$\begin{cases} u = u_0 + (u_d - u_0)(1 + \kappa_1 d^2 + \kappa_2 d^4) \\ v = v_0 + (v_d - v_0)(1 + \kappa_1 d^2 + \kappa_2 d^4) \end{cases} \quad \text{with} \quad d^2 = [r(u_d - u_0)]^2 + [v_d - v_0]^2 . \quad (2.9)$$

For greater accuracy, it is necessary to introduce a centre of radial distortion independent of the principal point.

2.2.5 Other models

Approximations of the general pinhole model

So far, it has been assumed that the optical centre of the camera is a finite point. It is possible to define other models called *affine models* by placing the optical centre in the plane at infinity

[72]. In addition, it is possible to construct approximations of the general pinhole model such as the *paraperspective* or *orthoperspective* models. A hierarchy of such camera models is presented in [6]. Such models are less accurate, however they present a reduced number of parameters, which can reduce considerably the complexity of many applications.

Thin lens model

With the pinhole model, the objects observed were always in focus, because only one ray coming from each visible point could enter the camera. But the aperture of a real camera is not a point and it is therefore necessary to use an optical system made of lenses and other elements to guarantee that the rays emerging from the same 3D point converge to the same image point. The behaviour of these systems is relatively complex, however, it can be modelled relatively accurately by the thin lens model [19]. With such a model only points located in a plane parallel to the image plane can be in focus. An example of application is in shape from defocus, where the amount of blur is used to infer the object geometry [27]. This model is much more complex than the pinhole model, in particular it does not present the linear properties of the latter model, and is therefore rarely used in computer vision applications.

2.3 Camera calibration

The problem of camera calibration consists in estimating the parameters of the model chosen for the camera. Many of the techniques used in computer vision are inspired from the photogrammetry literature. Typically, being able to calibrate a camera accurately is critical because it affects directly the accuracy of the reconstruction made from images. The task is carried out by deriving some relations between the 3D world and the images taken by the cameras. Each relation constrains the camera parameters. When present in a sufficient number, these constraints form a system, whose solution gives the values of each parameter. The complexity of the equations depends on the nature of the relations that are established, therefore it is of critical importance to consider adequate entities for the definitions of these relations. Typically, the entities considered for calibration are objects with known characteristics, for example 3D points with known coordinates (see Fig. 2.4). But it is possible to use more sophisticated

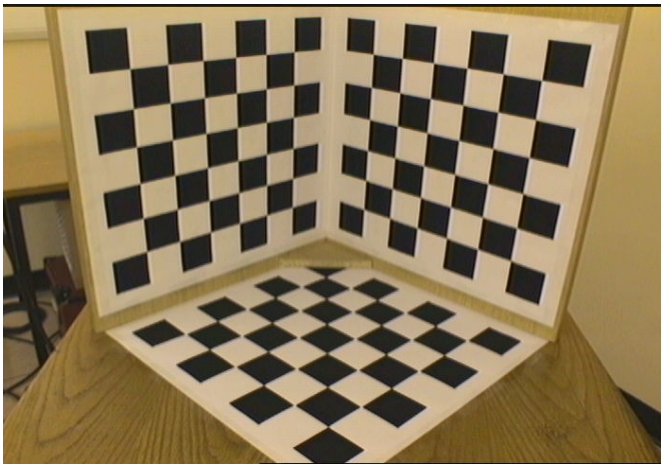


Figure 2.4: *A typical camera calibration pattern made of two orthogonal planes containing points with known 3D coordinates (control points).*

entities which, for example, can be more complex geometric shapes, or even imaginary objects such as the Image of the Absolute Conic (IAC) (see Section 2.3.3). Other approaches called *auto-calibration* (or *self-calibration*) remove completely the requirement of having any *a priori* knowledge about the scene observed. In this section, a taxonomy of the different camera calibration methods is proposed. The methods are classified according to the properties of the entities used to form the calibration constraints.

2.3.1 Calibration from point correspondences

The simplest correspondence which can be established is through 3D points with known coordinates. These points are called control points and are defined on a calibration pattern, usually made of two or three mutually orthogonal planes (see Fig. 2.4) and engineered with very high accuracy. The key idea is to find values of the intrinsic and extrinsic parameters which will best map the control points to their corresponding image points.

Linear methods

Linear methods have been used extensively for solving the calibration problem (see [63, 51] to cite only a few). This approach is called *Direct Linear Transform (DLT)* [1]. A good description of this class of methods is given in [72]. In the case of an ideal pinhole camera

(no lens distortion), each correspondence between a 3D point $\boldsymbol{P} = [X, Y, Z, 1]^\top$ and its image $\boldsymbol{p} = [u, v, w]^\top$ is constrained by Eq. (2.7). Denoting by $\boldsymbol{m}$ the vector formed by concatenating all the row vectors of the camera calibration matrix M:

$$\boldsymbol{m} = \begin{pmatrix} \boldsymbol{m}_1 \\ \boldsymbol{m}_2 \\ \boldsymbol{m}_3 \end{pmatrix} \qquad \text{where} \quad M = \begin{bmatrix} \boldsymbol{m}_1^\top \\ \boldsymbol{m}_2^\top \\ \boldsymbol{m}_3^\top \end{bmatrix}, \tag{2.10}$$

the following constraint can be derived from Eq. (2.7):

$$\begin{bmatrix} \boldsymbol{0}^\top & -w\boldsymbol{P}^\top & v\boldsymbol{P}^\top \\ w\boldsymbol{P}^\top & \boldsymbol{0}^\top & -u\boldsymbol{P}^\top \\ -v\boldsymbol{P}^\top & u\boldsymbol{P}^\top & \boldsymbol{0}^\top \end{bmatrix} \boldsymbol{m} = \boldsymbol{0}\,. \tag{2.11}$$

It appears that the three equations defined above are linearly dependent, that is one point correspondence leads to only two constraints on the elements of M (the first two equations for example). The general system having 11 unknowns (5 intrinsic parameters and 6 extrinsic parameters), it can be solved with a minimum of six world points in a general position (see [25] for a characterisation of all the degenerate configurations). In practice, it would be very inaccurate to consider only the minimum number of points because of the noise in the extraction of the image points, therefore a larger number of points is used and a least-squares solution can be computed. Stacking up the previous equations, a $3n \times 12$ matrix A such that $A\boldsymbol{m} = \boldsymbol{0}$ can be defined (note that A has dimension $2n \times 12$ if only two linearly independent equations are considered for each correspondence). In [63], the pseudo inverse is used to find the solution that minimises $||A\boldsymbol{m}||$ subject to the constraint that the last element of $\boldsymbol{m}$ is equal to 1, while in [72], Singular Value Decomposition (SVD) [112] is used to find the solution that minimises $||A\boldsymbol{m}||$ subject to the constraint that $||\boldsymbol{m}|| = 1$. Other constraints such as $||\boldsymbol{m}'_3|| = 1$, where $\boldsymbol{m}'_3$ is the 3-vector formed by the first three components of $\boldsymbol{m}_3$, have also been considered in [51] for their invariance to rigid camera motion. The quantity $A\boldsymbol{m}$ minimised by these techniques is called *algebraic error* [70].

It is important to note that these methods estimate the coefficient of the camera calibration matrix, but do not provide directly the values for the camera parameters. There are various ways to estimate these parameters. For example, [58, 51] give some analytic formulas for the computation of these parameters. A simpler way is to apply the RQ decomposition [112],

in order to decompose M in the form given in Eq. (2.7), from which each parameter can be identified subsequently as in [72]. It has been shown in [67] that an algebraic distance is very sensitive to the choice of the reference frames. In particular, this can lead to bad conditioning of the system, and thereby poor accuracy in the evaluation of the parameters. The solution is to apply an appropriate normalisation which guarantees that the system is well conditioned and that optimum results are obtained. There is a vast literature on methods to compute optimally the solution of such a system of equations [80, 67, 70, 98, 85, 100, 99, 30, 77, 31]. In general, the closed-form solution is attractive because it is very fast to compute. One major drawback is that it is limited to linear models, and therefore does not allow to solve for the distortion parameters. In addition, minimising a geometric distance rather than an algebraic distance usually leads to more accurate results.

Non-linear methods

Non-linear methods perform a direct search in the parameter space in order to find the parameters which minimise an appropriate cost function. This is a classical method from photogrammetry called *bundle adjustment* [127, 152]. Typically the cost function is of the form

$$\sum_i d(\boldsymbol{p}_i, K\,[R \mid \boldsymbol{t}]\,\boldsymbol{P}_i)^2\,, \tag{2.12}$$

where d is a geometric distance or error function. The method requires the use of a non-linear optimisation algorithm such as the Levenberg-Marquardt (LM) algorithm [112]. For example, the Gold Standard camera calibration algorithm described in [72] uses the DLT algorithm to compute an initial estimate for all the linear parameters, which are then refined by bundle adjustment. Generally the minimisation can be extended to several image frames, in order to have a larger number of correspondences and thereby improve the accuracy of the estimation of the parameters. One nice property of this method is its generality; it is able to accommodate arbitrary camera models, including complex lens distortion models, by simply including these parameters into the distance function d minimised. Non-linear methods can be more accurate than the linear ones. However because these methods require use of an iterative optimisation algorithm, convergence to the right solution is not always guaranteed, especially if there are a large number of parameters to optimise. In particular, there is a risk, if the method is initialised

badly, that the algorithm will converge to a local minimum which is different from the correct solution. These methods are also much more computationally expensive than linear methods.

Two-step methods

A good compromise consists in combining the two previous approaches: a subset of the parameters are computed using a linear method, then the remaining parameters can be estimated using a non-linear optimisation technique. The convergence of the latter is not guaranteed if the initial guess provided by the linear method is far from the optimum solution, in addition it is usually slow. An algorithm with faster convergence properties is proposed in [155]. The key idea in this paper is to use the Radial Alignment Constraint (RAC) in order to decompose the camera parameters into two groups. The first group of parameters contains the extrinsic parameters (except the position along the Z axis) and the scale factor and can be computed linearly. The second group contains the effective focal length f, the radial lens distortion coefficients and the position along the Z axis; the computation of these parameters require the use of non-linear optimisation techniques, however the convergence is usually extremely fast (one or two iterations according to [155]) because of the small number of variables. The method presented in [155] is able to accommodate radial lens distortion, however it assumes that some of the intrinsic parameters are provided by the manufacturer. This assumption is somehow relaxed in [86] where two additional intrinsic parameters are pre-computed (principal point and scale factor).

The approach in [155, 86] is limited to radial lens distortion models because with other types of distortions it is usually not possible to apply the RAC. A more general method was proposed in [161]. The method separates the set of camera parameters into two sets; the first set contains the external and internal non-distortion parameters, while the second set contains the distortion parameters only. The procedure involves optimising alternatively the first set of parameters (linear algorithm) while the second set is fixed, and then the second set of parameters (non-linear algorithm) while the first set is fixed. The procedure is repeated until convergence. The lens distortion parameters optimised are the radial distortion, the decentring distortion and the thin prism distortion.

Conclusions

Calibration from point correspondences is probably the best established approach for camera calibration. A comparative review of some of the most commonly used methods is given in [117]. This class of methods usually gives the best accuracy. The main limitation however is the lack of flexibility, in particular the requirement of using a high accuracy calibration pattern which is typically difficult and expensive to produce.

2.3.2 Calibration using Vanishing Points

General concept

In this section, methods using particular points called *Vanishing Points (VPs)*, which are defined by parallel lines, are considered. We start by introducing a few concepts of projective geometry which are required to understand the methods. In projective geometry, any set of parallel lines intersects in a point located infinitely far away called a Point at Infinity (PI). Mathematically, such points are characterised by their last homogeneous coordinate which is equal to zero, *i.e.* a PI can be written in the form $(\boldsymbol{d}^\top, 0)^\top$. The set of all PI form a plane called the plane at infinity $\boldsymbol{\pi}_\infty$, which represents all possible 3D directions. The projection of a PI $\boldsymbol{D} = (\boldsymbol{d}^\top, 0)^\top$ is a point

$$\boldsymbol{v} \sim K\,[R \mid \boldsymbol{t}]\,\boldsymbol{D} \sim KR\boldsymbol{d}\,, \tag{2.13}$$

called a VP. It can be observed from the previous equation that a VP is independent of translation of the camera. Intuitively, one can compare them to the image of stars in the sky or points far away on the horizon, which stay fixed as an observer moves with a translational motion in the scene. In the image plane, such points appear as the intersection of the projection of parallel lines. Analogously, parallel scene planes intersect in a line located in the plane at infinity and whose projection in the image is called a vanishing line. The line of intersection represents all the directions contained in the plane.

The general idea of VP(or vanishing line)-based methods is to use the invariance of VPs to camera translation in order to decompose the calibration into two stages. The intrinsic and rotation parameters are computed in a first stage from the VPs only; the translation parameters are then computed in a second stage from other known scene features (usually segments or

points). VPs are computed directly in the image as the intersection of parallel lines [26, 159, 43, 28, 160, 12, 33, 88]. For robustness, VPs are usually computed from the intersection of more than two lines, by minimising an appropriate criterion (see *e.g.* [33]). One VP provides two constraints on the intrinsic parameters and the rotation in the form of Eq. (2.13) (three equations minus the scale factor). Therefore with three VPs, the rotation (three parameters) and only three of the intrinsic parameters (usually the coordinates of the principal point and the focal length) can be computed in the first stage.

Calibration from images of sets of parallel lines

Caprile and Torre give a simple camera calibration method requiring only a cube for calibration target in [26]. It is assumed there that the camera has no skew and that its aspect ratio is known (for example it has been pre-calibrated). The method is based on the property that under these conditions, the principal point is located at the orthocentre[2] of a triangle with vertices defined by the VPs of the three mutually orthogonal sets of parallel lines defined by the cube. Once the principal point has been estimated, the focal length and then the rotation parameters are computed in a straightforward manner from the equations defined by the VPs. Finally, the translation parameters are obtained from the additional information provided by the correspondences of the projection in two images of a segment of known length and orientation. Degenerate configurations appear if one or more VP are at infinity in the image, *i.e.* if one or more sets of parallel lines are parallel to the image plane.

Similar calibration methods considering images of a parallelepiped have been presented in [159, 43, 28]. In [159], the principal point of a camera with zero skew and known aspect ratio, is computed as the orthocentre of a triangle whose edges are the vanishing lines of the three orthogonal planes defined by the calibration pattern. The authors also give some geometric characterisation of the camera orientation and focal length in terms of respectively the slope of the vanishing lines obtained and the area of the triangle previously constructed. Ultimately, they estimate the camera position from the image of known 3D points. In [43], it is shown that, for a camera with zero skew, known aspect ratio, and principal point (assumed to be at the image centre), the three virtual image lines intersecting at the principal point and each going through

[2]The intersection of the three altitudes of a triangle is called the orthocentre.

one of the three VPs, depend only on the rotation parameters. The method can be related to the previous methods [26, 159] by observing that the virtual lines constructed are actually the altitudes of the triangles previously defined. Point correspondences are used to derive the translation and focal length after the rotation has been computed. The same invariance property is used in [28] to compute the orientation of a camera with known intrinsic parameters, from one image of a planar grid containing two orthogonal sets of parallel lines.

In [160], it is shown that a single vanishing line constructed from three VPs can be used to compute the camera focal length and orientation. The three VPs are obtained from the image of an hexagonal pattern, by intersecting parallel opposite edges. The vanishing line is then fitted to the VPs obtained. The authors relate swing, pan and tilt angles (also the focal length) to some geometric characteristics of the vanishing line and the VPs, such as slope, intercept with image axis or ratio of distances. The translation parameters are obtained from the correspondence of known scene points on the grid. The other intrinsic parameters are not computed with this method.

Calibration from images of a plane

Another method for calibrating the intrinsic parameters of a camera is described in [12]. The method calibrates the camera from several images of a plane taken under different viewing angles. The main advantage compared to the previous VP-based methods is that it is not necessary to ensure the orthogonality of the planes observed. The plane contains a pattern defined by at least four points with known coordinates (no three of them being colinear), so that the planar homography between the plane and each image can be computed. The homography is then used to compute the vanishing line of the plane and some specific VPs on this line. It is shown that the principal point lies on a line perpendicular to the vanishing line and going through a particular VP (this line is actually the *centre line* described in [62, 61], another application in calibration is described in Section 2.3.3). If more than two images are used, the principal point can be estimated as the intersection of all the perpendicular lines constructed. This construction is valid for a known aspect ratio, however if it is not the case, it is still possible to use the same procedure to calibrate iteratively the camera, initialising with a good estimate of the aspect ratio. Other properties are used to estimate the focal length and the aspect ratio once

the other intrinsic parameters have been computed. The method requires two or three images of the calibration pattern, depending whether the aspect ratio must be computed or not (in all cases the skew is assumed to be zero).

Calibration from architectural scenes

Some important applications of VP methods are in architecture, where man-made structures such as buildings usually contain plenty of mutually orthogonal sets of parallel lines [33, 88]. An example of an interactive system for image-based reconstruction of buildings is presented in [33]. The operator is asked to assist in the marking of mutually orthogonal sets of parallel lines in the image, which are then used to calibrate the intrinsic parameters and the orientation of the camera under the assumption of known aspect ratio and zero skew. An additional point correspondence is finally used to obtain a metric calibration (up to a scaling factor). The algebraic solution proposed in [33] is mathematically equivalent to the previous methods. A similar calibration algorithm is presented in [88]. The originality is that the authors reformulate VP-based calibration in terms of some properties of the Image of the Absolute Conic ω (see next section for a formal definition). The main idea is that orthogonality is encoded by conjugacy with respect to the absolute conic Ω_∞. Then, two orthogonal VPs $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ are conjugate with respect to ω, *i.e.* $\boldsymbol{v}_1^\top \omega \boldsymbol{v}_2 = 0$. Similarly, it can be observed that a vanishing line $\boldsymbol{l}$ and a VP $\boldsymbol{v}$, respectively corresponding to a plane and a direction orthogonal to the plane, are pole-polar with respect to ω, *i.e.* $\boldsymbol{l} = \omega \boldsymbol{v}$. In any case, one constraint on the intrinsic parameters, which can be combined with other constraints to solve for calibration, becomes available (three such constraints arising from three orthogonal VPs are sufficient if we assume a camera with zero-skew and known aspect ratio).

Conclusions

In summary, the main advantage of VP-based methods is that they replace the requirement of a calibration pattern with accurately located control points on a pattern made of sets of parallel lines, usually required to be in mutually orthogonal planes. This presents some practical advantages, especially in the case of architectural scenes. One limitation however is that VPs can be difficult to localise accurately. For example, if the angle between the parallel scene lines and

the image plane is small, the point of intersection of the image lines, which defines the VP, is located far away in the image plane and usually cannot be computed accurately.

2.3.3 Calibration using the Image of the Absolute Conic

Even though it has been seen in the previous section that the use of VP can simplify calibration, parallelism and orthogonality remain strong constraints. Methods based on the Image of the Absolute Conic (IAC) propose to relax the orthogonality constraint by allowing arbitrary relative positioning of the planar calibration object observed. They also reformulate elegantly the calibration problem in terms of the estimation of an imaginary geometric object.

The absolute conic Ω_∞ was introduced to the computer vision literature by Faugeras and Maybank in [50]. The conic consists of the set of points $[X, Y, Z, W]^\top$ satisfying the equations

$$\left.\begin{aligned} X^2 + Y^2 + Z^2 &= 0 \\ W &= 0 \end{aligned}\right\} . \tag{2.14}$$

This conic is located in the plane at infinity and is the circle of radius $\mathrm{i} = \sqrt{-1}$, which consists purely of complex points. It is invariant under rigid motions and under uniform changes of scale. The image of the absolute conic Ω_∞ by the camera projection matrix is the conic

$$\omega = K^{-\top} K^{-1} , \tag{2.15}$$

which is also an imaginary object. It can be observed that the IAC is invariant to the position and orientation of the camera [49]. This is a very powerful property because it results immediately that computing the intrinsic parameters (*i.e.* the calibration matrix K) is equivalent to estimating the IAC ω. Once ω is known, K can be obtained from Eq. (2.15) using for example Cholesky factorisation [112]. The IAC provides a very convenient mental representation of the intrinsic parameters, and finds many applications in plane-based calibration [89, 88, 175, 138, 96, 95, 62, 61] and also in auto-calibration (see Section 2.3.6).

Calibration of cameras with constant intrinsic parameters

The general principle is given by Zhang in a seminal paper on plane-based camera calibration [175]. The main idea is to compute some particular points belonging to ω from the observation

of a planar pattern, and then to use appropriate techniques [173] to fit a conic to these points and recover ω. In particular, on each plane there exists two points called *circular points*, with canonical coordinates $\boldsymbol{I} = (1, \mathrm{i}, 0)^\top$ and $\boldsymbol{J} = (1, -\mathrm{i}, 0)^\top$. These two points are the two points of intersection of the calibration plane with the absolute conic. Therefore the images $\boldsymbol{P}$ and $\boldsymbol{Q}$ of the circular points belong to ω, *i.e.* $\boldsymbol{P}^\top \omega \boldsymbol{P} = 0$ and $\boldsymbol{Q}^\top \omega \boldsymbol{Q} = 0$. In practice, these two complex equations are equivalent, and two real constraints are obtained by isolating real and imaginary components of either equation. Since each plane provide two such points, and a conic is uniquely defined by five points, it is usually required to observe three planes in order to estimate all five intrinsic parameters. If the skew is assumed to be zero, then two planes are sufficient. Practically, it is equivalent to either observe several planes in a single image or to take several images of the same plane viewed from different orientations. The different constraints obtained are linear and a least-square solution can be found by applying techniques similar to the linear methods described in Section 2.3.1. In [175], the author follows the calibration by a non-linear minimisation (bundle adjustment) using LM in which it is possible to incorporate two extra parameters accounting for the radial lens distortion. Obviously a necessary condition for plane-based camera calibration is that the planes have different orientation (otherwise they would intersect the absolute conic in the same circular points, and the system would be under-constrained). An exhaustive study of the singularities is given in [138].

In [96], the square calibration target is replaced by one made of one circle and a pencil of lines passing through the centre of the circle. For each line, the VP is computed from the preservation of the cross-ratio defined by the two intersections of the line with the circle, the centre of the circle, and the PI of the line. All the VPs obtained in such a way are used to fit a line which is the line at infinity of the calibration plane. The image of the circular points are found as the intersection of this line with the image of the circle. The main advantage of this formulation compared to the previous one is that it is not necessary to establish any correspondence between points on the calibration target and points on its images. One drawback however is that it is not possible to compute two of the orientation parameters (this is due to the central symmetry of the target). Alternatively there exist other ways to estimate the circular points. [169] uses a method similar to [175], but uses a planar pattern made of at least three concentric conics to estimate the planar homography, and thereby the IAC. In [89] it is shown that computing two circular points associated with one plane is equivalent to carrying out a metric rectification of

this plane. This can be achieved in a stratified manner. In a first step the affine properties of the scene are recovered by identifying the line at infinity from two or more sets of parallel lines. In a second step the metric properties are recovered by using two constraints which can be a known angle between lines, or equality of two (unknown) angles, or a known length ratio. In this perspective, [175] is equivalent to using right angles and a ratio of one of the edges defined by a square. Right angles are usually the easiest to use because there is an abundance of them in man-made structures.

Calibration of zooming cameras

It has been shown in [138] that the method can be extended to zooming cameras. Two types of zooming models are considered: i) varying focal length only or ii) varying focal length and principal point. In [138], the zooming parameters corresponding to the model chosen result in additional unknowns each time the zoom factor is changed. The constraints defined in [175] can then be expressed with respect to all the unknowns (including zooming parameters) and stacked up in a matrix in order to solve the system by similar technique. A direct consequence of this approach is that the complexity of the system increases rapidly with the number of images (whereas it was constant in [175]), which can lead to convergence problems. In order to guarantee a good conditioning of the system, column rescaling such that columns have equal norms is applied to the matrix containing these constraints [138]. Another issue is the optimality of these methods, in particular it appears that the distance minimised is algebraic and therefore is not physically meaningful. An optimal solution (in the sense it minimises the Cramer-Rao lower bound) is proposed in [95]. However the solution is valid under the assumption that only the focal length is varying (the other parameters have been pre-calibrated). An interesting solution to avoid the increase of the dimensionality when the focal length is varied is given in [62, 61]. The main contribution there is to show that Poncelet's theorem can be used to define some invariants to the focal length. In particular, it is demonstrated that when observing a plane with known metric properties, the camera centre must lie on a circle called the *centre circle* [62]. The centre circle projects onto the image plane in a line segment called the *centre line*, which is the locus of the principal point. Analytic expressions for these curves are given in [62, 61]. It is shown that the centre line is independent of the focal length and can be used to represent a geometric cost function, whose minimisation allows computation of

the coordinates of the principal point and the aspect ratio (it is assumed that the skew is zero). Once these parameters are computed, focal length can be computed for each image in a simple way.

Conclusions

The main advantage of calibration methods based on the IAC is the simplification of the geometry of the calibration pattern used from the traditional 3D grid made of control points to a simple planar pattern which can be produced at low cost with a standard printer.

2.3.4 Calibration using other geometric entities

The two previous sections were entirely dedicated to two very important geometric entities which are VPs and the IAC. In this section, we list a few other calibration methods based on other useful geometric entities. The common characteristic of these techniques is that they exploit some geometric properties of the calibration object, such as invariance, in order to simplify the calibration process.

Lines

After using 3D objects (*e.g.* orthogonal planes) or 2D objects (planes undergoing an unknown motion) it seems natural to consider 1D objects. Lines have been used in camera calibration for different purposes [176, 40]. [176] investigates the requirement for calibrating a single camera from a set of aligned points. In particular, it is demonstrated that it is not possible to calibrate a single camera from a free-moving 1D object, however it becomes possible with three aligned points separated by known distances, if one point is fixed. Camera calibration is possible with a free-moving rigid bar carrying two markers under the requirement that at least two cameras are observing the scene (see for example [18]). 1D calibration objects are of major interest for the calibration of multi-camera set-ups where it is required for all the cameras to observe simultaneously the calibration pattern, which is usually impractical with a 2D or 3D pattern. Another application of lines is for calibrating the distortion [40]. The method is based on the fundamental property that a camera follows the pinhole model if and

only if the projection of every line in space onto the camera is a line. In practice, the method minimises a cost function which measures the total distortion error in all segments in the image. Different distortion models at different orders are accommodated by the method. The only assumption of the method is that there exists straight lines in the scene. The idea originated in the photogrammetry literature under the name of the *plumb line method* [24].

Spheres

Spheres have strong invariance properties which can be used in calibration. In [107], it is shown that the aspect ratio of a camera can be computed from the image of a sphere. The method is based on the observation that, because of the radial distortion, the occluding contour of a sphere appears as a distorted ellipse, which can be approximated by a fourth order polynomial. In practice, such a polynomial is fitted to the extracted occluding contour, and the aspect ratio can be estimated from the coefficients of this polynomial. In [131], it is shown that spheres can also be used to determine the principal point and the focal length. In particular, it is shown that after correction of the lens distortion, the major axis of the ellipse representing the occluding contour goes through the principal point, which can be determined from the intersection of at least two images of a sphere. It is also shown that the focal length is related to some intrinsic properties of the ellipse (eccentricity, length of the major axis and distance from the principal point). More recently, it has been demonstrated that the IAC [144] or its dual [3] can be computed from the outline of three spheres, from which the intrinsic camera parameters can be estimated.

Techniques based on geometric properties of the scene increase the flexibility of the method because they take advantage of the geometric cues present in the scene. The patterns used are common shapes such as edges of building, or patterns produced easily by a standard printer (planar grid).

2.3.5 Active calibration

It is shown in this section that it is possible to replace the knowledge of the geometry of the scene by some knowledge about the motion of the camera. This class of methods is called

active calibration. The main idea is to use some specific controlled motion in order to simplify the computation of the camera parameters, usually by exploiting some invariance properties of the intrinsic parameters with respect to the motion. The motion is dictated by the degrees of freedom of the platform on which the camera is mounted. The most typical motions considered are translation, rotation or planar motion.

Pure rotation of the camera (*i.e.* rotation around the optical centre) is a motion frequently encountered in computer vision. In [131, 132], the author considers the internal calibration of a camera mounted on a rotary stage. The method uses pairs of images separated by a pure rotation with known axis and angle of rotation. In such circumstances, it is possible to predict the location of the features observed in the second image from their position in the first image, given the intrinsic parameters. An error in the intrinsic parameters results in a error in the estimated location of the features. The intrinsic parameters can be estimated by minimising the squared distance between the predicted and measured feature points. The parameters estimated include the focal length, the aspect ratio, the position of the principal point and the radial distortion. The method requires rotation around two orthogonal axes (the X and Y axis of the camera) if all the parameters must be estimated. If it is not required to estimate the aspect ratio, one rotation axis is sufficient. One drawback of the method is that it is necessary to adjust the position of the camera very accurately with respect to the rotary stage in order to ensure that the optical centre coincides with the axis of rotation.

In [10], it is shown that it is possible to compute the focal length, aspect ratio and image centre of a camera carrying small pan, tilt and roll movement, by solving a simple linear system of equations. The approach does not require any calibrated pattern, but uses only scenes with stable edges. In [34], the calibration of a camera with pan, tilt and zoom motion is considered. Similarly to [131, 132], the idea is to search for the parameters that minimise the predicted image and the observed image after zooming or rotating the camera by a known angle. Repeating this approach for a large number of zoom settings yields a look-up table of image magnification and zoom centres, which are then linearly interpolated. After the calibration of the zoom parameters, the other parameters are recovered by generating pure translation motion around the pan and tilt angle. Contrary to [131, 132, 10] which considered sparse features, [34] opted for a dense optical flow approach based on image warping, which makes the method more robust in the case of an outdoor environment.

A method for active calibration of a camera mounted on a robot arm and observing a light spot is presented in [134]. For each camera parameter, a controlled motion is performed (involving either rotation or translation), which defines a cost function whose minimisation results in the parameter value. Each parameter is estimated sequentially in this approach. The originality of the method is that the search for the optimum value is performed directly in the 3D space, which results in the robot doing some repetitive movements until the solution has been found.

There exist many other methods exploiting the properties derived from specific controlled motion. In addition, it is possible to combine these methods with the ones using geometric properties of the scene. For example, in [13], a stationary camera is calibrated from the images of a planar pattern fixed on a turn table, by considering VP properties. It is shown that the locus of the VP generated by the planar pattern is a conic section, which can be used to determine the focal length, principal point and aspect ratio. Similarly, it is shown in [38] that a VP traverses a conic section when the camera moves with an arbitrary translation and a fixed axis of rotation.

Active camera calibration methods use properties of controlled motions to calibrate a camera without requiring any accurate calibration grid. These methods should be considered every time the camera is mounted on an active device possessing the appropriate degrees of freedom to generate the motion required. One limitation of these methods however is that they depend on the assumption that the device on which the camera is mounted is able to generate a known motion. Deviations from the expected motion will usually affect the calibration accuracy.

2.3.6 Auto-calibration

All calibrating methods presented so far exploit some knowledge about either the structure (geometry) of the scene or about the relative motion between the scene and the camera. In either case, the task is onerous because of the requirement of an accurate calibration pattern or the necessity to generate accurate motions of the camera or the object. In addition, calibration must be done before the vision tasks. Auto-calibration (also called self-calibration) relaxes all of these requirements, by estimating the camera parameters directly from a sequence of generic images. This offers great flexibility by allowing calibration to be done, for example, with the same images used for the vision tasks.

It is well-known in computer vision that without any knowledge of the scene and the cameras,

there exists a projective ambiguity in the reconstruction, *i.e.* if a sufficient number of point correspondences are provided, it is possible to estimate the structure of the scene and the camera matrices only up to a projective transformation (see for example [72, 48]). The main idea of auto-calibration is to exploit the rigidity of the scene and some constraints on the intrinsic or extrinsic parameters in order to remove this ambiguity and thereby estimate the camera parameters. Theoretically, the ambiguity can be removed only up to a similarity transformation, *i.e.* it is not possible to compute the absolute positions and orientations of the cameras (neither the scale of the reconstruction). In a nutshell, auto-calibration determines the intrinsic parameters for each camera and the relative position and orientation of the cameras with respect to the first one. It will be observed that the constraints generated are intimately related to the absolute conic.

The original idea of auto-calibration is due to Faugeras *et al.* [49]. Their approach is based on the Kruppa equations, which relate the epipolar transformation to the IAC in the case of cameras with fixed intrinsic parameters. Geometrically, the two epipolar planes tangent to the absolute conic give rise to two epipolar lines tangent to the IAC in each image. However, because the IAC is invariant under a rigid motion, it produces two constraints from the correspondence of the tangents in the two images, which are represented algebraically by the Kruppa equations. Since the IAC is determined by five parameters, three different cameras are sufficient to solve for all the intrinsic parameters. The Kruppa equations are quadratic, therefore there exist multiple solutions, and their computation is usually difficult. In addition, the Kruppa equations define constraints on pairs of images rather than the whole sequence, which results in weaker constraints and more ambiguities.

A second approach to auto-calibration is *stratification*. In [47], the world is described as a succession of strata: projective, affine and Euclidian (or metric). In this framework, auto-calibration is broken down into two steps. In the first step, affine properties are recovered from an initial projective reconstruction by identifying the plane at infinity, while the second step consists in recovering the Euclidian properties via the identification of the absolute conic. The most difficult task among the two is usually to identify the plane at infinity. Pollefeys and Van Gool define a constraint called the *modulus constraint* which can be used for this purpose in the case of a camera with fixed intrinsic parameters [109]. The constraints are non-linear, however the number of unknowns involved is limited to three, which simplifies their estimation. The

recovery of the metric properties follows from the constraint that the IAC should be the same for all views. This results in linear equations.

A third approach is based on the *absolute dual quadric* [151]. The absolute dual quadric is the quadric represented by the 4×4 matrix:

$$Q_\infty^* = \begin{bmatrix} I & \mathbf{0} \\ \mathbf{0}^\top & 0 \end{bmatrix}. \tag{2.16}$$

Geometrically it consists of the planes tangent to Ω_∞. The advantage of using the absolute dual quadric is that it encodes both the plane at infinity and the absolute conic at the same time. The dual absolute quadric projects into the Dual Image of the Absolute Conic (DIAC) $\omega^* = \omega^{-\top} = KK^\top$. As with the absolute conic, the dual absolute quadric is fixed under a rigid motion of the camera. This property, in addition to some constraints on the intrinsic parameters, can be used to compute the camera parameters. Depending on the type of constraints on the intrinsic parameters (*e.g.* known principal point, zero skew, known aspect ratio, constant intrinsic parameters...), the constraints obtained are either quadratic or linear.

The early auto-calibration methods considered only the calibration of cameras under the assumption of constant intrinsic parameters. It is however possible to auto-calibrate a camera with less restrictive constraints, using for example only the zero-skew assumption [109]. Auto-calibration may look like an attractive solution, however one criticism is the lack of stability of the method. Usually a good initialisation is required, and even though it is the case, convergence is not always guaranteed (see [20] for an evaluation of self-calibration). In addition, there exist some critical motion sequences for which the solution is ambiguous. A taxonomy of the different critical motion sequences is given in [135], in the case of constant intrinsic parameters, and in [137], in the case of zooming (variable focal length) cameras. In practice some important cases of critical motion sequences occur for orbital motion, planar motion, pure translation or rotation. Examples of algorithms for auto-calibration of a rotating camera are given in [66] in the case of constant intrinsic parameters, or in [122, 2] in the case of zooming cameras. The case of a camera undergoing a translation or planar motion are treated respectively in [97] and [7]. In addition, it is interesting to note that Triggs proposed an auto-calibration method from images of a plane [150].

2.4 Conclusion

There exists a multitude of camera calibration techniques. From the most constraining approach requiring accurately located features in the scene, to auto-calibration which on the contrary does not require any information about the structure or the motion of the camera, a very rich collection of methods has been encountered. Some of them are based on geometric properties of the scene, while others focus on specific motions of the camera. There is usually a trade-off between accuracy and flexibility of the methods. For example if auto-calibration is the most flexible technique, it is also the least stable and accurate, while methods based on point correspondences remain the most trusted techniques when high accuracy is required.

Many methods have one common characteristic: the use of invariance properties. Invariance properties can for example be defined with respect to some geometric entities or with respect to some specific camera motions. The geometric entities involved can be concrete objects such as points, lines or spheres, but also imaginary objects such as the absolute conic or the absolute dual quadric used in auto-calibration. Invariants allow decoupling of the camera parameters and define constraints on subsets of the parameters. This is a powerful property because it implies reduction of the number of unknowns solved simultaneously. The rest of this part of the thesis investigates how invariants can be applied to increase the accuracy of camera calibration.

Chapter 3

Calibration of a translating camera using Points at Infinity

3.1 Introduction

The key idea developed in this chapter and the following one is that invariants can be used to increase the accuracy of the camera calibration process. It has already been observed in the previous chapter that specific geometric entities (real or even imaginary) and also specific motion sequences exhibit some invariance properties which can be used during camera calibration. In this perspective, this chapter presents a novel camera calibration method based on the invariance properties of Points at Infinity (PI) to decouple the translation component from the other camera parameters. It also gives some insight into the influence of the use of invariants on the accuracy of the estimation of the camera parameters in the case of this novel method.

There are two main motivations for decoupling the translation parameters from the other camera parameters. Firstly, decomposition in the parameter space leads to simpler sub-problems. Secondly, if the translation parameters are decoupled from the others, data from additional images obtained by translation does not introduce additional parameters to the problem. That is, the data size can be increased, and thereby estimation accuracy, without increasing the problem dimensionality. The idea of parameter decomposition has been used in other areas of computer vision. In [8], it was shown that for two collections of 3D points related by a rotation and trans-

lation the estimation of the motion can be decoupled based on the properties of the centroid. A similar problem for 2D motion projections is described in [74] and [115]. In the case of motion estimation from line correspondences, the direction of lines has been used to compute the rotational part of the transformation [9]. Additionally, work on shape matching has considered the decomposition of rotation and translation for a 2D transformation of planar shapes [4].

Our approach presents some similarities with previous methods based on Vanishing Points (VPs) [26, 159, 43, 28, 160, 12, 33, 88]. VPs have strong invariance properties, however, there are usually not many in images and they are difficult to compute. Even if considerable effort has been directed towards their estimation [123], existing VP-based calibration methods usually require pairs of parallel scene lines to define VP location. In contrast, our approach computes a Point at Infinity (PI) from known single straight lines in the scene, and formulates a novel constraint which relates single line orientation to the projections of the PI. Thus, equations linking scene and image data can be expressed independently of translation. Methods that use lines for the estimation of motion and structure have been previously considered in [90, 162]. A discussion of the advantages of the use of lines in terms of accuracy in measurements is given in [162]. If orientation of lines can be more accurately and reliably measured than point location, then this results in more accurate features. In an implementation, straight lines can be defined by edges in the scene or by pairs of points; generally their number exceeds the number of parallel scene line pairs or usable distinguished image points (such as corners) that are necessary for many other calibration methods.

In Section 3.2, the use of PI in the inverse image formation problem is considered and an invariant for the equations linking the coordinates of 3D points and their projections is defined. Section 3.3 formulates the two stage camera calibration procedure based on this invariant. Finally experimental results with synthetic and real data are discussed in Section 3.4.

3.2 Inverse image formation and Points at Infinity

It is assumed in this chapter that image formation is modelled by a standard pinhole camera as described in Eq. (2.7). For generality all the intrinsic parameters are included in the model, *i.e.* the camera matrix is of the form described in Eq. (2.5). The equation mapping a 3D scene

point $\boldsymbol{P}_i = (X, Y, Z, 1)^\top$ to the corresponding 2D image point $\boldsymbol{p}_i = (u, v, 1)^\top$ under such conditions is summarised below

$$\boldsymbol{p}_i \sim K[R|\boldsymbol{t}]\boldsymbol{P}_i\,, \quad \text{with} \quad K = \begin{bmatrix} f & -f\cot\theta & u_0 \\ & fr/\sin\theta & v_0 \\ & & 1 \end{bmatrix}. \tag{3.1}$$

Many approaches to camera calibration have been described in the previous chapter. The majority of the methods, including the Gold Standard algorithm described in [72], use point-based information and simultaneously estimate all the parameters by minimising a functional of the form described in Eq. (2.12). This cost function can be generalised to a sequence of images by extending the sum to all the images. The resulting cost function is called the *re-projection error*, and its minimisation leads to the *Maximum Likelihood* (ML) estimate under some standard hypotheses on the noise distribution (measurement errors are Gaussian, see *e.g.* [72], pp 86–87). When camera motion is considered, the solution can involve a large number of parameters. However, if invariants are used, it can be simplified such that the minimum depends only on a subset of the parameters. For example, VPs can be used to decouple the camera position from the other parameters. Contrary to previous methods using VPs [26, 159, 43, 28, 160, 12, 33, 88], the decomposition method proposed here does not require a calibration pattern containing parallel lines, but can be implemented from arbitrary lines or pairs of points in a known scene. The only requirement for our method is that the directions defined by the lines or pairs of points are known.

It has already been observed in the previous chapter that the projection of a PI in an image is a VP, and that an important property VPs is that they are independent of camera translation. PI are defined by the direction of straight lines in the scene. If a pair of points $\boldsymbol{P}_i = (X_i, Y_i, Z_i, W_i)^\top$ ($W_i \neq 0$) and $\boldsymbol{P}_j = (X_j, Y_j, Z_j, W_j)^\top$ ($W_j \neq 0$) is considered, the direction of the line $(\boldsymbol{P}_i\boldsymbol{P}_j)$ is represented by the PI $\boldsymbol{D}_{ij} = W_i\boldsymbol{P}_j - W_j\boldsymbol{P}_i$, which can be written in the form $\boldsymbol{D}_{ij} = (\boldsymbol{d}_{ij}^\top, 0)^\top$, where $\boldsymbol{d}_{ij}$ is the vector formed by the first three components of $\boldsymbol{D}_{ij}$. The projection of $\boldsymbol{D}_{ij}$ into the image defines a VP $\boldsymbol{v}_{ij} \sim K[R|\boldsymbol{t}]\boldsymbol{D}_{ij} = KR\boldsymbol{d}_{ij}$ which is translation invariant. Since KR can be interpreted as an homography, and an homography preserves collinearity, $\boldsymbol{v}_{ij}$ must lie on the line $(\boldsymbol{p}_i\boldsymbol{p}_j)$ (see Fig. 3.1), that is $\boldsymbol{l}_{ij}^\top\boldsymbol{v}_{ij} = 0$, where $\boldsymbol{l}_{ij} \sim \boldsymbol{p}_i \times \boldsymbol{p}_j$ is the homogeneous representation of $(\boldsymbol{p}_i\boldsymbol{p}_j)$. With the notation $H = KR$,

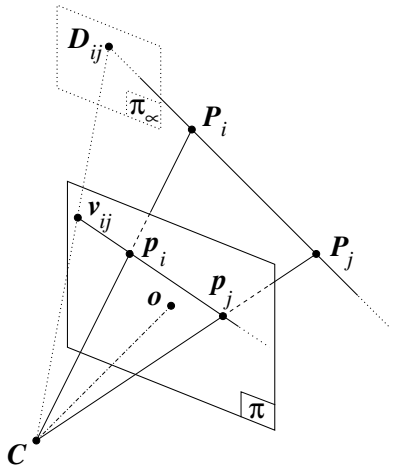


Figure 3.1: *Projection of a pair of 3D points in an image. The pair of 3D points* $(\boldsymbol{P}_i, \boldsymbol{P}_j)$ *defines a direction which is represented by a point* $\boldsymbol{D}_{ij}$ *in the plane at infinity* $\boldsymbol{\pi}_\infty$*. This point projects into a VP* $\boldsymbol{v}_{ij}$ *which is constrained to lie on the image line passing through* $\boldsymbol{p}_i$ *and* $\boldsymbol{p}_j$*.*

the following equation independent of the translation is obtained:

$$\boldsymbol{l}_{ij}^{\top} H \boldsymbol{d}_{ij} = 0\,. \tag{3.2}$$

Thus the minimisation problem originally defined in Eq. (2.12), in terms of the distances between observed and estimated points, can now be reformulated in terms of the distance between observed image lines and their corresponding estimated VP, by minimising the cost function defined below

$$\sum_{i,j} d(\boldsymbol{l}_{ij}, H\boldsymbol{d}_{ij})^2\,. \tag{3.3}$$

In this equation, d denotes the distance between observed and estimated points. This general notation is deliberate; it will be seen next that different expressions can be considered for this distance, thus defining different cost functions. It is important to note that such cost functions involve only intrinsic and orientation parameters. Once these parameters have been determined, the translation can be computed by considering Eq. (3.1) for known K and R matrices. As such, the original minimisation problem can be divided into two sub-problems. The next section shows how this decomposition can be applied in the context of camera calibration.

3.3 Application to camera calibration

The general problem of computing all the camera parameters from one or several images related by a translation, using a camera calibration object, is considered. The calibration object consists either of 3D lines with known directions, or 3D points with known coordinates (pairs of points are used to define lines with known directions in this case). In total, the system has $8 + 3n$ unknowns (where n is the number of images): 5 for K, 3 for R and, for each image, 3 for $\boldsymbol{t}$. In general, when correspondences between 3D points and their images are known, the camera parameters can be computed simultaneously by solving for a matrix $M = K[R|\boldsymbol{t}]$ satisfying $\boldsymbol{p}_i \sim M\boldsymbol{P}_i$ for each world to image point correspondence, as described previously in Section 2.3.1. Alternatively, the results of Section 3.2 can be used to decompose the full parameter space into two smaller sub-systems. The first one contains only the parameters from K and R (8 parameters), whereas the second one contains the remaining parameters from $\boldsymbol{t}$ (there are n independent systems of 3 parameters, one for each image). Thus two simpler problems are defined:

1. *Intrinsic parameters and orientation estimation:* Given a set of world directions $\boldsymbol{d}_{ij}$ and the associated image lines $\boldsymbol{l}_{ij}$, compute a 3×3 full rank matrix $H = KR$ such that $\boldsymbol{l}_{ij}^{\top} H \boldsymbol{d}_{ij}$ is minimised for each (i, j).

2. *Position estimation:* Given a set of world to image point correspondences $\boldsymbol{P}_i$ and $\boldsymbol{p}_i$, and two known matrices K and R, compute a vector $\boldsymbol{t}$ such that $\boldsymbol{p}_i \sim K[R|\boldsymbol{t}]\boldsymbol{P}_i$ for each i.

Note that there is no restriction on the nature of the translation motion followed by the camera. The translation can be arbitrary, for example it does not have to be restricted to a single linear or planar path. The only requirement is that the orientation of the camera does not change.

The 3×3 matrix H defined in Problem 1 is the homography between the plane at infinity and the image plane. Once H is known, K and R can be recovered by a simple RQ decomposition [72] (p 150). However, Problem 1 is different from a simple homography estimation problem. Namely, there exists no strict correspondence, but only a constraint that establishes that a VP should lie on the image line. This is fundamentally different to other camera calibration methods that propose computing the VP from parallel lines before estimating the camera parameters

[26, 159, 43, 28, 160, 12, 33, 88]. It should also be noted that the equations defined in Problem 1 are similar to the estimation of the fundamental matrix via the 8-point algorithm [67]. In the case of the fundamental matrix estimation however, the solution matrix must be degenerate to satisfy the singularity constraint (rank 2), whereas in this case a full rank matrix (rank 3) is sought. Depending on the application, it is not always required to solve Problem 2. If required, it can be solved in a rather straightforward manner using least-squares techniques. For this reason the focus in this section is on solving Problem 1. The reader interested in the details of the resolution of Problem 2 is referred to Appendix A.

3.3.1 Practicality

Problem 1 uses only the directional information contained in the scene, derived from 3D lines with known direction, or pairs of known 3D points. One practical advantage of the decomposition method is that it can be used in situations where only directional information is present in the scene, as for example in the case of architectural applications [33, 88], wherein directions are defined by edges of buildings. The method is also useful in situations when one is interested only in the intrinsic parameters or in the orientation of the camera. Another advantage over standard camera resectioning is that when multiple images obtained from a purely translating camera are used, data size is increased, without requiring to compute additional translation parameters. Accurate calibration can therefore be done from directional information only. If all the camera parameters are required, a full camera calibration is performed by solving both problems sequentially. In that case, Problem 1 still uses only directional information, while Problem 2 requires additional information, such as point correspondences.

3.3.2 Linear solution

In this section, a simple linear solution is developed. In comparison to non-linear methods, linear methods are significantly faster and easier to implement. It should be noted that Eq. (3.2) being structurally identical to the equation relating the fundamental matrix to image points in correspondence in two images, similar methods can be applied. Our solution is similar to the eight-point algorithm described in [67]. One important difference however is that the matrix H has rank 3, while the fundamental matrix is a rank-2 matrix.

Writing $\boldsymbol{l}_{ij} = (l_{ij_x}, l_{ij_y}, l_{ij_z})^\top$ and denoting by $\boldsymbol{h}$ the entries of H in row major order, *i.e.*

$$\boldsymbol{h} = \begin{pmatrix} \boldsymbol{h}_1 \\ \boldsymbol{h}_2 \\ \boldsymbol{h}_3 \end{pmatrix} \qquad \text{where} \quad H = \begin{bmatrix} \boldsymbol{h}_1^\top \\ \boldsymbol{h}_2^\top \\ \boldsymbol{h}_3^\top \end{bmatrix},$$

a straightforward development of Eq. (3.2) leads to the following equation which is linear in the unknown $\boldsymbol{h}$:

$$(l_{ij_x}\boldsymbol{d}_{ij}^\top, l_{ij_y}\boldsymbol{d}_{ij}^\top, l_{ij_z}\boldsymbol{d}_{ij}^\top)\boldsymbol{h} = \boldsymbol{a}_{ij}^\top \boldsymbol{h} = 0\,. \tag{3.4}$$

Each direction defines one such constraint on the unknowns. From a set of n directions, a $n \times 9$ matrix A is obtained by stacking up the terms $\boldsymbol{a}_{ij}^\top$ defined in Eq. (3.4) for each direction. The vector $\boldsymbol{h}$ is then computed by solving the linear system $A\boldsymbol{h} = \boldsymbol{0}$. The system has 8 unknowns (H has 9 entries, but it is defined up to a non-zero scale factor), therefore a minimal solution is obtained from 8 directions in a general position. The term general position will be clarified in Section 3.3.4, where the degenerate configurations will be described. In the case of exactly 8 correspondences, A has rank 8 and the solution is obtained by searching for its right null-space. The null-space being of dimension 1, the corresponding solution is defined up to a scale factor. This is consistent with the fact that the homogeneous solution matrix H is also defined up to a non-zero scale factor.

In practice it is best to consider a large number of correspondences in order to diminish the influence of noise and increase the accuracy of the solution computed. The resulting system of equations is overconstrained, and because of noise, there exists generally no solution satisfying exactly $A\boldsymbol{h} = \boldsymbol{0}$. In the absence of an exact solution, an approximate solution minimising an appropriate cost function is sought. It has been chosen here to minimise the residual error defined by $\|A\boldsymbol{h}\|$. This error has no direct physical meaning; for this reason it is sometimes called the algebraic error, in contrast to other measures which minimise for example a geometric distance (see [70]). It is necessary to enforce another constraint during minimisation in order to avoid the trivial solution $\boldsymbol{h} = \boldsymbol{0}$. Several constraints have been considered, however minimisation subject to the constraint that $\|\boldsymbol{h}\| = 1$ is one of the most common, and it has been shown to produce good results in the case of other applications (see [67, 70]), and for this reason has been also considered here. This is a standard minimisation problem and it is well-known that its solution is the unit eigenvector corresponding to the smallest eigenvalue

of $A^\top A$ [67]. A simple way for computing this eigenvector is for example the Singular Value Decomposition (SVD) algorithm [112]. The method is summarised in Algorithm 1. It should be noted that when a large number of correspondences is considered, A has a very large number of rows, and it may not be possible to carry out the SVD due to memory limitations. A simple solution is to replace the original system of linear equation $A\boldsymbol{h} = \mathbf{0}$, by the system of normal equations $A'\boldsymbol{h} = \mathbf{0}$ with $A' = A^\top A$, which has dimension 9×9. These two systems are mathematically equivalent, however in practice the second implementation is preferred because it has a constant complexity and memory requirement.

Algorithm 1 Basic linear computation of KR

1. For each world direction $\boldsymbol{d}_{ij}$ and the associated image line $\boldsymbol{l}_{ij}$, compute the vector $\boldsymbol{a}_{ij}$ defined in Eq. (3.4).
2. Stack up all the vectors $\boldsymbol{a}_{ij}$ into a single $n \times 9$ matrix A.
3. Compute the SVD of A (or $A^\top A$ if considering the normal equations). After decomposition, the matrix is written in the form $A = UDV^\top$, where U and V are two orthogonal matrices and D is a diagonal matrix with positive diagonal entries, arranged in descending order down the diagonal. $\boldsymbol{h}$ is given by the last column of V.
4. $H = KR$ is obtained from $\boldsymbol{h}$.

Normalisation

It has been proven in a recent stream of work [80, 67, 70, 98, 85, 100, 99, 30, 77, 31] that without an appropriate normalisation scheme, algorithms minimising an algebraic distance are usually bound to perform poorly. In this section, similar considerations are applied to the novel linear calibration method developed. We chose to apply a similar normalisation strategy as the one described in [67]. Other techniques in agreement with more recent publications could have been considered, however the technique chosen has the advantage of being simple to implement, and lead to very good results for our application. Given the similarity between the equations defined in Problem 1 and the ones defined for the fundamental matrix in [67], many of the results demonstrated there apply directly to our case, and will therefore not be proven

again in this section.

Ideally, the result of the camera calibration should be independent of the choice of the coordinate system (origin and scale) for each image. However, it has been shown in [67, 70] that this is not the case, and that in practice some reference frames will give better results than others. The difference in accuracy observed can be attributed to the numerical condition of the system of equations involved. The aims of the normalisation are therefore: i) to eliminate the undesirable effects due to the arbitrary choice of the origin and scale for measuring the data, ii) to improve the numerical stability of the algorithm used to solve the system.

Mathematically, a change of coordinate system is equivalent to applying a similarity transformation to the input data. For this reason, two homographies are considered in order to represent the possible transformations. The first homography T affects the image data, transforming the end-points $\boldsymbol{p}_i$ of each image segment into $\tilde{\boldsymbol{p}}_i \sim T\boldsymbol{p}_i$, or equivalently transforming the image lines $\boldsymbol{l}_{ij} \sim \boldsymbol{p}_i \times \boldsymbol{p}_j$ into $\tilde{\boldsymbol{l}}_{ij} \sim \tilde{\boldsymbol{p}}_i \times \tilde{\boldsymbol{p}}_j \sim T^{-\top}\boldsymbol{l}_{ij}$. The second homography T' affects the scene data, transforming the PI $\boldsymbol{d}_{ij}$ into $\tilde{\boldsymbol{d}}_{ij} \sim T'\boldsymbol{d}_{ij}$. Denoting by H and $\tilde{H}$ the matrices defined in Eq. (3.2) respectively before and after transformation of the input data, we can write that

$$(\tilde{\boldsymbol{p}}_i \times \tilde{\boldsymbol{p}}_j)^\top \tilde{H}\tilde{\boldsymbol{d}}_{ij} = \left[T^{-\top}(\boldsymbol{p}_i \times \boldsymbol{p}_j)\right]^\top \tilde{H}T'\boldsymbol{d}_{ij} = (\boldsymbol{p}_i \times \boldsymbol{p}_j)^\top T^{-1}\tilde{H}T'\boldsymbol{d}_{ij} = 0\,,$$

and it results that $H = T^{-1}\tilde{H}T'$. This shows that there exists a one-to-one correspondence between H and $\tilde{H}$. However, it has been shown in [67] that these solutions do not give rise to the same error subject to the constraint $\|\boldsymbol{h}\| = \|\tilde{\boldsymbol{h}}\| = 1$. In fact the smallest unit eigenvector for the first equation matrix A is usually not an eigenvector for the other equation matrix A'. This means that the solution obtained is expected to vary according to the reference frame chosen. The question that then naturally arises is which transformations to apply in order to define a canonical frame where the results are optimal.

To answer this question it is necessary to consider numerical stability. Our linear method computes the unit eigenvector corresponding to the smallest eigenvalue of $A^\top A$. It has been shown in [67, 60] that the accuracy of the computation of this eigenvector is related to the condition number of the system matrix, which is defined as the ratio of the largest to the smallest eigenvalue. In order to reduce the sensitivity to small perturbations, and thereby increase the accuracy of the computation of the eigenvector, the condition number must be made as close to unity as possible. After observing that the major reason for poor conditioning of $A^\top A$ is

the lack of homogeneity in the input data used to construct this matrix, Hartley shows that there exists a simple strategy based on translating and scaling the input data. Because of the similarity of the two problems, it is possible to adopt a similar normalisation scheme, which is described below.

The end-points of the image lines with homogeneous coordinates $(x, y, w)^\top$ can be treated exactly like the input points described in [67], *i.e.* they are normalised such that they satisfy

$$\begin{cases} \sum_{i=1}^{n} x_i = \sum_{i=1}^{n} y_i = 0\,, \\ \frac{1}{n}\sum_{i=1}^{n} x_i^2 = \frac{1}{n}\sum_{i=1}^{n} y_i^2 = 1\,, \\ \forall i \quad w_i = 1\,. \end{cases} \tag{3.5}$$

In practice, such a normalisation is achieved by first translating the end-points such that their centroid is at the origin, and then scaling them such that their two principal moments are both equal to unity. After transformation, the data forms a circular cloud of points of average radius one about the origin. Alternatively, the normalisation could have been applied directly to the coordinates of the image lines $\boldsymbol{l}$, however this requires a different normalisation method, like the one proposed for $\boldsymbol{d}$, because the last coordinate of $\boldsymbol{l}$ is not guaranteed to be non-zero (see next paragraph).

In the case of $\boldsymbol{d} = (U, V, W)^\top$, it is also possible to make an analogy with the input points considered in [67], however the major difference is that the set of points is not guaranteed to be bounded in this case. In particular, the PI corresponding to lines parallel to the XY plane are of the form $(U, V, 0)^\top$, *i.e.* they are located at infinity. Thus, the previous normalisation framework is no longer possible, because concepts such as the centroid are not defined or would have too large values. To address this limitation, PI are transformed such that they satisfy

$$\begin{cases} \sum_{i=1}^{n} U_i = \sum_{i=1}^{n} V_i = 0\,, \\ \sum_{i=1}^{n} U_i^2 = \sum_{i=1}^{n} V_i^2 = \sum_{i=1}^{n} W_i^2\,, \\ \forall i \quad U_i^2 + V_i^2 + W_i^2 = 1\,. \end{cases} \tag{3.6}$$

This can be achieved by first translating and scaling the data in order to satisfy the first two equations, then normalising each point so that their norm is one, and iteratively repeating these two procedures until convergence. In practice, convergence is obtained after only a few iterations. This normalisation scheme was suggested in [72] for application in the case when some of the points are at or near infinity.

The whole procedure is summarised in Algorithm 2. In practice, experimental results with the image of a synthetic grid used for calibration in Section 3.4.1 showed a reduction in the condition number of the matrix of normal equations $A^\top A$ from 6.7×10^7 to 12.5, when data normalisation is carried out. This confirms that the normalisation scheme adopted is appropriate.

Algorithm 2 Normalised linear computation of KR

1. *Normalisation of* $\boldsymbol{l}$*:* Compute a similarity transformation T, consisting of a translation and scaling, that takes the end-points points of the line segments $\boldsymbol{l}_{ij}$ to a new set of end-points centred at the origin $(0, 0)^\top$ and such that the two principal moments are equal to unity. Compute the coordinates of the normalised line-segments $\tilde{\boldsymbol{l}}_{ij}$ (alternatively apply the same normalisation as for $\boldsymbol{d}$ directly to the line coordinates $\boldsymbol{l}$).

2. *Normalisation of* $\boldsymbol{d}$*:* Compute a similarity transformation T', consisting of a translation and scaling, that takes the points $\boldsymbol{d}_{ij}$ to a new set of points $\tilde{\boldsymbol{d}}_{ij}$ with homogeneous coordinates (U_{ij}, V_{ij}, W_{ij}) satisfying Eq. (3.6).

3. *Linear solution:* Apply Algorithm 1 to the set $\tilde{\boldsymbol{d}}_{ij} \leftrightarrow \tilde{\boldsymbol{l}}_{ij}$ to obtain $\tilde{Q}$.

4. *De-normalisation:* Set $H = T^{-1}\tilde{Q}T'$.

3.3.3 Minimisation of a geometric distance

The linear solution is computationally attractive. However, it presents some limitations such as the non-invariance to the coordinate reference frame, which required to introduce normalisation. Another criticism of this method is that the algebraic distance it minimises has little physical meaning. In this section, a non-linear method which minimises a geometric distance is introduced.

The geometric distance d_{geom} from a VP $\boldsymbol{v}$ with homogeneous coordinates $(u, v, w)^\top$ $(w \neq 0)$ to the corresponding line $\boldsymbol{l}$ with homogeneous coordinates $(a, b, c)^\top$ is defined as the shortest distance from a point to a line in the image plane, and is given by the following standard result from geometry:

$$d_{\text{geom}}(\boldsymbol{v}, \boldsymbol{l}) = \frac{1}{\sqrt{a^2 + b^2}} \left| a\frac{u}{w} + b\frac{v}{w} + c \right| . \tag{3.7}$$

Direct minimisation of the sum of squared geometric distance can lead to inaccurate results because some individual measurements with large uncertainties may corrupt the overall sum. There are two main situations in which large uncertainties are expected. The first case is when short image segments are observed; in this case, starting point and end point are so close that the computation of the line coordinates is usually inaccurate. The second case is when the scene direction observed is nearly parallel to the image plane; in this case the corresponding VP is located near infinity and therefore far away from the observed image line. In both cases, the computation of the point-to-line distance is inaccurate because either the line or the point have large associated uncertainties. It is proposed here to evaluate the uncertainty in the computation of the geometric distance associated with each pair ($\boldsymbol{l}_{ij}$, $\boldsymbol{d}_{ij}$), in order to compute a weighted sum of squared distances.

The overall uncertainty can be represented by the covariance matrix of the vector of geometric distances. In order to simplify the estimation of the covariance matrix, we make the following assumptions:

- Image lines are defined by their end-points, and the (x, y) coordinates of each end-point follow independent Gaussian distributions centred at the true end-point coordinates and with standard deviation σ for each coordinate,
- Scene directions are represented by pairs of 3D points whose coordinates follow independent Gaussian distributions centred at the true 3D point coordinates and with standard deviation σ' for each coordinate,
- Errors in pairs of image lines and 3D directions in correspondence are assumed independent.

Because the errors in the inputs defining the different pairs ($\boldsymbol{l}_{ij}$, $\boldsymbol{d}_{ij}$) are independent, the error in the geometric distance corresponding to this pair can be represented by its variance or by its standard deviation σ_{geom}. The geometric distance can therefore be corrected by multiplication by the inverse of the standard deviation σ_{geom}. The cost function to be minimised is the sum of squared distances defined by:

$$\sum_{i,j} \frac{1}{\sigma_{\text{geom}}^2} d_{\text{geom}}(\boldsymbol{v}_{ij}, \boldsymbol{l}_{ij})^2 \,. \tag{3.8}$$

This is comparable to the Mahalanobis distance. Its minimisation should lead to an optimum solution which takes into account the distribution of errors present in individual geometric distances.

We must now determine σ_{geom} for a given pair of image lines and VP. The distribution of d_{geom} depends on both the distribution of $\boldsymbol{l}$ and $\boldsymbol{v}$. We first show that under the assumptions made, all VPs have the same associated covariance and therefore only the covariance in the image lines $\boldsymbol{l}$ needs to be considered. VPs are related to points at infinity $\boldsymbol{d}$ by the relation $\boldsymbol{v} = \mathsf{H}\boldsymbol{d}$. This defines a linear relation. However we know that all the 3D directions have the same covariance because of the assumptions made. It results immediately from the linearity of the previous relation that all VPs also have the same covariance.

Given an image line with coordinates $\boldsymbol{l} = (a, b, c)^\top$ and a VP with coordinates $\boldsymbol{v} = (u, v, w)^\top$, we compute an approximation of the variance of $d_{\text{geom}}(\boldsymbol{v}, \boldsymbol{l})$ using error propagation as described in [72] (pp 123–125). Each coordinate of the end points $\boldsymbol{p}_i = (x_i, y_i, 1)^\top$ and $\boldsymbol{p}_j = (x_j, y_j, 1)^\top$ of the observed image segment $\boldsymbol{l}$ are independent variables following a Gaussian distribution centred at the exact location of the end point, and with standard deviation σ, therefore it can be shown that a first-order approximation of the variance of the distribution of $d_{\text{geom}}(\boldsymbol{v}, \boldsymbol{l})$ is:

$$\sigma^2_{\text{geom}} = \boldsymbol{J}\boldsymbol{\Sigma}\boldsymbol{J}^\top , \tag{3.9}$$

where

$$\boldsymbol{\Sigma} = \sigma^2 \begin{bmatrix} 2 & 0 & -x_i - x_j \\ 0 & 2 & -y_i - y_j \\ -x_i - x_j & -y_i - y_j & x_i^2 + x_j^2 + y_i^2 + y_j^2 \end{bmatrix} , \tag{3.10}$$

and

$$\boldsymbol{J} = \frac{1}{\sqrt{a^2 + b^2}} \begin{bmatrix} \frac{u}{w} - \frac{a(a\frac{u}{w} + b\frac{v}{w} + c)}{a^2 + b^2} & \frac{v}{w} - \frac{b(a\frac{u}{w} + b\frac{v}{w} + c)}{a^2 + b^2} & 1 \end{bmatrix} . \tag{3.11}$$

$\boldsymbol{\Sigma}$ is the covariance matrix of the distribution of the image lines and $\boldsymbol{J}$ is the Jacobian matrix of the transformation mapping an image line and VP to a geometric distance, evaluated at $(\boldsymbol{v}, \boldsymbol{l})$. The details of the computation are given in Appendix B.

The distance d_{mah} is non-linear. A solution can be computed by using standard non-linear minimisation algorithms, such as the Levenberg-Marquardt (LM) algorithm [112] initialised with the result of the linear method. In practice, it has been observed that the choice of the weights

is not very important. During the minimisation, the orientation is parameterised by a three-vector using the Rodriguez formula, as recommended in [152]; this eliminates the problems of singularities which can appear with other parameterisation such as Euler angles. In the parameterisation, the three parameters define a vector parallel to the rotation axis whose magnitude represents the rotation angle [46].

3.3.4 Degenerate configurations

It has been seen in Section 3.3.2 that at least eight world directions $\boldsymbol{d}_{ij}$ and their associated image lines $\boldsymbol{l}_{ij}$ in a "general position" are necessary to compute H. In this section, the term "general position" is clarified and a comparison with the degeneracies in the case of camera resectioning [25] [72] (chapter 21) is given. It is assumed that there are at least eight 3D directions and their associated image lines. The general study of the degeneracies is made in the case of a single camera, however the study generalises easily to the case of multiple translated cameras, by noticing that translating the camera is actually equivalent to introducing additional translated 3D lines in the scene.

Let us now suppose that there is a degeneracy. There exists two distinct rank 3 matrices H and H' satisfying Eq. (3.2) for all (i, j). It results immediately from the bilinearity of Eq. (3.2) that $H_\theta = H + \theta H'$ (where θ is a scalar value) is also a solution. However the determinant of this matrix $\det(H_\theta)$ is a real-coefficient polynomial of degree 3 in θ, thus it has at least one real solution θ_0 different from zero (if $\theta_0 = 0$, then $H_\theta = H$, and H_θ has rank 3, which contradicts the fact that its determinant is zero). H_{θ_0} does not have full rank, *i.e.* $\text{rank}(H_{\theta_0}) \leq 2$, because by construction $\det(H_{\theta_0}) = 0$. In addition, it is clear that $\text{rank}(H_{\theta_0}) \neq 0$, otherwise there would exist a non-zero θ such that $H + \theta H' = 0$, *i.e.* $H \sim H'$, which contradicts the original assumption that the solutions are distinct. We conclude that there exists a matrix H_{θ_0} which has rank 1 or 2. According to the rank of this degenerate matrix, two types of degeneracies are defined. Both configurations are described below. It is important to note that they can occur for any number of 3D lines (not restricted to the minimum case of eight lines), as long as the features are arranged according to the characteristic patterns defined below.

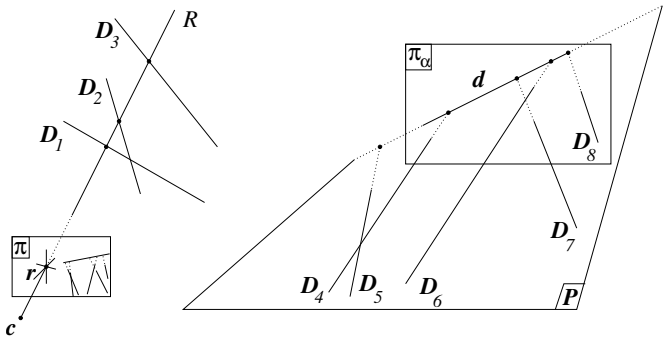


Figure 3.2: *The "rank 1" degenerate configuration. The 3D lines either form a Linear Line Complex with a ray $\mathcal{R}$ going through the camera centre $\mathbf{c}$ (lines $\boldsymbol{D}_1$, $\boldsymbol{D}_2$, $\boldsymbol{D}_3$), or are parallel to a plane $\mathcal{P}$ (lines $\boldsymbol{D}_4, \ldots, \boldsymbol{D}_8$).*

"Rank 1" degeneracy

A "rank 1" degeneracy occurs if there exists a ray $\mathcal{R}$ going through the camera centre and a plane $\mathcal{P}$, such that all the 3D lines either intersect $\mathcal{R}$ or are parallel to $\mathcal{P}$ (see Fig. 3.2). The set of 3D lines intersecting at a common line $\mathcal{R}$ forms a pattern called *Linear Line Complex* (see [133]). A proof of this result is now given. Given that the degenerate solution matrix has rank 1, it can be written in the form $H_\theta = \boldsymbol{r}\boldsymbol{d}^\top$, where $\boldsymbol{d}$ is a 3-vector orthogonal to the null-space of H_θ, and $\boldsymbol{r}$ is a 3-vector in the span of H_θ. Replacing in Eq. (3.2), we obtain $\boldsymbol{l}_{ij}^\top \boldsymbol{r}\boldsymbol{d}^\top \boldsymbol{d}_{ij} = 0$, which is equivalent to $\boldsymbol{l}_{ij}^\top \boldsymbol{r} = 0$ or $\boldsymbol{d}^\top \boldsymbol{d}_{ij} = 0$. The first case $\boldsymbol{l}_{ij}^\top \boldsymbol{r} = 0$ means that the point $\boldsymbol{r}$ belongs to the image line $\boldsymbol{l}_{ij}$, or equivalently that the corresponding 3D line intersects the ray $\mathcal{R}$ obtained by back projecting $\boldsymbol{r}$. The other case $\boldsymbol{d}^\top \boldsymbol{d}_{ij} = 0$ means that the 3D line with direction $\boldsymbol{d}_{ij}$ intersects the plane at infinity somewhere on the line $\boldsymbol{d}$; if we call $\mathcal{P}$ any plane with line at infinity $\boldsymbol{d}$, it follows that the 3D line with direction $\boldsymbol{d}_{ij}$ is parallel to $\mathcal{P}$.

"Rank 2" degeneracy

It is difficult to give a simple geometric characterisation of the "rank 2" degeneracy, but our experience is that this configuration does not follow a simple regular structure and is rather unlikely to happen in engineered patterns. For illustration purposes, an image of a sample "rank 2" degenerate configuration generated with Matlab® is shown in Fig. 3.3. The image was generated in the following manner. Given a rank-3 matrix H and a set of directions $\boldsymbol{d}_i$, an arbitrary rank 2 matrix H_2 was defined, and used to construct the set of lines $\boldsymbol{l}_i \sim (H_2 \boldsymbol{d}_i) \times$

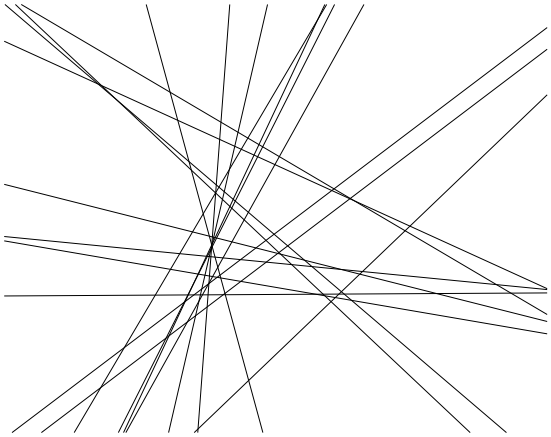


Figure 3.3: *Camera image of a sample "rank 2" degenerate configuration. Contrary to the "rank 1" case, there is no simple geometric pattern characterising the arrangement of the 3D lines and the pose of the camera.*

$(H\boldsymbol{d}_i)$. Such features satisfy $\boldsymbol{l}_i^\top H_2 \boldsymbol{d}_i = 0$, because $H_2\boldsymbol{d}_i$ and $\boldsymbol{l}_i$ are orthogonal by construction.

Analogy with degeneracies in camera resectioning

It is interesting to note that there exists an analogy between these degenerate configurations and the ones occurring in the case of camera resectioning. The degenerate configurations in the case of camera resectioning have been studied in [25] and [72] (chapter 21). In particular, it is shown there that the most important degenerate configurations arise when i) the points all lie on the union of a plane and a single straight line containing the camera centre, or ii) the camera and points all lie on a twisted cubic. It is straightforward to show that case i) is equivalent to our "rank 1" degenerate configuration, when pairs of points are used to form lines. Computer simulations have confirmed the hypothesis that case ii) corresponds to the "rank 2" degenerate configuration, however it remains to prove mathematically that they are strictly equivalent.

3.3.5 Constrained camera calibration

Until now, a general projective camera has been considered. In the case of restricted cameras, for example zero-skew, known aspect ratio, known principal point or known intrinsic parameters, the camera calibration is still possible by minimising either an algebraic or a geometric error. The minimisation of an algebraic error (which leads to a smaller minimisation problem)

can be done by defining a reduced measurement matrix as described in [70]. However it is not always possible to estimate a restricted camera matrix with a linear algorithm. For this reason, a geometric distance is usually preferred. The same geometric distance as in the case of a general camera can be considered, with the difference that only the camera parameters to be estimated must be included in the minimisation, while the other constraints are enforced.

3.4 Results

In this section, the decomposition method is evaluated with images of a calibration grid. Three different implementations are considered: linear method minimising the algebraic distance d_{alg} with or without normalisation of the input data, and non-linear method minimising d_{mah} (no normalisation required in this case). In the implementation, the linear methods use the SVD algorithm, while the non-linear method uses the LM algorithm. In the result graphs, the different methods are respectively labelled *decomposition (norm. linear)*, *decomposition (linear)* and *decomposition (non-linear)*. For comparison, the results of two additional camera calibration methods described below are included.

Camera calibration from point correspondences [72, 70]. The procedure consists of two stages. In the first stage, a linear solution is found by SVD; both scene and image points have been normalised by applying a translation and scaling to the input data prior to SVD, as recommended in [70]. In the second stage, a non-linear solution is found by non-linear minimisation (bundle adjustment). The LM algorithm is used at this stage; it is initialised with the results of the linear method. The whole camera calibration procedure is described in [72] (p 170) under the name of Gold Standard Algorithm. The result graphs corresponding to this method are labelled *standard*.

Camera calibration from VPs [33, 88]. VPs are computed from the intersection of parallel lines. Three sets of mutually orthogonal parallel lines define three VPs which can be used to compute the intrinsic parameters (assuming zero skew and a known aspect ratio) and the rotation. The parameters are computed linearly using SVD. Pre-normalisation is applied to the end points of the lines in order to guarantee good conditioning of the equation matrix [68]. Once these parameters have been computed, the translation is recovered by considering additional point

correspondences. The results graphs corresponding to this method are labelled *vanishing* or *vanishing (aspect ratio = 1)*. In the first case, the aspect ratio is obtained from the standard calibration method, while in the second case the aspect ratio is not precalibrated and assumed to be 1.

Experiments were performed with synthetic and real data. In both cases, a calibration grid made of two orthogonal planes containing control points was used. The control points provide the input data necessary for the standard calibration method. The input directions required for the decomposition method are obtained by considering pairs of control points. The set of parallel lines required for calibration from VP is obtained by least-square fitting of a line to each set of aligned control points; this defines three sets of parallel lines which are mutually orthogonal.

The general criterion of evaluation used is the Root Mean Squared (RMS) point reprojection error, which is defined by $\epsilon_{\text{rep}} = \sqrt{\frac{1}{N}\sum_i d(\boldsymbol{p}_i, \mathsf{K}[\mathsf{R}|\boldsymbol{t}]\boldsymbol{P}_i)^2}$. It measures how closely the control points mapped to the image by the estimated camera matrix match the noisy input data. In our implementation, the control points used for computing ϵ_{rep} are different from the ones used for camera calibration. The problem with using the same set of points for both calibration and evaluation is that it leads to biased results, because the RMS point reprojection error is a residual error in this case, which is not a good indicator of the quality of the solution obtained. For example, with exactly eight correspondences, the residual error is zero because there exists an exact transformation which matches the control points to the noisy image points; however this does not mean that the transformation estimated is accurate, on the contrary it is more likely to be inaccurate. If more correspondences are considered, the residual error will increase because it becomes more difficult to fit a model to the noisy data; this behaviour is contradictory to what is expected for the accuracy. More details on this topic can be found in [72] (chapter 4). However, the RMS reprojection error does not exhibit this behaviour when computed with different sets of points, because points used for the evaluation are independent from the ones used for estimation. This gives a more reliable measure of the accuracy of the calibration methods. In practice, the points used for the computation of the reprojection error are contained in a third plane orthogonal to the two other calibration planes used for calibration. It should be noted that the reprojection error thus defined is different from the cost function minimised by the non-linear decomposition method, for this reason the latter method

may not necessarily show a reduction in the error when compared with the linear decomposition method.

In the case of simulations with synthetic data, the ground truth values of the parameters are available, therefore it is possible to compute another criteria called RMS estimation error. It is defined by $\epsilon_{\text{est}} = \sqrt{\frac{1}{N}\sum_i (x_i - \overline{x}_i)^2}$, where $\overline{x}_i$ are the ground truth parameters, x_i are the estimated parameters, and N denotes the number of trials or repetitions of the experiment. This criterion measures how closely the estimated camera parameters match the original noise-free camera parameters. This measure is not available with real data because noise-free values are not accessible in that case.

3.4.1 Synthetic data

Each plane in the synthetic grid contains 100 control points. The camera to calibrate has the following parameters: $u_0 = 384\,\text{pixels}$, $v_0 = 247\,\text{pixels}$, $f = 714.3\,\text{pixels}$, $r = 1.167$, and $\theta = 90.018\,^\circ$.

Firstly, Gaussian noise was added to the spatial coordinates of the extracted image points, in order to study the robustness of the camera calibration method. The noise injected in both coordinates is independent. This is usually a reasonable assumption when image lines are defined by pairs of points. If lines were to be extracted directly from the image, a more complicated model of noise would be required. In this set of experiments, a single image is considered for calibration. The standard deviation of the noise injected in the image coordinates ranged from 0 to 1 pixel. For each noise level, the simulation was repeated 100 times, with a different seed used for the random number generator each time, so as to guarantee statistically meaningful results. Fig. 3.4 shows the RMS estimation error for each of the camera parameters, and Fig. 3.5 shows the RMS point reprojection error, in both cases with respect to the level of the noise injected in the image. It can be observed that the results of the decomposition method are very similar to the other two methods. It can be noticed that the normalised decomposition method performs better than the non-normalised method. In general, the non-linear method leads to good results, even if the improvement over the linear methods is negligible here. The results show that the decomposition method can accurately compute the parameters under noisy conditions. The method computing VPs is not as accurate because it uses only partial information:

only parallel direction can be used, and it is not able to estimate the skew and aspect ratio.

In a second set of experiments, the influence of the number of images used on the accuracy of the calibration is considered. For this purpose, a sequence of 10 images separated by a pure translation[1] motion of the camera, is generated. To relate the accuracy to the number of images considered, subsets of 1 to 8 images are randomly selected, and the camera calibration is performed with the images selected. The experiments are repeated 100 times for each size of the subset. Again, for each experiment, a different seed is used for the random number generator, in order to guarantee statistically meaningful results. Fig. 3.6 shows the RMS estimation error for each of the intrinsic parameters and Fig. 3.7 shows the RMS point reprojection error with respect to the number of images considered. The noise level of the Gaussian noise was set to $\sigma = 1$ pixel for all the experiments and no information about the translation was used. It can be observed that the RMS point reprojection error decreases rapidly for the different methods when the number of images increases, and it seems to converge to some asymptotic values. It seems that the normalised method leads to more accurate results than the unnormalised one, although the improvement is not very significant. The non-linear method seems accurate but it does not give the expected improvement in accuracy over the linear methods. The decomposition method appears to be slightly more accurate than the other methods when the number of images is increased.

3.4.2 Real data

A sequence of 20 images of a grid was produced with a Pulnix TMC-7DSP camera equipped with a 6 mm lens. The calibration grid was made of three planar grids, each containing 36 control points generated by a printer, which were positioned on three mutually orthogonal planes (see Fig. 3.9). The coordinates of the control points were verified with a measuring tape, the accuracy is estimated to be of the order of one millimetre. The coordinates of the images of the control points were extracted using the algorithm available in [21], which is based on the Harris corner detector [64]. The camera is mounted on a robot arm (see Fig. 3.8) that is used to generate a translation motion (up to the robot's accuracy). The images obtained present some

[1]Note that in the case of real experiments, the translation motion will never be perfect, and therefore lower accuracy is expected.

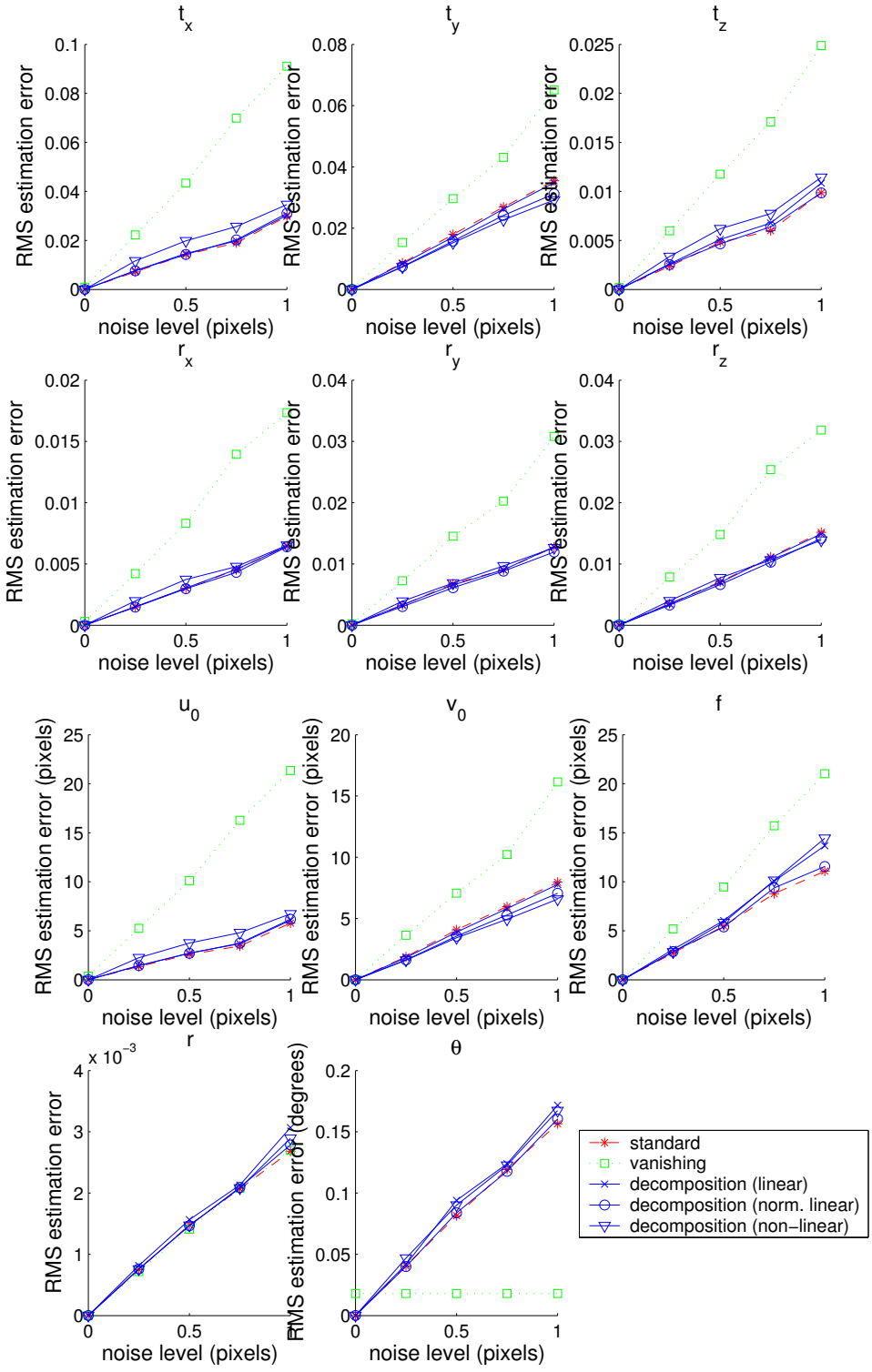


Figure 3.4: *RMS estimation error for each camera parameter with respect to the noise level in the case of synthetic experiments of camera calibration from a single image. The RMS error is computed across 100 trials.*

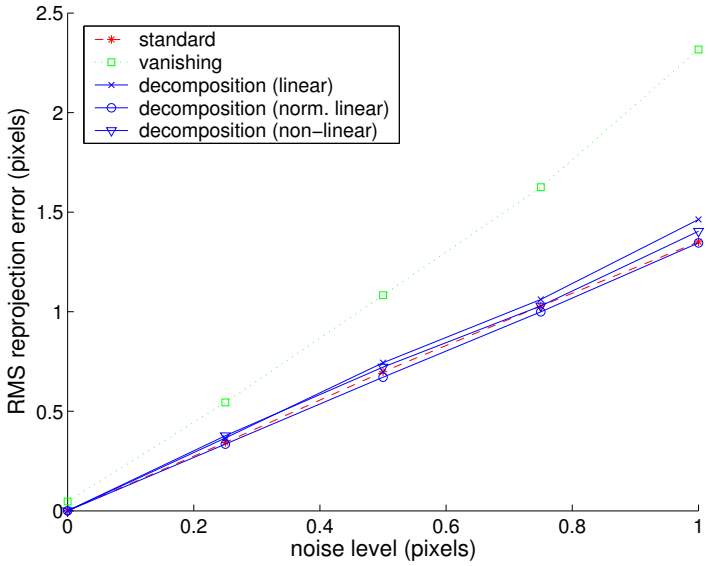


Figure 3.5: *RMS point reprojection error with respect to the noise level in the case of synthetic experiments of camera calibration from a single image. The results were obtained from 100 experiments.*

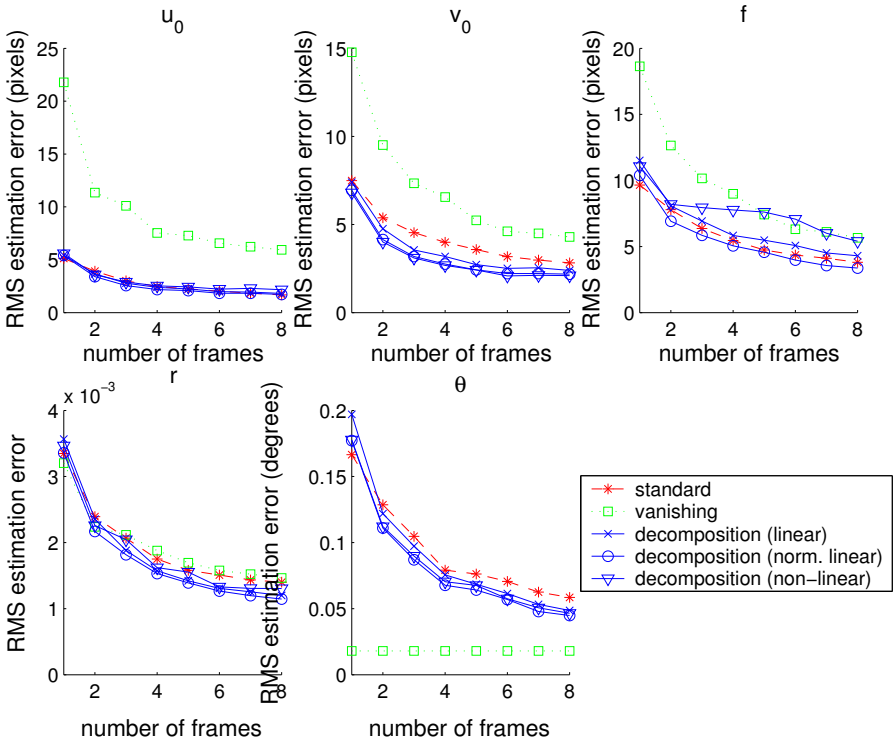


Figure 3.6: *RMS estimation error for each intrinsic camera parameter with respect to the number of images considered in the case of synthetic experiments of camera calibration with a translating camera. The noise level was fixed to $\sigma = 1$ pixel. The results were obtained from 100 experiments.*

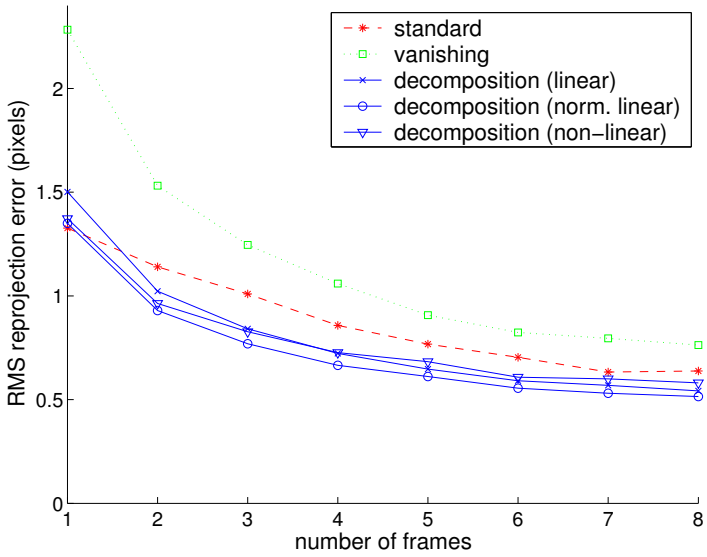


Figure 3.7: *RMS point reprojection error with respect to the number of images considered in the case of synthetic experiments of camera calibration with a translating camera. The noise level was fixed to* $\sigma = 1$ *pixel. The results were obtained from 100 experiments.*

Figure 3.8: *Camera mounted on the robot arm used to generate the translation motion.*

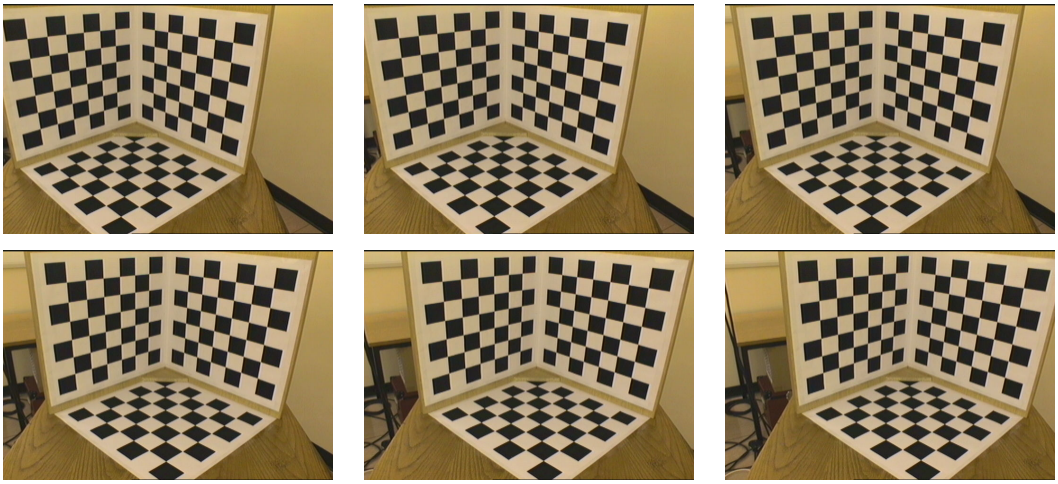


Figure 3.9: *Real images used for calibration. The images are obtained by translating the camera mounted on a robot arm.*

Table 3.1: *Estimated intrinsic parameters in the case of real experiments of camera calibration with a translating camera. The results are given in the case of eight input images. For each method the mean value and the standard deviation were obtained from 100 experiments.*

		u_0 (pixels)	v_0 (pixels)	f (pixels)	r	θ (degrees)
standard	mean	327.4	257.1	695.2	1.107	90.17
	std	0.146	0.192	0.619	0.237×10^{-3}	8.98×10^{-3}
vanishing	mean	329.5	257.5	695.4	1.107	90
	std	0.703	0.181	0.669	0.256×10^{-3}	0
vanishing	mean	321.0	325.4	676.9	1.000	90
(aspect ratio = 1)	std	0.558	0.243	0.676	0	0
decomposition	mean	326.7	258.6	696.0	1.107	90.18
(linear)	std	0.203	0.513	0.444	0.251×10^{-3}	6.06×10^{-3}
decomposition	mean	326.6	257.7	695.2	1.107	90.19
(norm. linear)	std	0.194	0.400	0.459	0.233×10^{-3}	5.89×10^{-3}
decomposition	mean	326.6	257.7	695.2	1.107	90.19
(non-linear)	std	0.194	0.400	0.458	0.231×10^{-3}	5.92×10^{-3}

lens distortion. In this case, only the radial distortion is corrected, and a first-order coefficient appears to be sufficient [155]. Here the lens distortion can be appropriately calibrated from the image of lines, by requiring them to be straight. This technique is known as the plumb-line method in the photogrammetry literature [127]; an implementation for computer vision is presented in [40]. In the case of calibration from multiple images separated by translation of the camera, random subsets of 1 to 8 images were selected. We made 100 trials for each subset size. Table 3.1 shows the mean and the standard deviation of the parameter values obtained for the different methods considered, in the case where eight images were considered. Fig. 3.10 shows the RMS point reprojection error with respect to the number of images considered.

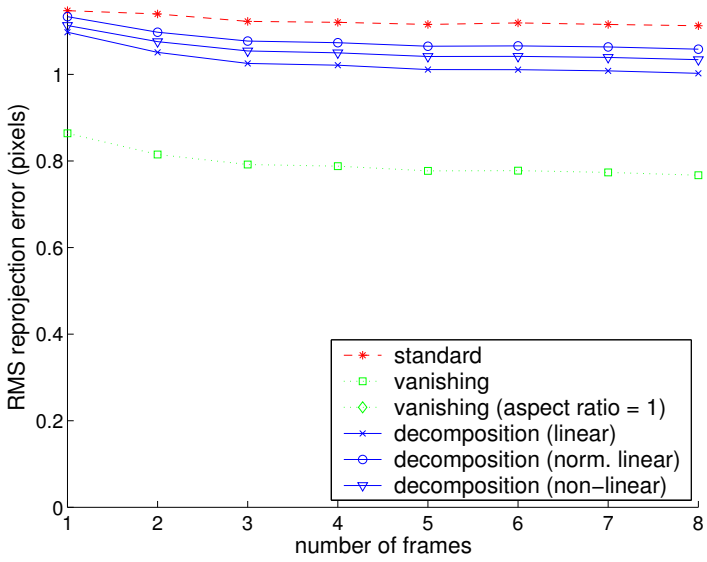


Figure 3.10: *RMS point reprojection error with respect to the number of images considered in the case of real experiments of camera calibration with a translating camera. The results were obtained from 100 experiments. The graph corresponding to the VP method with an aspect ratio assumed to be 1 leads to large RMS point reprojection error of the order of 15 which are not visible in the figure.*

It can be seen that the accuracy of the methods increases with the number of images. The decomposition method and the method computing VPs (with the aspect ratio obtained from the standard method) perform better than the method using point correspondences. With the method using point correspondences, the size of the parameter space increases each time a new image is included. For example, if n images are considered, then there are $8+3n$ parameters to estimate (5 intrinsic plus 3 for the orientation, and 3 for the position of the camera corresponding to each image). When the size of the parameter space increases, the risk of being trapped in a local minimum increases and it becomes less likely to converge to the global minimum. In comparison, with both the decomposition method and the method computing VPs, the size of the parameter space remains fixed for the first stage (8 parameters). Similarly to the synthetic data analysis, the linear method with normalisation is more stable than the unnormalised solution. In the particular case where the method using VPs is provided with the aspect ratio obtained from the standard method, it gives more accurate results than the other methods. However, the method computing VPs has some limitations: i) it assumes the camera has zero skew, ii) it requires to provide a value for the aspect ratio. Some inaccuracies are expected if the VP method is provided with an incorrect value for the aspect ratio. For example, it can be

seen in Fig. 3.10 that the method performs poorly when initialised with an aspect ratio of one, which is however a reasonable value for most CCD cameras. It is possible to provide a better value for the aspect ratio, *e.g.* by using the non-linear standard camera calibration method, but this requires to run another preliminary calibration algorithm.

Overall if we compare the results of synthetic and real data, it can be seen that the decomposition method usually performs better than the other methods when multiple images are used. It can be noticed that in some cases the non-linear decomposition method is actually less accurate than the linear one. From a theoretical point of view, there is no reason for the non-linear method to be more accurate since the cost functions used are different (see introduction of this section). Practically, the results observed are very close and the discrepancy is not significant; they could be due to the approximations performed when formulating the non-linear cost function, to different noise conditions in synthetic and real experiments, or to some inaccuracies in the translation motion in the case of real experiments.

3.5 Conclusions

It has been observed that it is possible to decouple the translation parameters from the other parameters during camera calibration. The main idea is that directions in the scene are represented by PI, which project to VPs, and can be used to derive equations that are independent of translation. This property has been used to formulate a novel camera calibration method. Its originality consists in replacing the strict constraint that a PI maps to a VP, by a softer constraint that establishes that a VP should lie on the corresponding image line. This new formulation presents some advantages in terms of flexibility compared to standard VP methods, because it is not required to have sets of parallel lines present in the scene.

The main advantage of the decomposition method remains its ability to split the parameter space into two smaller sets of parameters; the first set contains the intrinsic and position parameters, while the second one contains only the translation parameters. This presents some advantages in terms of accuracy over more conventional methods such as the Gold Standard algorithm, which solves for all the parameters simultaneously. The advantage becomes all the more significant when a translation motion is used to generate more data from the scene, as

this does not imply any increase in the dimensionality of the problem, when the decomposition method is considered. This however requires an apparatus able to generate an accurate translation motion. Deviations from the expected translation motion may affect the accuracy of calibration.

Experimental results showed that the accuracy of the decomposition method is comparable to other methods when a single image is considered. If several images separated by a translation motion are considered, the decomposition method performs often better than the standard method or the method computing VPs.

Chapter 4

The Normalised Image of the Absolute Conic (NIAC) and its use for zooming camera calibration

4.1 Introduction

In this chapter, the study of invariants in camera calibration is continued, defining and exploiting a novel invariant in the case of a zooming camera moving freely in 3D space. The ability to zoom is of considerable interest in computer vision, as it enables to focus on selected parts of the scene which present higher interest, however this also requires more complex calibration techniques. In the case of motorised zoom lenses, the relationship between the lens control parameters and the camera parameters can be determined from the results of calibration at a series of sampled lens settings. For example, in [142], the parameter values estimated at the sampled positions are stored in a look-up table, from which parameters corresponding to new settings can be derived by interpolation. Note that the principle is similar to the direct calibration method used by Trucco *et al.* in the case of triangulation-based range sensors using structured laser light [153]. [29] showed that this calibration process can be speeded up by replacing the previous algorithm by an adaptive algorithm, which selects automatically which values should be included in the look-up table based on the accuracy. Willson produced a more compact representation by fitting a polynomial at the sampled values for each parameter [166].

The latter technique has been reported to be accurate, however it imposes smooth variations on the parameters, which may be too restrictive in certain cases. It has been shown in [5] that a more general algorithm can be obtained by using neural networks.

In the previous approaches, parameter values at new settings are inferred from the values at a sample of lens settings. For accurate results, the procedures require a dense sampling over the range of all possible settings, which is a demanding task. In addition, even though a dense sampling is carried out, there is no guarantee of obtaining accurate results if there exists some discontinuities in the parameter variations. Finally, these techniques require the use of motorised lenses with indexed position settings, which is not the case for all camera technologies. Self-calibration methods relax all these assumptions by offering the possibility to calibrate the camera directly from the same images which are used for the vision task. The concept of self-calibration was first introduced in [49] by Faugeras *et al.* in the case of cameras with fixed lens settings, and then generalised to zooming cameras by Pollefeys *et al.* in [111]. The approach is very attractive, however there exists a number of critical motion sequences for which the solution is ambiguous [135, 137]. An example of a degenerate configuration which occurs frequently in practice is the case of a rotating camera. In this case it is not possible to resolve depth because of the absence of parallax. For this reason, specific algorithms for rotating and zooming camera have been developed, for example in [2, 122]. Other methods have tried to simplify the self-calibration task by reducing the number of parameters to estimate. For example, Sturm has shown in [136] that pre-calibration can be used to model the interdependency between the zooming camera parameters, which reduces self-calibration to the estimation of a single parameter. Generally self-calibration techniques rely on sufficient and accurate point correspondences, and require good initial values. Convergence problems and noise usually limit the accuracy of such techniques (see [20]).

One of the main reasons why the calibration of a zooming camera is difficult is that it increases significantly the number of parameters to estimate, in particular when multiple images are used for calibration. Previous works have taken advantage of invariants to decouple the camera parameters into simpler sub-problems and guarantee that the number of unknowns of each sub-problem is constant. Some examples of invariants used in camera calibration include Vanishing Points (VPs), which are invariant to translation [26, 159, 43, 28, 160, 12, 33, 88], or the Image of the Absolute Conic (IAC), which is invariant to changes in position and orientation [89, 88,

175, 138, 96, 95, 62, 61] (see Chapter 3 for more examples of invariants). In this chapter, the invariance properties of the IAC are extended to include invariance to zooming, by defining a novel invariant called the *Normalised Image of the Absolute Conic (NIAC)*. It is shown that the camera parameters independent of the position, orientation and zooming are determined uniquely by the NIAC. The invariance properties are used to define a stratified method requiring only three or four views (depending on the camera model) of a square grid in arbitrary positions. This enables the calibration to be decoupled into three sub-problems:

1. Estimation of intrinsic parameters independent of the zoom (computed through the NIAC),

2. Estimation of focal length representing the zoom for each image (computed through the IAC for each image)

3. Estimation of extrinsic parameters for each image.

In general, zooming imposes a large-scale non-linear minimisation which is usually unstable and less likely to converge to the solution. This is not the case with our method for which each of the sub-problems has small dimension, and can therefore be solve more efficiently and accurately.

In comparison with other plane-based calibration techniques for zooming cameras [138, 62, 61], our method has the advantage of increased generality (it is not restricted to zero-skew cameras as in [138, 62, 61]) and also better accuracy when compared to [138] which computes simultaneously all the intrinsic parameters. Under the NIAC framework, the method presented in [62, 61] is actually a special case of our algorithm for zero-skew cameras. Our method is the only one which minimises an exact geometric distance. [138] minimises an algebraic distance, which requires careful normalisation of the data, while [62, 61] defines only an approximation of a geometric distance. In order to accommodate all types of cameras, such as zero-skew or non-zero skew cameras, several implementations are proposed. We start by presenting the zooming model adopted in this chapter, giving a theoretical and experimental justification for it. Then the novel invariant is introduced. The following section shows how it can be used for calibrating a zooming camera. Finally some results with synthetic and real data are given.

4.2 Zooming camera model

The zooming camera model described in Section 2.2.3 is adopted in this chapter. With this model, zooming is equivalent to varying only the focal length of the camera. It has been shown by Willson in [165, 163, 166, 164] that this is not the physically most accurate model, mainly because the principal point can exhibit significant changes in position, and also because the optical centre can move along the Z axis while zooming. However, there are several motivations for considering a fixed principal point. Firstly, it simplifies considerably the camera calibration process. For example, under this assumption, it is possible to calibrate a zooming camera from three images of a planar grid taken with arbitrary positions and orientations. Generally, a model with varying principal point would require to observe at least two planar grids simultaneously in order to define a sufficient number of constraints on the camera parameters. Grids consisting of several planes are more complicated, and they are difficult to position in the scene if we would like them to be observable from all camera viewpoints - especially in the case of a camera moving freely in the 3D space. Second, even though this model is not the most accurate for the computation of each individual parameter, it turns out that it is sufficiently accurate for the computation of the overall projection matrix because the error made by considering that the principal point is fixed is compensated by an error in the position of the camera centre. For most applications in computer vision, it is sufficient to calibrate only the projection matrix, without accessing each single parameter, and such a model is sufficiently accurate. Theoretical and experimental results supporting this model are given in this section.

4.2.1 Theoretical justification

Zooming is obviously independent of the position and orientation of the camera. Therefore, we can make the simplifying assumption that the camera is located at the origin of the world frame and pointing along the Z axis, without loss of generality. Under this assumption, the camera projection matrix is given by:

$$M_{f,u_0,v_0} = \begin{bmatrix} f & -f\cot\theta & u_0 & 0 \\ & fr/\sin\theta & v_0 & 0 \\ & & 1 & 0 \end{bmatrix}.$$

In a zooming camera model with varying principal point, zooming induces a variation Δf, Δu_0 and Δv_0 in the values of the focal length and the coordinates of the principal point. The projection matrix becomes

$$\mathsf{M}_{f+\Delta f,u_0+\Delta u_0,v_0+\Delta v_0} = \begin{bmatrix} f+\Delta f & -(f+\Delta f)\cot\theta & u_0+\Delta u_0 & 0 \\ & (f+\Delta f)r/\sin\theta & v_0+\Delta v_0 & 0 \\ & & 1 & 0 \end{bmatrix},$$

and a point $\boldsymbol{P} = (X, Y, Z, 1)^\top$ is projected into the image point:

$$\boldsymbol{p} = \mathsf{M}_{f+\Delta f,u_0+\Delta u_0,v_0+\Delta v_0}\boldsymbol{P} = \mathsf{M}_{f+\Delta f,u_0,v_0}\boldsymbol{P} + \begin{bmatrix} Z\Delta u_0 \\ Z\Delta v_0 \\ 0 \end{bmatrix}.$$

If we now consider a alternative zooming model where the principal point is fixed, but the camera centre is translated by the vector $[\Delta t_X, \Delta t_Y, 0]^\top$, then the same point $\boldsymbol{P}$ projects into the image point

$$\boldsymbol{p}' = \mathsf{M}_{f+\Delta f,u_0,v_0}\left(\boldsymbol{P} - \begin{bmatrix} \Delta t_X \\ \Delta t_Y \\ 0 \end{bmatrix}\right) = \mathsf{M}_{f+\Delta f,u_0,v_0}\boldsymbol{P} - \begin{bmatrix} (f+\Delta f)(\Delta t_X - \cot\theta\Delta t_Y) \\ \frac{(f+\Delta f)r}{\sin\theta}\Delta t_Y \\ 0 \end{bmatrix}.$$

It can be observed that the two models are equivalent if and only if

$$\begin{cases} Z\Delta u_0 &= -(f+\Delta f)(\Delta t_X - \cot\theta\Delta t_Y), \\ Z\Delta v_0 &= -\frac{(f+\Delta f)r}{\sin\theta}\Delta t_Y, \end{cases}$$

which requires that

$$\begin{cases} \Delta t_X &= -\frac{Z}{f+\Delta f}(\Delta u_0 + \frac{\cos\theta}{r}\Delta v_0), \\ \Delta t_Y &= -\frac{Z\sin\theta}{(f+\Delta f)r}\Delta v_0. \end{cases}$$

If all scene points are located in a plane parallel to the image plane, then Z is the same for all scene points and the two equations above can be satisfied simultaneously for all scene points, *i.e.* the two models are strictly equivalent. If this is not the case, but the depth relief is small with respect to the average depth, Z can be replaced by the average depth value, and the two models are still equivalent up to a first order approximation.

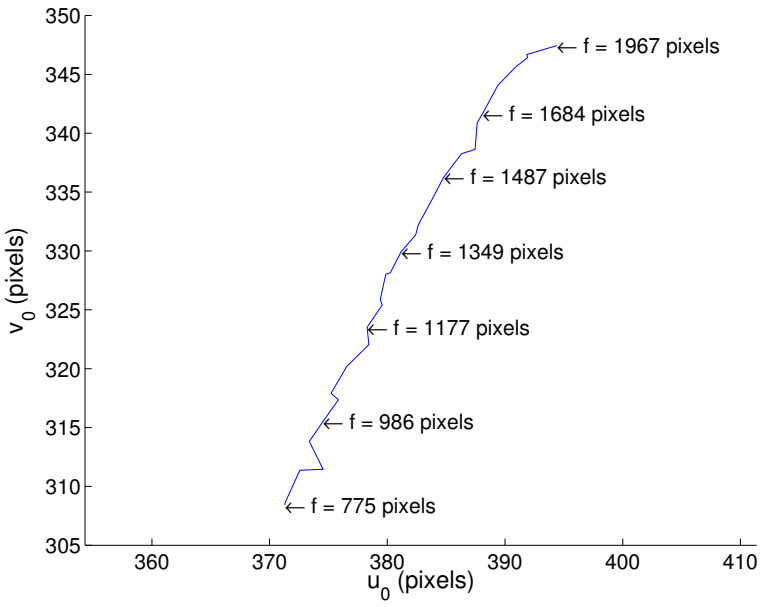


Figure 4.1: *Displacement of the principal point while the camera is zooming. The curve is obtained from 36 zoom settings; some estimated focal lengths have been annotated on the graph for information.*

4.2.2 Experimental validation

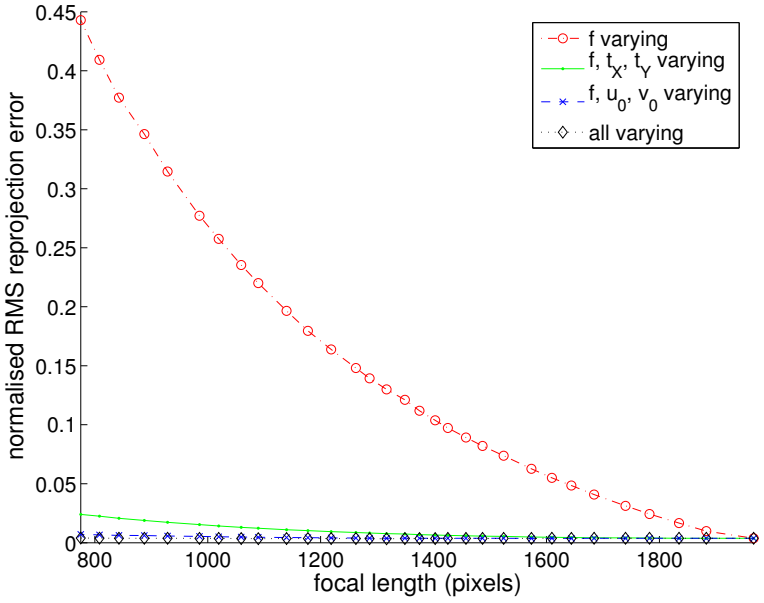
Some experiments were carried out in order to evaluate the accuracy of the chosen model. The camera used is a Sony DXC-9100P equipped with a Fujinon S12×5BRM-38 zooming lens which has a 5–60 mm focal length range. This is a progressive scan camera; the resolution of the images produced is 720×576 pixels. The camera is mounted on a tripod, and its pose and orientation are kept constant during the experiment, so that variations in the parameters are due only to zooming. The camera is pointing at a calibration grid made of two orthogonal square grids of size 420 mm. The calibration grid is located approximately 2500 mm away from the camera. A collection of images is acquired for different zoom setting.

The position of the principal point has been determined for each zoom setting by calibrating the camera using the Gold Standard algorithm described in [72]. It can be observed in Fig. 4.1 that the principal point describes an approximately linear motion in the image while the camera is zooming. The amplitude of the movement is about 25 pixels along the horizontal axis and more than 40 pixels along the vertical axis.

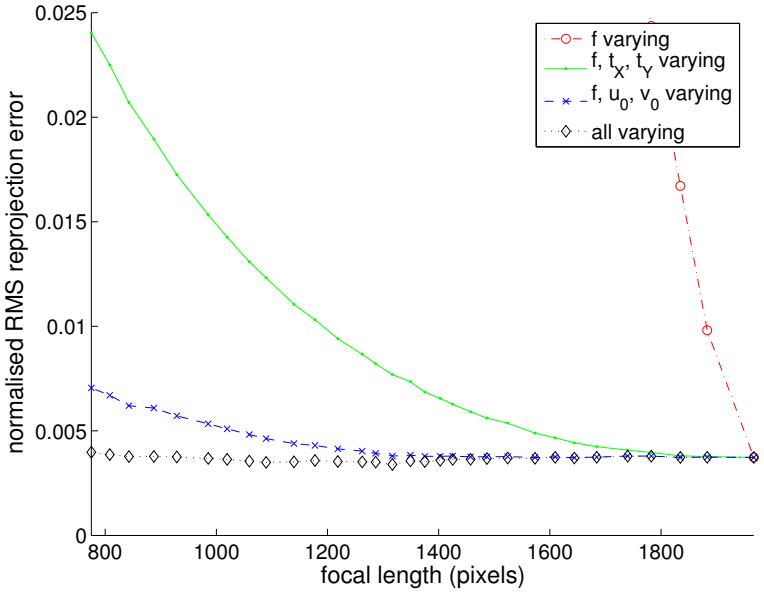
The experiment carried out by Willson in [165] has been repeated, including the model which compensates the motion of the principal point by a motion of the camera centre, which had not

been considered in [165]. Briefly, the experiment starts by calibrating all the camera parameters for the largest zoom setting, then the zoom factor is reduced and the camera recalibrated for each new setting, allowing only a given number of camera parameters to vary; this number is dictated by the choice of the zoom model. For example, the simplest model consists in allowing only the focal length to vary, while a more elaborate model includes also the two coordinates of the principal point. For each model, the normalised Root Mean Squared (RMS) reprojection error is computed and gives a measure of the accuracy of the model. The normalisation consists in scaling the absolute RMS reprojection error values by the inverse of the mean radius of the cloud of image points, for each image. Such a normalisation is necessary if we want to eliminate the influence of the scale of the grid which varies significantly due to zooming. For information, the mean radius of the cloud of image points is 60 pixels for the smallest zoom factor and 168 pixels for the largest zoom factor. The minimum error obtainable is given by performing a full calibration independently for each setting. In all cases, the calibration is done using the Gold Standard algorithm described in [72]. The linear method is used to initialise the varying zoom parameters, which are then refined by non-linear optimisation.

The results are shown in Fig. 4.2. It can be observed that the minimum error remains approximately constant over the range of zoom variation. The model allowing only the focal length to vary is not surprisingly the least accurate; it results in an increase in the reprojection error by a factor of 120 over the range of zoom variation. The model which allows the principal point to vary is the most accurate, it is able to capture most of the variations exhibited by this camera, resulting in an error increase by only a factor of 2. With the third model where the principal point is fixed, but the position of the camera centre is allowed to vary with the focal length, the error increases by only a factor of 6 over the whole range of zoom variation. This is 20 times more accurate than the model with only f varying, and is a fairly close approximation of the camera model with varying principal point. These results confirm that the motion of the principal points tends to be compensated by the motion of the optical centre. From a practical point of view this means that even though there may be a significant error in the estimation of the parameters when assuming a fixed principal point, the estimation of the overall camera projection matrix can be done much more accurately (20 times with this camera) because errors are compensated. This is still not as accurate as considering a variable principal point, however it is usually accurate enough for most applications. Also, it must be considered



(a)



(b)

Figure 4.2: *Comparison of the accuracy of the different zooming camera models. The bottom graph is a magnification of the top graph.*

that the loss in accuracy due to having a simpler model is balanced by a gain in flexibility of the calibration method (possibility to use simpler calibration targets because fewer parameters must be estimated) and also the possibility to define simpler invariants. The possibility to use invariants may ultimately translate into an improvement in accuracy when multiple images are used, which may not be observed with a more complex camera model. In some cases, the error induced by the fixed principal point assumption may still be too large, even though only the camera matrix is computed. In such cases, it is necessary to compute the coordinates of the principal point for each image, in addition to the other camera parameters. This does not mean that simpler calibration techniques assuming a fixed principal point are not useful. These can for example be used to provide an initial estimate of the camera parameters which can be then refined and extended to a more general camera model including varying principal point by bundle-adjustment.

4.3 A novel invariant: the NIAC

This section defines a novel geometric entity called the Normalised Image of the Absolute Conic (NIAC) which encapsulates the camera parameters invariant to zooming.

4.3.1 Invariance properties of the IAC

For a given focal length f, the IAC is defined by the conic coefficient matrix $\omega_f = (\mathsf{K}_f \mathsf{K}_f^\top)^{-1}$ or equivalently by the following equation (see Appendix C):

$$(u-u_0)^2 + \frac{1}{r^2}(v-v_0)^2 + 2\frac{cos\theta}{r}(u-u_0)(v-v_0) = -f^2\,. \tag{4.1}$$

It appears immediately that any given IAC is centred at the principal point and that f is related to only the scale of the IAC. Under the model defined previously, zooming therefore produces a one-parameter family of IAC which can be parameterised by the focal length f. The effect of varying f is illustrated in Fig. 4.3. The set of IAC obtained is *homothetic* (curves are related by an expansion or geometric contraction) and concentric, with centre the principal point (u_0, v_0) of the camera.

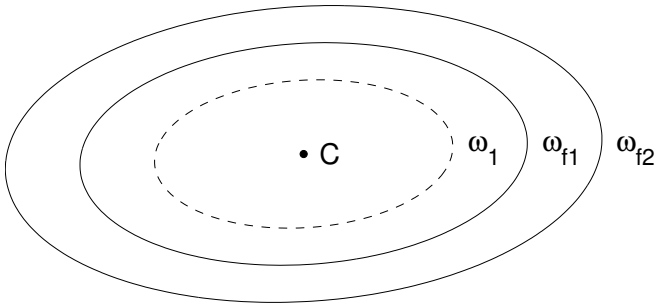


Figure 4.3: *Illustration of the transformation of the IAC while the camera is zooming. The different IAC ω_{f_i} are all centred in the principal point C and homothetic. It is possible to choose one of them, for example the one with focal length 1 as a reference, that we call NIAC ω_1.*

Table 4.1: *A hierarchy of invariants and their properties. VPs are invariant to translation. The IAC extends the invariance properties to rotation. Ultimately, the NIAC adds invariance to zooming.*

invariant	motion
NIAC	translation, rotation, zoom
IAC	translation, rotation
VPs	translation

4.3.2 The NIAC

We define the Normalised Image of the Absolute Conic (NIAC) as *the IAC corresponding to a focal length of* 1. The NIAC is an imaginary conic represented by the symmetric matrix $\omega_1 = (K_1 K_1^\top)^{-1}$. By construction ω_1 is invariant to the position, orientation and change in the focal length of the camera. It has four degrees of freedom, corresponding to the coordinates of the principal point (u_0, v_0), the aspect ratio r, and the angle between the two axis of the camera θ.

In terms of invariant, the NIAC can be considered as the natural extension of the IAC to zooming cameras. In the hierarchy of invariants, at the bottom we have the VPs which are invariant to translation, then the IAC which extends the invariance properties by adding rotation, and finally the NIAC which adds the zoom invariance (see Table 4.1). Because the NIAC encapsulates all the intrinsic parameters invariant to zooming, calibrating these parameters is equivalent to estimating ω_1. Once ω_1 is known, K_1 and therefore the intrinsic parameters invariant to zooming can be recovered from Cholesky factorisation [112].

4.4 Application to camera calibration

Before describing the novel camera calibration method, a brief reminder of the principle of plane-based camera calibration using the IAC is given. The main idea is to replace the computation of the calibration matrix K representing the intrinsic parameters, by the estimation of the IAC. The absolute conic being an imaginary object, it is a priori not directly observable, however it has been shown in [175, 138] that it is possible to compute the image of two remarkable points belonging to it from the observation of any planar calibration target. These two points are called *circular points*, and we give a summary of their computation below.

4.4.1 Computation of the circular points

Let us suppose that the camera is pointing at a planar calibration target. By definition, the circular points of this plane are the two points of intersection with the absolute conic. For simplicity and without loss of generality, it is assumed that the calibration plane is located in the plane $Z = 0$, in which case the points of intersection with absolute conic are the two points $\boldsymbol{I} = [1, \mathrm{i}, 0]^\top$ and $\boldsymbol{J} = [1, -\mathrm{i}, 0]^\top$. Because the plane is marked with known control points, it is also possible to compute the homography H between the calibration plane and its image, from which we can derive that the images of the two circular points are: $\boldsymbol{P} = H\boldsymbol{I} = \boldsymbol{h}_1 + \mathrm{i}\boldsymbol{h}_2$ and $\boldsymbol{Q} = H\boldsymbol{J} = \boldsymbol{h}_1 - \mathrm{i}\boldsymbol{h}_2$. Both points lie on the IAC. In the case of a camera with constant intrinsic parameters, each image of a calibration plane provides two such points on the IAC. A general conic is defined uniquely by five points. Therefore it is sufficient to make three plane observations (two in the case of a zero skew camera because of the additional zero-skew constraint) in order to obtain a sufficient number of constraints and determine uniquely the IAC, and therefore K.

In the case of a zooming camera, it is necessary to consider a more general invariant defined for example by the NIAC, which is invariant to translation, rotation and zoom. The calibration algorithm can be broken into three stages. In the first stage, the invariant intrinsic parameters encapsulated in the NIAC are computed; such parameters are the coordinates of the principal point, the aspect ratio and the skew parameter. This is the most complicated stage of the method. The next stages concentrate, separately for each image, first on the computation of

the focal length, which represents the zooming effect, then on the computation of the extrinsic parameters, *i.e.* position and orientation.

4.4.2 Computation of K_1

The matrix K_1 represents the intrinsic parameters of the camera which are invariant to a change in position, orientation and zooming. These parameters are characterised uniquely by the NIAC $\omega_1 = (K_1 K_1^\top)^{-1}$. Like the IAC, the NIAC is an imaginary conic, it is therefore not directly observable, and a special construction is needed. As for the IAC, the information is provided by the observation of a sufficient number of calibration planes, which provide a set of pairs of images of the circular points. However, this time there exists as many different IAC as there are pairs of images of circular points, therefore a more elaborate strategy is needed.

We start by observing that, if the parameters from K_1 are known, it is possible to define a normalised image reference frame in which the NIAC is a unit circle centred at the origin. In this normalised image reference frame, the camera has effectively a unit aspect ratio, zero skew, and its principal point is at the origin. Such a reference frame is obtained by applying an image transformation T which is composed of a shear transformation along the X axis (to eliminate the skew), a scaling along the Y axis (to correct the aspect ratio), and a translation (to map the principal point to the origin). The transformation obtained is parametrised by four parameters t_1,t_2,t_3 and t_4 ($t_3 \neq 0$):

$$T = \begin{bmatrix} 1 & 0 & t_2 \\ & 1 & t_4 \\ & & 1 \end{bmatrix} \begin{bmatrix} 1 & & \\ & t_3 & \\ & & 1 \end{bmatrix} \begin{bmatrix} 1 & t_1 & 0 \\ & 1 & 0 \\ & & 1 \end{bmatrix} = \begin{bmatrix} 1 & t_1 & t_2 \\ & t_3 & t_4 \\ & & 1 \end{bmatrix} . \tag{4.2}$$

The main idea of the method is that calibration can be reformulated in terms of identifying the unique transformation T which maps the NIAC to a unit circle centred at the origin. Because all IAC are homothetic and concentric, T maps the set of IAC into a set of concentric circles centred at the origin. We show that such a configuration can be characterised uniquely by the perpendicular bisectors to the chords defined by the pairs of images of circular points on the IAC. The result is stated below. The concept is illustrated in Fig. 4.4.

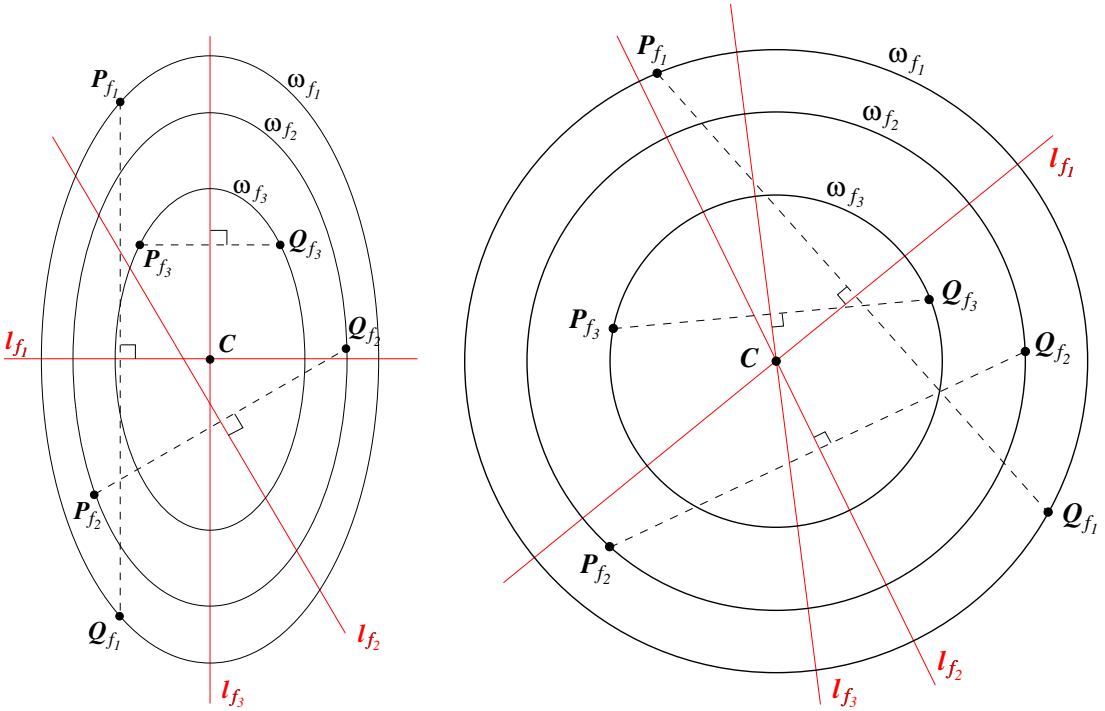


Figure 4.4: *ω_{f_1}, ω_{f_2} and ω_{f_3} are three concentric homothetic conics centred at C. On each conic ω_{f_i}, the points P_{f_i} and Q_{f_i} represent the images of the circular points. They define a chord on each conic. We assume that none of the chords passes through C and that no two chords are parallel. The perpendicular bisectors to the chords are represented by the lines l_{f_1}, l_{f_2} and l_{f_3}. In the general case where the conics are non-circular (a), the perpendicular bisectors pass through the centre C if and only if the chord is parallel to an axis of the conic (*e.g. *l_{f_1} and l_{f_3}*). Because there exists only two axis, l_{f_1}, l_{f_2} and l_{f_3} cannot be concurrent at C. The only case where l_{f_1}, l_{f_2} and l_{f_3} are concurrent at C is when the conics are circular (b).*

Result 1 *Consider n ($n \geq 3$) concentric homothetic conics centred at C. Take one chord on each conic such that no chord passes through C, and no two chords are parallel. The perpendicular bisectors to the chords are concurrent*[1] *in C if and only if the conics are circular.*

Proof If the conics are circles, it is clear that the perpendicular bisector to each chord is a diameter, and therefore that the set of all perpendicular bisectors are concurrent at the common centre C of the set of circles. Reciprocally, let us assume that the conics are not circles. Then each chord must be parallel to the diameter conjugate to the perpendicular bisector (see [120],

[1] Three of more lines are concurrent if they meet at one point.

p 120). However, because they are orthogonal, the perpendicular bisector and its conjugate diameter define the two axes of the conic. It results that the chords are all parallel to one of the axes of the conic on which they lie. The conics being concentric and homothetic, they share the same axes. Also, because none of the chords are parallel, this defines $n \geq 3$ distinct axes, which is impossible because there can be only a pair of axes. We deduce that the conics are circular. This completes the proof of Result 1. □

Given a pair of images of circular points $\boldsymbol{P} = \boldsymbol{HI} = \boldsymbol{h}_1 + \mathrm{i}\boldsymbol{h}_2$ and $\boldsymbol{Q} = \boldsymbol{HJ} = \boldsymbol{h}_1 - \mathrm{i}\boldsymbol{h}_2$, with $\boldsymbol{h}_1 = [h_{11}, h_{21}, h_{31}]^\top$ and $\boldsymbol{h}_2 = [h_{12}, h_{22}, h_{32}]^\top$, it can be shown that after mapping by T, the equation of the perpendicular bisector is:

$$\boldsymbol{l} = [-(d_1 + t_1 d_2), -t_3 d_2, (m_1 + t_1 m_2 + t_2)(d_1 + t_1 d_2) + (t_3 m_2 + t_4) t_3 d_2]^\top , \qquad (4.3)$$

with

$$\begin{cases} m_1 &= \frac{1}{h_{31}^2 + h_{32}^2}(h_{31}h_{11} + h_{32}h_{12}) , \\ m_2 &= \frac{1}{h_{31}^2 + h_{32}^2}(h_{31}h_{21} + h_{32}h_{22}) , \\ d_1 &= h_{32}h_{11} - h_{31}h_{12} , \\ d_2 &= h_{32}h_{21} - h_{31}h_{22} . \end{cases} \qquad (4.4)$$

The derivation is given in Appendix D. It follows that calibrating K_1 is equivalent to finding the unique values of the parameters t_1,t_2,t_3 and t_4 for which the perpendicular bissectors $\boldsymbol{l}$ are concurrent at the origin. Once this transformation has been estimated, the NIAC is given by $\omega_1 = \mathsf{T}^\top \mathsf{T}$ and the intrinsic parameters by $\mathsf{K}_1 = \mathsf{T}^{-1}$. A number of algorithms for estimating these parameters are presented below.

Non-linear solution minimising a geometric distance

The first method proposed consists in finding the solution which minimises the sum of squared distances d_{geom} between the line l defined in Eq. (4.3) and the origin for each image:

$$d^2_{\text{geom}} = \frac{[(m_1 + t_1 m_2 + t_2)(d_1 + t_1 d_2) + (t_3 m_2 + t_4) t_3 d_2]^2}{(d_1 + t_1 d_2)^2 + (t_3 d_2)^2} . \qquad (4.5)$$

In the case of a zero-skew camera, we have $t_1 = 0$, and the previous expression simplifies to

$$d^2_{\text{geom}} = \frac{[(m_1 + t_2) d_1 + (t_3 m_2 + t_4) t_3 d_2]^2}{d_1^2 + (t_3 d_2)^2} . \qquad (4.6)$$

A minimum of four images is required to determine uniquely the fixed intrinsic parameters in the case of a general camera. With a zero-skew camera, three images are sufficient, because t_1 is already known to be zero. Minimising such cost functions requires non-linear techniques such as the Levenberg-Marquardt algorithm. Given the very small number of unknowns, the method usually requires very few iterations before converging. Also, it can cope with poor accuracy initialisations. Experiments showed good convergence properties, however it is *a priori* not guaranteed that there exist no local minima in the vicinity of the solution which may affect the convergence of the algorithm to the correct solution. In practice, a reasonable initialisation which gives good results is to choose the principal point at the image centre, an aspect ratio of one and zero-skew. Alternatively, the method defined in the next paragraph can be used for initialisation.

Linear solution minimising an algebraic distance

Contrary to non-linear methods, linear methods are usually simpler to implement, because they do not need any initialisation and do not suffer from convergence problems. However, they are usually not so accurate, because the distance minimised is not geometric and can lack physical meaning. In this case, the following algebraic constraint is defined by requiring the origin to lie on the line l:

$$[0,0,1]\boldsymbol{l} = (m_1 + t_1 m_2 + t_2)(d_1 + t_1 d_2) + (t_3 m_2 + t_4) t_3 d_2 = 0\,. \tag{4.7}$$

In practice, considering this equation does not present any advantage over the previous method because the equation remains non-linear in the case of a general camera. However, in the case of a zero-skew camera, the unknown values u_0, v_0 and r are related to the entries of T by

$$\mathsf{T} = \mathsf{K}_1^{-1} = \begin{bmatrix} 1 & 0 & -u_0 \\ & \frac{1}{r} & -\frac{v_0}{r} \\ & & 1 \end{bmatrix}, \tag{4.8}$$

and the following substitution can be carried out: $t_1 = 0$, $t_2 = -u_0$, $t_3 = \frac{1}{r}$ and $t_4 = -\frac{v_0}{r}$. It leads to the equation

$$(m_1 - u_0)d_1 + \left(\frac{1}{r}m_2 - \frac{v_0}{r}\right)\frac{1}{r}d_2 = 0\,, \tag{4.9}$$

which appears to be linear in the unknowns r^2u_0, v_0 and r^2:

$$\begin{bmatrix} -d_1 & -d_2 & m_1 d_1 \end{bmatrix} \begin{bmatrix} r^2 u_0 \\ v_0 \\ r^2 \end{bmatrix} = -m_2 d_2 \,. \tag{4.10}$$

This equation is similar to the one obtained by Gurdjos *et al.* in [62, 61] using the centre-line constraint. A least-square solution can be obtained by using for example the pseudo-inverse.

Both linear and non-linear algorithms for computing the invariant intrinsic parameters encapsulated in the NIAC are summarised in Algorithm 3.

4.4.3 Computation of F_1

Computing F_1 is a simple matter of finding the isotropic scaling factor f which maps the NIAC into a conic passing through the images of the two circular points for each image. We proceed as follows. Having computed K_1, the system of IAC can be mapped into the system of circles centred at the origin, which takes the form

$$\omega'_f = \mathsf{F}_{\frac{1}{f^2}} = \begin{bmatrix} \frac{1}{f^2} & & \\ & \frac{1}{f^2} & \\ & & 1 \end{bmatrix} \sim \begin{bmatrix} 1 & & \\ & 1 & \\ & & f^2 \end{bmatrix} , \tag{4.11}$$

while the transformed images of the circular points are mapped to

$$\boldsymbol{P}' = \mathsf{K}_1^{-1}(\boldsymbol{h}_1 + \mathrm{i}\boldsymbol{h}_2) = \boldsymbol{h}'_1 + \mathrm{i}\boldsymbol{h}'_2 \quad \text{and} \quad \boldsymbol{Q}' = \mathsf{K}_1^{-1}(\boldsymbol{h}_1 - \mathrm{i}\boldsymbol{h}_2) = \boldsymbol{h}'_1 - \mathrm{i}\boldsymbol{h}'_2 \,. \tag{4.12}$$

By requiring $\boldsymbol{P}'$ and $\boldsymbol{Q}'$ to be on ω'_f, we obtain

$$(\boldsymbol{h}'_1 \pm \mathrm{i}\boldsymbol{h}'_2)^\top \omega'_f (\boldsymbol{h}'_1 \pm \mathrm{i}\boldsymbol{h}'_2) = 0 \,, \tag{4.13}$$

which, after equating both real and imaginary parts to zero, leads to the two equations

$$\begin{cases} \boldsymbol{h}'^\top_1 \omega'_f \boldsymbol{h}'_1 &= \boldsymbol{h}'^\top_2 \omega'_f \boldsymbol{h}'_2 \,, \\ \boldsymbol{h}'^\top_1 \omega'_f \boldsymbol{h}'_1 &= 0 \,. \end{cases} \tag{4.14}$$

Substituting ω'_f by its expression in Eq. (4.11), and writing $\boldsymbol{h}'_1 = [h'_{11}, h'_{21}, h'_{31}]^\top$ and $\boldsymbol{h}'_2 = [h'_{12}, h'_{22}, h'_{32}]^\top$, the two following equations are obtained:

$$\begin{bmatrix} {h'_{31}}^2 - {h'_{32}}^2 \\ h'_{31} h'_{32} \end{bmatrix} f^2 = \begin{bmatrix} {h'_{12}}^2 + {h'_{22}}^2 - {h'_{11}}^2 - {h'_{21}}^2 \\ -h'_{11} h'_{12} - h'_{21} h'_{22} \end{bmatrix} . \tag{4.15}$$

Algorithm 3 Computation of the parameters encapsulated in the NIAC

1. For each view of the grid, estimate the homography $H = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}$ between the calibration plane and its image.

2. For each view, precompute constants m_1, m_2, d_1 and d_2 defined in Eq. (4.4).

3. *Linear solution minimising an algebraic distance:*

   (a) Assemble all vectors $[-d_1, -d_2, m_1 d_1]$ and all constants $-m_2 d_2$ into respectively an $n \times 3$ matrix A and an n-vector $\boldsymbol{b}$ (where n is the number of views).

   (b) Compute pseudo-inverse $\mathsf{A}^+ = (\mathsf{A}^\top \mathsf{A})^{-1} \mathsf{A}^\top$.

   (c) The parameters are given by $[r^2 u_0, v_0, r^2]^\top = \mathsf{A}^+ \boldsymbol{b}$.

   or non-linear solution minimising a geometric distance:

   (a) If the camera skew is non negligible, find the parameters t_1, t_2, t_3 and t_4 which minimise the sum of squared distances defined in Eq. (4.5). If the skew is negligible, set $t_1 = 0$ and replace previous distance by the one defined in Eq. (4.6). The parameters can be initialised from the results of the previous algorithm by setting $t_1 = 0$, $t_2 = -u_0$, $t_3 = \frac{1}{r}$ and $t_4 = -\frac{v_0}{r}$.

   (b) The solution is $\mathsf{K}_1 = \begin{bmatrix} 1 & t_1 & t_2 \\ & t_3 & t_4 \\ & & 1 \end{bmatrix}$, from which the parameters can be computed.

It should be noted that the first equation is indeterminate if $h'_{31} = h'_{32}$, and the second one if $h'_{31} = 0$ or $h'_{32} = 0$. Both equations are simultaneously singular if $h'_{31} = h'_{32} = 0$, which corresponds to the case where the optical axis of the camera is perpendicular to the calibration plane. If this configuration is discarded, there exists always at least one equation and f can be computed uniquely. If more than one equation are available, a least square solution can be computed. In general two equations are considered, but in some instances, more may be available, this is for example the case when several images are taken without varying the focal length.

4.4.4 Computation of R and t

For each image, we have the following constraint on the extrinsic parameters:

$$\begin{bmatrix} \boldsymbol{r}_1 & \boldsymbol{r}_2 & \boldsymbol{t} \end{bmatrix} \sim F_f^{-1} K_1^{-1} H \,. \tag{4.16}$$

The equality is up to a scale factor. The absolute value of the scale factor can be determined by requiring the norm of the first two columns of the term on the right hand-side to be one (the columns of a rotation matrix are unit vectors), while the sign is obtained by requiring the observed object to be in front of the image plane of the camera. $\boldsymbol{r}_3$ is given by $\boldsymbol{r}_3 = \boldsymbol{r}_1 \times \boldsymbol{r}_2$. The orthogonality of the matrix is usually not satisfied due to noise, but can be enforced for example by computing the Singular Value Decomposition (SVD) of R and requiring each singular value to be equal to one (see [154]).

4.4.5 Practical considerations

Normalisation

Normalisation is carried out before computing the homographies between the calibration plane and the image plane. The technique employed is described in [72] and consists in normalising world points and images points such that their centroid coincide with the origin and the average distance from the origin is $\sqrt{2}$. In the case of the non-linear methods, no extra minimisation is required. In the case of the linear method, it can be shown that weighting each term in Eq. (4.10) by the inverse of $\sqrt{d_1^2 + d_2^2}$ produces a very good approximation of the geometric distance

defined in Eq. (4.6), which also ensures good conditioning of the system. More information can be found on this topic in [62, 61]. The fact that our techniques requires very little normalisation is a strong advantage over other plane-based calibration techniques such as [138], which rely heavily on normalisation.

Degenerate Configurations

It has been seen earlier that a minimum of three or four views of the calibration plane is necessary, depending on the camera model. In addition, the three following assumptions have been made during the discussion: i) the optical axis of the camera is not perpendicular to the calibration plane (Section 4.4.3 and Appendix D), ii) the chords defined by the pairs of images of the circular points do not pass through the principal point of the camera (Result 1), iii) and no such chords are parallel (Result 1). After observing that the chords are the vanishing lines of the calibration planes observed, it is straightforward to show that ii) corresponds to the case where the optical axis of the camera is parallel to the calibration plane, while iii) corresponds to the case where two cameras are related by a translation and/or a rotation along an axis parallel to the calibration plane. This characterises all the degenerate configurations.

4.5 Results

In this section, the methods presented earlier are tested and evaluated. A comparison with the method presented in [138] is also given. When referring to these methods, the following terminology is adopted: *Sturm & Maybank* denotes the Sturm and Maybank method described in [138], *linear NIAC (s=0)*, *non-linear NIAC (s=0)* and *non-linear NIAC* denote the methods based on the NIAC which minimise, respectively, an algebraic distance with zero-skew assumption, a geometric distance with zero-skew assumption, and a geometric distance with a general camera model (no zero-skew assumption). It should be noted that the method *linear NIAC (s=0)* is identical to the method derived by Gurdjos *et al.* using the centre line constraint in [62, 61]. Gurdjos *et al.* refer to a theorem of projective geometry (Poncelet's theorem) to characterise the locus of the principal point when a zero-skew camera is zooming. Although

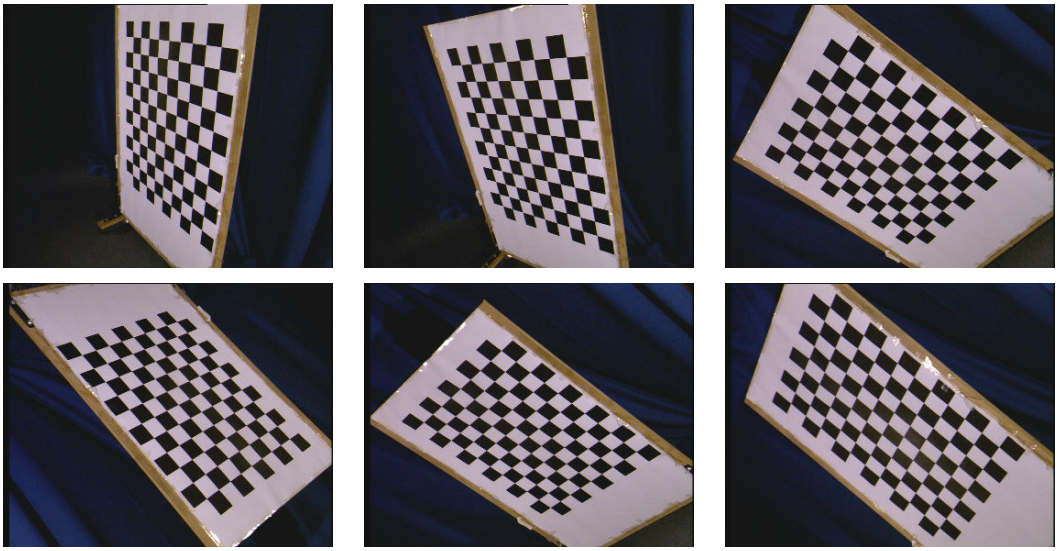


Figure 4.5: *Real images used for calibration. Each image illustrates a different zoom setting.*

this is different from the NIAC concept, both methods result in the same linear system of equations in the case of a zero-skew camera.

In all experiments, the camera is pointing at a planar calibration grid (see *e.g.* Fig. 4.5). The position, orientation and zoom are varying for each frame. For each image, the homography is computed using the Direct Linear Transform (DLT) method as described in [72], with the appropriate normalisation. Then the different methods are applied.

4.5.1 Synthetic data

The calibration target used for simulations consists of a square grid of size $20\,\text{cm} \times 20\,\text{cm}$ which contains 10×10 control points. The grid coincides with the plane $Z = 0$ of the world reference frame. The synthetic camera has the following constant intrinsic parameters: $u_0 = 384\,\text{pixels}$, $v_0 = 247\,\text{pixels}$, $r = 1.167$. We conducted different sets of experiments for the following values of the skew angle: $\theta = 89.9\,^\circ$ and $\theta = 89\,^\circ$. In practice most cameras will exhibit very little skew, and the skew parameter s can be identified to zero, *i.e.* $\theta \approx 90\,^\circ$. The focal length is the only varying intrinsic parameters. For each frame, the focal length is assigned a random value between 476 pixels and 1428 pixels, following a uniform distribution on this interval. The optical centre of the camera is located on a sphere with radius 0.5 m and centred at the middle of the calibration grid. The position and orientation of the camera is generated by

applying the following Euler transformation. A random rotation is applied successively around the Z axis (rotation), the X axis (precession) and finally the Z axis (nutation). The rotation around the X axis is constrained between $30\,^\circ$ and $70\,^\circ$, so as to be in the optimum condition required by the *Sturm & Maybank* method. Under such conditions, the grid occupies the whole image at the maximum zoom factor. Some Gaussian noise is added in the coordinates of each imaged control point in order to simulate image noise.

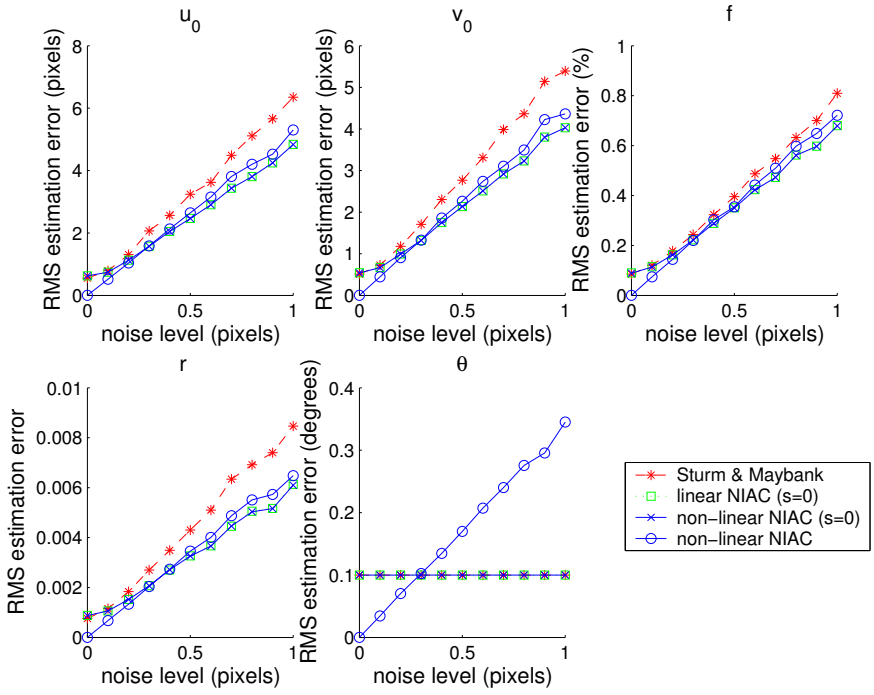
Knowing the ground truth parameter values, it is possible to compare the accuracy of the different methods. The evaluation criterion adopted is the RMS estimation error defined by: $\epsilon_{\text{est}} = \sqrt{\frac{1}{N}\sum_i (x - \overline{x})^2}$, where $\overline{x}$ is the ground truth parameter and x is the estimated parameter. The RMS value is computed for each parameter, from a total of 1000 experiments. In the case of the fixed intrinsic parameters, an absolute error is computed, while a relative error is computed for the focal length. This is a good measure of how closely the estimated camera parameters match the noise-free camera parameters. It was chosen not to compute the RMS reprojection error. The main reason is that, due to the arbitrary motion of the camera, it is difficult to place an extra calibration pattern in the scene which is visible from all viewpoints without generating occlusions in some views. This means that, to avoid occlusions, the RMS reprojection error would have to be evaluated on the same control points used for calibration. In that case the RMS reprojection error corresponds to a residual error, which is well known to be a poor measure of the quality of the solution obtained (see [72], Chapter 4). For example, the *non-linear NIAC* method is expected to always lead to lower residuals because it has one extra degree of freedom compared to the other methods and can therefore fit the data better, which does not mean it computes more accurately the camera parameters.

Two types of experiments were carried out. In the first case, the influence of the image noise was studied by varying the standard deviation of the spatial perturbation added to the image feature coordinates from 0 to 1 pixel. 10 images are considered during these experiments. The results are shown in Fig. 4.6. In the second set of experiments, the influence of the number of image frames was considered. The image noise level was constant and set to 1 pixel during these experiments. The results are shown in Fig. 4.7. In each case, the experiments were done with two values for the skew angle: $\theta = 89.9^\circ$ and $\theta = 89^\circ$. The case $\theta = 89.9^\circ$ corresponds to a low skew, which is the case of most cameras. It should also be mentioned that in the case of a purely skewless camera ($\theta = 90^\circ$), the results obtained are similar to the case $\theta = 89.9^\circ$,

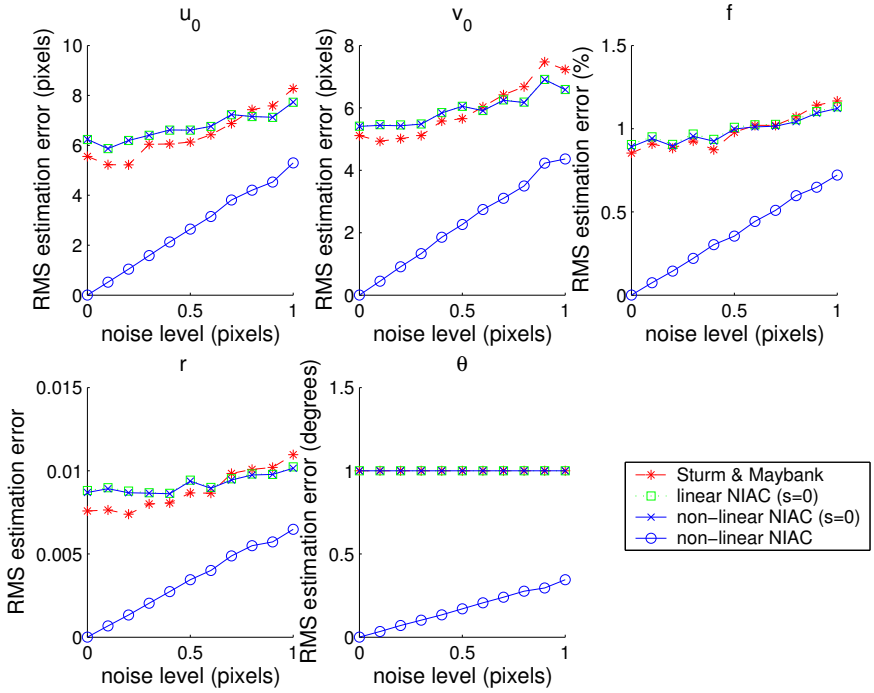
and have been omitted for this reason.

It can be observed that the error in the estimation of the parameters increases linearly with the image noise level. The methods based on the NIAC are usually more accurate than the *Sturm & Maybank* method. In the case of a small skew, the best performing methods are the ones using the NIAC (either linear or non-linear) based on the zero-skew assumption ($s = 0$). However if the skew parameter differs more significantly from zero, then the *non-linear NIAC* method becomes more accurate. Similarly, when varying the number of frames, it appears that the *linear NIAC ($s = 0$)* and *non-linear NIAC ($s = 0$)* are always more accurate than the *Sturm & Maybank* method. The *non-linear NIAC* is usually not so accurate when a small number of frames is considered. However, with a larger number of frames it becomes more accurate than *Sturm & Maybank*, and than the other methods based on the NIAC in the case of larger skew.

The methods based on the NIAC present the following advantages compared to the *Sturm & Maybank* method. Firstly they exploit some invariance properties, which guarantees that the number of parameters estimated at each stage is small and constant, while the number of unknowns estimated simultaneously by the *Sturm & Maybank* method increases linearly with the number of images and has no bound. This presents an advantage because it means that the complexity of the problem does not increase with the number of images considered. Secondly, the *Sturm & Maybank* method is based on the minimisation of an algebraic distance, while the methods based on the NIAC consider either a geometric distance (case of *non-linear NIAC* and *non-linear NIAC (s=0)*) or a close approximation of a geometric distance (case of *linear NIAC (s=0)*). The minimisation of a geometric distance usually leads to more accurate results that the minimisation of algebraic distances which sometimes lack physical meaning. It appears that the methods which rely on the zero-skew assumption (case of *linear NIAC (s=0)* and *non-linear NIAC (s=0)*) generally produce very similar results. Closer inspection of the graph would show that the non-linear method is slightly more accurate, but that the improvement is not as large as expected. This suggests that the algebraic distance minimised by the linear method is a very good approximation of a geometric distance. This is explained by the fact that the aspect ratio is close to one. It is expected that the results of the linear method would deteriorate if the aspect ratio differed more significantly from one. The *non-linear NIAC* method is the most accurate when a significant skew is present ($\theta = 89\,^\circ$) and a large number of views is considered. In the case of smaller skew values or no skew at all, the *non-linear NIAC* method is usually

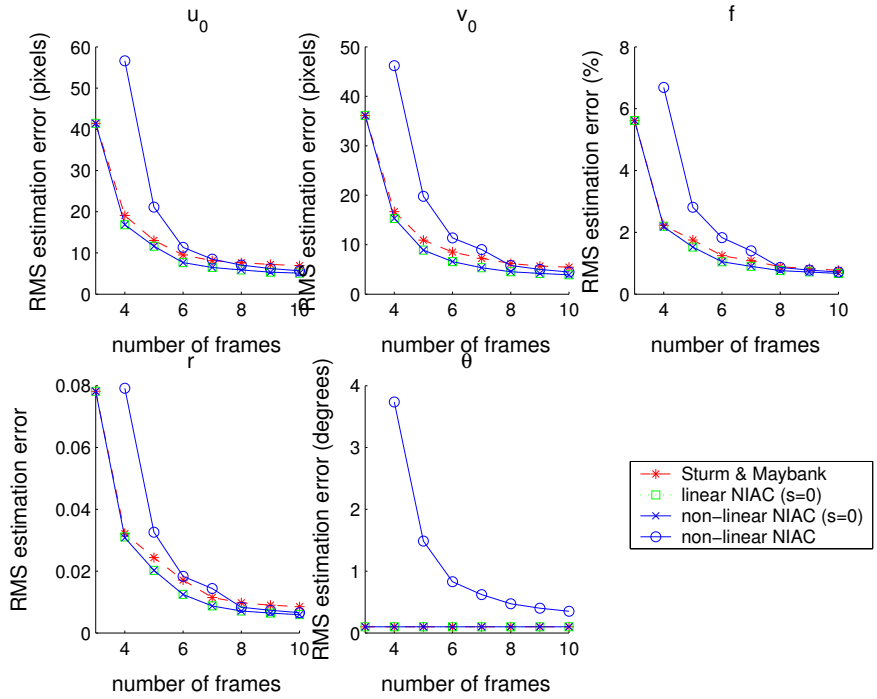


(a) $\theta = 89.9°$

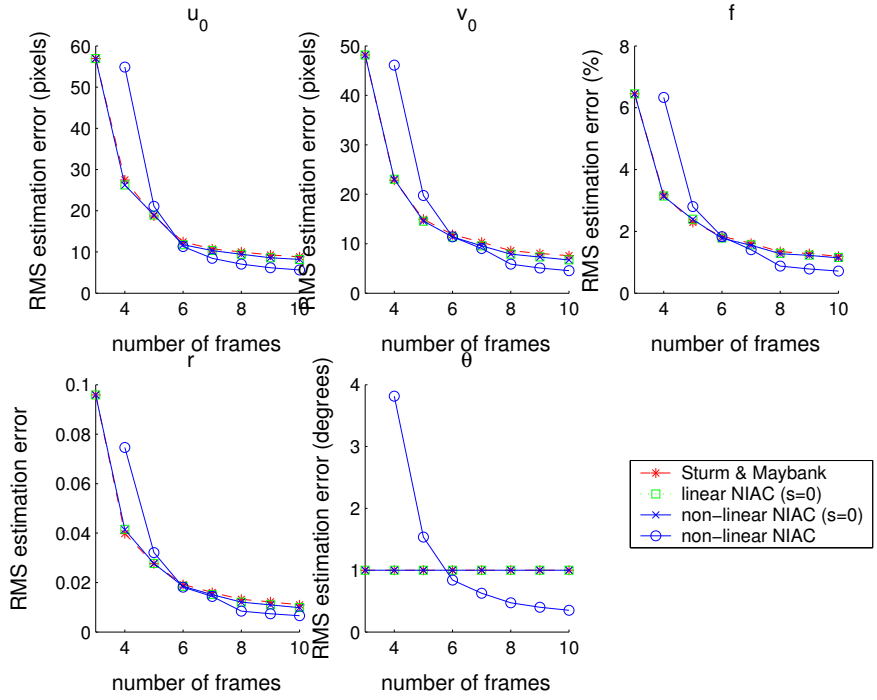


(b) $\theta = 89°$

Figure 4.6: *Results with synthetic data. Influence of the noise. The RMS estimation error was computed for each parameter from a total of 1000 experiments. The noise level indicated represents the standard deviation of the zero mean Gaussian noise added to the image coordinates. 10 images were used for calibration. The experiments were made with* $\theta = 89.9°$ *(a) and* $\theta = 89°$ *(b).*



(a) $\theta = 89.9°$



(b) $\theta = 89°$

Figure 4.7: *Results with synthetic data. Influence of the number of frames. The RMS estimation error was computed for each parameter from a total of 1000 experiments. The noise added to the image coordinates is Gaussian with zero mean and standard deviation* $\sigma = 1$ *pixel. The experiments were made with* $\theta = 89.9°$ *(a) and* $\theta = 89°$ *(b).*

Table 4.2: *Intrinsic parameters invariant to zooming calibrated from the 30 real images.*

	u_0 (pixels)	v_0 (pixels)	r	θ (degrees)
Sturm & Maybank	371.0	299.2	0.912	90
linear NIAC (s=0)	366.3	317.5	0.914	90
non-linear NIAC (s=0)	366.3	317.7	0.913	90
non-linear NIAC	364.9	316.3	0.912	90.19

less accurate than the other methods. This suggests that including the skew parameter in the calibration penalises the method in the case of negligible skew values. Most cameras presenting negligible skew, it is usually preferable not to include this parameter in the calibration.

4.5.2 Real data

We carried out some experiments with real data. The camera used is a Sony DXC-9100P equipped with a Fujinon S12×5BRM-38 zooming lens which has a 5–60 mm focal length range. The lens exhibits very low lens distortion which can be ignored during calibration. We grabbed a sequence of 30 images of the grid shown in Fig. 4.5. The camera is hand-held, and the zoom settings are changed manually by the person who holds the camera. Each group of five successive images were acquired with a constant zoom setting, varying only the position and orientation of the camera. We show in Fig. 4.5 one image for each of the six zoom settings.

We calibrate the camera using all 30 images. The values obtained for intrinsic parameter invariant to zooming are shown in Table 4.2, while Fig. 4.8 shows the mean focal length computed for each group of images acquired at constant focal length, for each method. Error bars representing plus or minus three times the standard deviation have been added to the graphs. It can be observed that the different methods produce consistent values. Also, it appears that the methods based on the NIAC generally produce slightly smaller standard deviations, which suggests they capture better the set of values expected for the focal length.

4.6 Conclusions

A novel technique for calibrating a zooming camera has been presented. The technique capitalises on the invariance properties of a novel mathematical object called the Normalised Image of the Absolute Conic (NIAC) in order to simplify the calibration equations. The NIAC is a

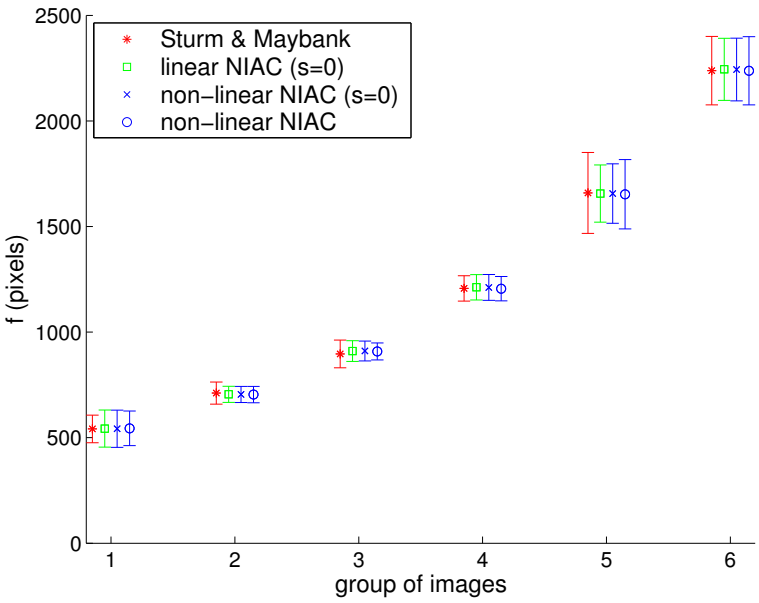


Figure 4.8: *Results with real data. The graphs show the error bars for the estimation of the focal length for each group of images in Fig. 4.2. Each image within a group was acquired with a fixed focal length. The error bars represent plus or minus three times the standard deviation of the estimated focal lengths.*

mathematical representation of the intrinsic camera parameters which are invariant to zooming, translation and rotation of the camera. In practice, the NIAC can be estimated from a minimum of three or four images (depending on the camera model) of a planar calibration grid taken from arbitrary positions, orientations and zoom settings. The main idea consists in using the invariance properties to decompose the calibration problem into three simpler sub-problems, each having constant number of unknowns. Different implementations have been proposed in order to accommodate the different types of cameras, in particular with zero and non-zero skew cameras. Results with synthetic and real images showed that the algorithms based on the NIAC are usually more accurate than the Sturm and Maybank algorithm [138] which estimates all the parameters simultaneously.

An apparent limitation of the method is the assumption of a fixed principal point. Theoretical results and experiments suggest that this is a valid assumption as long as camera calibration consists only in computing the projection matrix, *i.e.* it is not necessary to estimate separately all camera parameters. This is the case of many applications in computer vision.

Part II

Photometric aspect: Image-based object reconstruction using Helmholtz Stereopsis

Chapter 5

Background

5.1 Introduction

Image-based object reconstruction consists in inferring information on the geometry of a 3D scene (also called *structure*) from a set of 2D images acquired with a camera. In contrast with the first part of the thesis which focused on estimating the properties of the camera used, the objective of this part is to measure geometric properties of the objects observed by the camera. These two tasks are obviously complementary. The more knowledge about the sensor we have, the more information about the scene it is possible to extract. The applications of image-based object reconstruction are numerous; they include robotics, measurement for quality control, virtual reality applications and the entertainment industry, for example the creation of special effects in films.

Imaged-based object reconstruction has been a focus of research for more than 40 years now, however there exists still no fully automatic system giving a satisfactory general solution to the problem. One reason why image-based object reconstruction is such a challenging task is that it is usually an ill-posed problem in the case of non-model based reconstruction techniques: the solution is usually not unique and does not depend continuously on the data. To tackle this problem, a wide variety of methods have emerged; they all attempt to regularise the problem and make it tractable by making some assumptions. Some common assumptions simplify the geometry of the object, by assuming for example smooth surface variations, or simplify the reflectance properties of the object, by assuming for example a Lambertian reflectance model.

Another reason why reconstruction using a camera is such a challenging task is that a camera is a passive sensor, *i.e.*, it does not interact with the scene, but only produces a snapshot of the light intensities emitted by the scene. Extracting depth information from such an image can be difficult because the intensity of each pixel is related only indirectly to the geometry of the scene. This contrasts with active sensors which generate a signal and measure how it is perturbed by the scene to derive geometry information; a common example of an active sensor is the laser range scanner, which computes depth information from the measure of the time of flight of an emitted laser beam reflecting on the object surface [154]. Active methods are now well established, and they are typically more accurate than passive methods, nevertheless they require complicated equipment, and can fail to detect the surface of objects with non-Lambertian properties.

This chapter presents an overview of the main image-based object reconstruction methods. We concentrate on automatic reconstruction techniques involving cameras - this includes active methods which use controlled light sources, but not other active methods which project a predefined pattern on the scene and for which reconstruction is more trivial. This chapter plays a similar role to Chapter 2 in the case of camera calibration. The aim of this review is to compare the different state-of-the-art reconstruction techniques, and motivate the choice of Helmholtz Stereopsis (HS) as the reconstruction technique adopted in this thesis. The reader already familiar with these techniques may want to quickly read through this chapter or proceed directly to the next one.

5.2 Conventional stereo methods

Conventional stereo techniques are inspired by the human vision system. They use two or more images of a scene taken from different viewpoints to estimate depth. The images can be obtained from a collection of fixed cameras or from a single moving camera. Two fundamental problems can be distinguished: i) the correspondence problem and ii) the reconstruction problem. We start by describing how these problems are solved in the case of a pair of images, and then generalise to N views.

5.2.1 Two-view geometry

Correspondence problem

The correspondence problem consists of matching points which correspond to the projection of the same physical scene point in each image. This appears at first sight as a very difficult problem because a point in one image can be matched a priori with any point in the other image. However, there exist a number of assumptions which can be made in order to make this problem tractable. We distinguish two main classes of methods: area-based and feature-based.

Area-based methods compare intensity profiles in neighbourhoods of potential matches in order to define correspondences. Good reviews on this topic can be found for example in [46, 45, 118]. Traditionally, a correspondence is represented by a value called *disparity* which measures the amount of displacement between two corresponding pixels. For each pixel in the first image, the corresponding pixel in the second image is the one which maximises a measure of similarity between the two neighbourhoods, thus yielding a dense set of correspondences represented by a disparity map. Commonly adopted measures of correlation are the Sum of Squared Differences (SSD) or the normalised cross-correlation. Such measures are computed over pixel neighbourhoods defined by generally fixed-size rectangular windows.

Area-based methods present several limitations. Firstly, the window must be large enough to include sufficient intensity variations. Secondly, it is implicitly assumed that the area covered by the window is not significantly distorted between the two images; if the scene exhibits rapid depth variations, this may require to use small dimension windows. There is obviously a trade-off between maximising the intensity variations and minimising the disparity variations when choosing the window shape and size. A number of adaptive algorithms which are able to adjust automatically the window to the intensity and disparity patterns have been proposed [79, 56]. These techniques have been reported to improve significantly the reconstruction, however matching remains difficult in the case of poorly textured surfaces or non smooth objects. In addition, the underlying assumption of area-based methods is that corresponding points have similar intensities in the two images. This requires that: i) the intensity of the light reflected by object surface varies slowly with respect to the direction of viewing, ii) the illumination conditions do not vary significantly between the acquisition of the two images, and iii) the

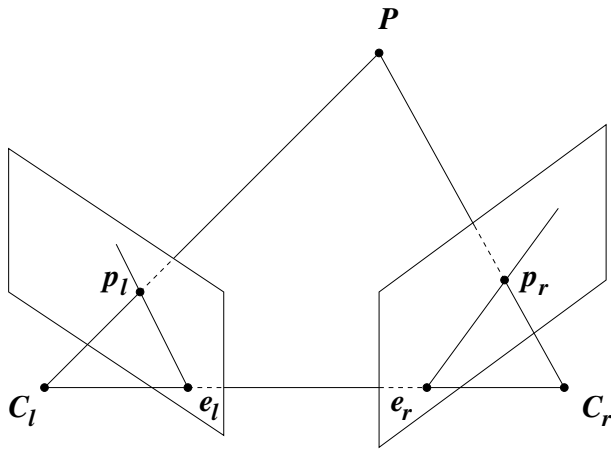


Figure 5.1: *Illustration of the epipolar geometry. The baseline $(C_l C_r)$ intersects the image planes in two epipoles e_l and e_r. Given an image point p_l in the left image, its corresponding point p_r in the right image is constrained to lie on a line passing through the epipole e_r and called epipolar line.*

baseline (distance separating the two cameras centres) is small with respect to the distance from the surface observed.

Feature-based methods address some of the previous limitations by considering distinguished image primitives which present the advantage of being more stable under illumination and view-point changes than image windows. While the earliest implementation were restricted to small baseline (see [46]), more recently a new class of algorithms for wide-baseline stereo have emerged [11, 113, 143, 94, 158]. The advantage of matching images separated by a wide baseline is that it enables more accurate subsequent triangulation. The features considered by these methods are selected for their invariance properties with respect to perspective foreshortening and illumination variations. A variety of features have been considered. They are defined for example by corners [11], segments formed by pairs of corners [143], quadrangles delimited by edges [113] or regions driven by the local extremal properties of the intensity function [94, 158]. Each feature being attributed some descriptors characterising its invariance properties, the matching problem consists in finding pairs of features which minimise an appropriate metric in the space of descriptors. Most methods try to minimise the Mahalanobis distance, however it has been shown in [94] that more robust metrics can be considered. One disadvantage of feature-based techniques, compared to area-based techniques, is that they provide only a sparse reconstruction of the scene.

Epipolar geometry

Finding a correspondence would be a very time consuming task if every single pixel or feature in the second image has to be checked for correspondence. Fortunately, there exists a simple geometric constraint which enables to restrict the search to a single line; this constraint is called the *epipolar constraint* [46, 154, 72]. The *epipolar geometry* is illustrated in Fig. 5.1. In a nutshell, the epipolar geometry imposes that the two optical centres and the two image points in correspondence must be coplanar (so that the two incoming light rays intersect in a 3D point); the plane thus defined is called an *epipolar plane*. Given one point in an image, this forces the corresponding point to lie on the image line defined by the intersection of the epipolar plane with the image plane of the other camera. Mathematically, the epipolar geometry is encoded in a matrix called the *fundamental matrix* F which satisfies the equation $\boldsymbol{x}_1^\top F \boldsymbol{x}_2 = 0$ for any pair of image points $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ in correspondence. In the latter equation, $F\boldsymbol{x}_2$ represents the equation of the epipolar line corresponding to the image point $\boldsymbol{x}_1$, on which $\boldsymbol{x}_2$ is constrained to lie. In practice, once the fundamental matrix has been estimated, the search for correspondences can be simplified further by applying a preliminary warping of the images such that conjugate epipolar lines are horizontal; this process is called *rectification* [57].

The fundamental matrix has rank 2, it has therefore seven degrees of freedom (9 entries minus one degree of freedom for the scale factor and another degree of freedom because of the rank 2 constraint), and can be computed from a minimum of seven point correspondences [72]. Many methods for fundamental matrix estimation have been proposed and a good review with comparative evaluation can be found in [174]. The eight-point algorithm [67] remains a popular algorithm for estimating the fundamental matrix because of its simplicity (it is linear) and because it performs nearly as well as more complex algorithms involving non-linear minimisation if appropriate normalisation is carried out [67, 174, 98]. The estimation of the fundamental matrix and the matching problem are intimately related: the fundamental matrix requires point correspondences, however point correspondences are constrained by the fundamental matrix. Robust techniques based for example on Random Sample Consensus (RANSAC) [52] have been considered to solve automatically the problem [147]. With such techniques, the set of putative correspondences can be re-assessed at each iteration in order to assure consistency with the fundamental matrix estimate. It should be mentioned that in the case where the intrinsic

parameters of the camera have been calibrated, image points can be expressed in camera coordinates, rather than image coordinates, in which case the fundamental matrix takes a particular form and is called the *essential matrix* E [91].

Finally, the epipolar geometry constraint is by far the most commonly used constraint in stereo vision; however there exists a number of other geometric constraints which can be considered. These include: continuity, uniqueness, ordering, disparity gradient constraints, *etc*. The reader interested in these constraints is referred for example to [46].

Reconstruction problem

Once image points have been matched, the reconstruction problem reduces to intersecting pairs of rays defined by the backprojection of points in correspondence. This procedure is called *triangulation*. This task can be carried out unambiguously in the case of fully calibrated cameras, because image points and the camera projection matrices define uniquely the incident rays that must be intersected. This is however not the case when the cameras are partially calibrated or uncalibrated. In particular it has been shown that the camera projection matrices and therefore the 3D structure of the scene can be recovered only up to an arbitrary similarity transformation in the case where only the intrinsic camera parameters are known [91] or only up to an arbitrary projective transformation in the case of uncalibrated cameras [65].

Even for calibrated cameras, given two camera matrices and some point correspondences, the backprojected rays will generally not meet perfectly in 3D space because of errors in the localisation of the matched image points and estimation of the epipolar geometry. For this reason, it is convenient to reformulate the triangulation problem in terms of minimisation of an appropriate cost function. This cost function must be invariant to the class of transformations characterising the ambiguity in the reconstruction in order to provide meaningful results [71]. For example, although computation of the mid-point of the common perpendicular to both rays, or computation of an optimum solution to a system of linear equations as in [65], both provide valid results in the case of fully calibrated cameras or cameras with known intrinsic parameters, these methods are not suitable for the case of uncalibrated cameras. For this reason, Hartley and Sturm proposed an analytical solution, unaffected by projective transformation of the input data, which minimises the sum of squared distances between image points and conjugate

epipolar lines in each image [71].

5.2.2 N-view geometry

The accuracy of the reconstruction can naturally be improved by considering a larger number of overlapping views of the scene, thus providing more powerful disambiguation constraints. In particular, it has been shown that algebraic representations analogous to the fundamental matrix in the two-view case can be defined for three and four views; these entities are called respectively *trifocal* [68, 148] and *quadrifocal* [149, 69] *tensors*. With the fundamental matrix, they provide very powerful tools for scene reconstruction, because they encapsulate the multilinear matching constraints arising in two, three or four views, into a single algebraic object. Unfortunately, there exist no generalisation of these tensors to more than four views [149]. In many applications, a much larger number of views is considered, therefore different reconstruction techniques must be adopted. One of the main challenges that arises then is how to ensure consistency of the correspondences and reconstruction with all images.

Multi-baseline approaches

Okutomi and Kanade have proposed a multi-baseline approach which is able to find correspondences using an arbitrary number of views separated by a lateral displacement, thus eliminating the need for subsequent consistency enforcement [105]. The main idea is that, if the search for correspondences is expressed in terms of scene depth estimation (or equivalently its inverse) rather than the disparity for each pixel in a reference image, then the measure of correlation can be extended to multiple frames by summing the SSD computed for each pair of images. The authors report that the method results in more precise matching because the measure of correlation presents a sharper global extremum as more baselines are added. This implementation is limited to a particular spatial configuration, however it has been shown in [101] that it can be generalised and used successfully with a large number of non-aligned cameras. In this work, they use a set of 51 cameras mounted on a geodesic dome of 5-metre diameter. They first produce an initial reconstruction for each group of 3 to 6 neighbouring cameras with a modified version of the multiple-baseline algorithm presented in [105]; the reconstructions is

then merged into a consistent model by using a volumetric integration method. Collins proposed a different technique, which can cope with a more general camera configuration [35]. In this approach, the matching problem, which was expressed so far in the image space, is reformulated in the 3D space. Matching is carried out by sweeping a plane in the 3D space and backprojecting all image features into this plane. Counts resulting from all image features are accumulated and used to estimate the likelihood of a 3D feature being present at each cell in the plane. Contrary to [105], the computational cost is linear in the number of images, however it produces only a sparse reconstruction based on detected edges.

Structure from motion

The reconstruction of the scene from correspondences established across multiple images obtained by moving a camera is called *structure from motion*. The most general approach to determine structure from motion is *bundle adjustment* [152], which consists in estimating the projection matrices and the 3D points which minimise the reprojection error defined in Eq. (2.12). This approach gives a Maximum Likelihood (ML) estimate in the case of additive Gaussian image noise, however its solution requires a large-scale minimisation for which there is no direct solution and a starting point must be provided [72]. In the case of an affine camera, Tomasi and Kanade developed a factorisation algorithm which has the advantages of involving only linear equations and provides a ML affine reconstruction [146].

The approach has been generalised to projective cameras by Sturm in [139]. The algorithm iterates between estimation of some homogeneous scale factors for each image point, and performing a factorisation similar to [146] for the given scaling factors. This poses the problem of the choice of the initialisation, and also the convergence properties of this iterative algorithm are not clear. In addition, the algorithm no longer provides a ML estimate in this case. A general drawback of factorisation methods [146, 139] is that they assume all points are simultaneously visible in all images, which is usually not the case because features can become temporarily occluded. There exist many other structure from motion algorithms (see for example [106]); in particular, we can distinguish between *batch algorithms*, which process all images at the same time, and *sequential algorithms* which update the structure and motion whenever a new frame is added to the sequence.

Many approaches do not solve the correspondence and reconstruction problems independently, but combine them into an iterative algorithm [14, 53, 110]. In these approaches, correspondence and reconstruction are first solved for sub-groups of images for which there exists multilinear matching constraints which can be computed directly, then the results are refined by enforcing a consistency constraint within all sub-groups. For example, in [14], image pairs or image triplets defined by consecutive frames are used to track and match image primitives and produce sequentially a reconstruction of the scene. In [53], image triplets are also considered, however the sequential approach is replaced by a hierarchical approach where structure and motion are estimated first for each image triplet, which are then registered into subsequences and finally into the entire sequence of images; registration is done by estimating 3D homographies which maximise the overlap between reconstructions. This approach presents the advantage of distributing the errors optimally across the sequence of images. In [110], Pollefeys *et al.* propose an alternative approach where two images are chosen for reference and define the initial structure and motion, which is then updated every time a new view is added. Hartley and Zisserman discuss the possible strategies for obtaining an initial reconstruction in [72]. They recommend to terminate any reconstruction with a global minimisation step using bundle adjustment. It should be mentioned finally that the techniques presented in Section 2.3.6 are particularly useful for upgrading the final reconstruction from projective to metric.

5.3 Volumetric methods

Volumetric methods eliminate some of the limitations inherent to conventional stereo techniques by reasoning directly in 3D. In particular, volumetric techniques eliminate the necessity to extract features or to have textured objects that can be matched easily. Also these techniques can cope with arbitrary camera positions, *i.e.* it is not necessary to impose a small baseline between images. With such techniques, the 3D space is usually restricted to a bounding box enclosing the scene to reconstruct and is discretised into small elements called *voxels*. The reconstruction problem can then be expressed in terms of classifying the voxels into different states according to their properties, for example transparent (empty) or opaque (full). We distinguish two main classes of volumetric methods depending on the cue used to infer the 3D information: *shape from silhouette* and *shape from photo-consistency*. An additional review of

volumetric methods can be found in [124].

5.3.1 Shape from silhouettes

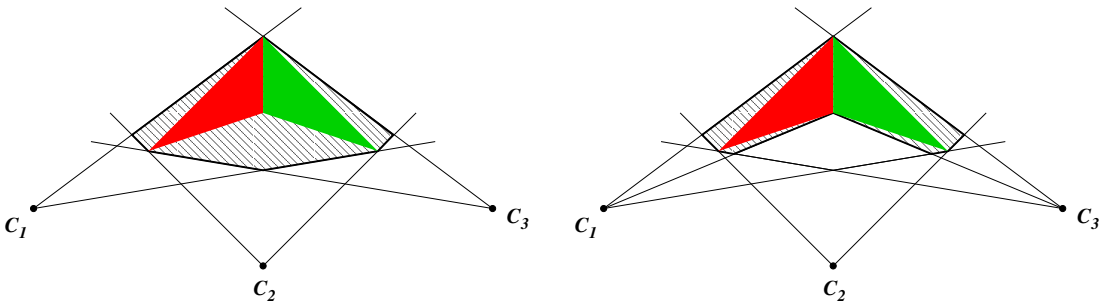
The main idea of this class of methods is that the result of the segmentation of the projection of an object from the background defines a 2D shape, called a *silhouette*, which backprojects into a cone tangent to the 3D object. The volume resulting from the intersection of the cones generated by all images defines an approximate reconstruction of the observed object. The first implementation is due to Martin and Aggarwal. In [93], they apply simple thresholding techniques followed by a connected-component analysis in order to extract the object silhouette. They represent the scene by what they call a volume segment representation and consists of a set of line segments parallel to one axis.

Other implementations have considered a voxel representation based on *octrees* [140, 59]. With these methods the 3D space is discretised into elementary volume elements called *voxels*. The reconstruction task consists in labelling voxels as either opaque or transparent depending on whether they belong to the scene objects or not. A memory-efficient representation is obtained using octrees. The reconstruction starts at a coarse resolution with all voxels initialised as opaque. Each voxel is projected into each image, and depending on its location within the silhouettes, the voxels are either labelled opaque, transparent or ambiguous. Ambiguous voxels are processed recursively at finer levels until all voxels have been classified or the required level of accuracy has been obtained. In [140], for example, the object is placed on a turntable which is used to generate the multiple views required for reconstruction. The camera has been precalibrated and the turntable indexed so that the projection matrix is known for each view. It has been shown in [59] that a reconstruction can be obtained from uncalibrated cameras. In this paper, the authors use two cameras pointing at approximatively orthogonal directions to define a projective sampling of 3D space. They relate each novel view to these two views by computing the corresponding trifocal tensor, which is used to project and determine the state of each voxel. As in [140], a hierarchical coarse-to-fine approach based on octrees is adopted.

The previous approaches used a regular discretisation of the 3D space. Such representations are rather simple to implement, however they are computationally expensive and they lack precision because of the quantisation effect [22]. With such techniques, high accuracy can be

obtained only at the cost of adopting a high-resolution discretisation of space, which increases significantly the run-time. This trade-off is overcome in [22] by adopting an irregular grid. The irregular grid consists of tetrahedrons defined by applying Delaunay triangulation on sample points belonging to the object surface. A final reconstruction is obtained by extracting the surface of the visual hull.

The main limitations of volumetric techniques based on silhouette intersection are i) the inability to reconstruct concavities (unless the camera can be placed near the object inside the concavity, which is usually not the case) and also ii) the necessity to be able to segment the object from its background. In [84], Laurentini studies these ambiguities and proposes the term *visual hull* to define the best reconstruction obtainable by volume intersection techniques. The visual hull inferred by a given number of images may not always be enclosed in the convex hull of the object, depending on the object geometry and camera configurations, however it is guaranteed to contain the object, thus providing an upper bound for reconstruction. Segmentation can be done for example by blue screen technique (chroma keying), which requires a special laboratory setting to ensure that the object is surrounded by a uniform background with a given colour not appearing on the foreground object. This poses a problem if the object contains a similar colour. Alternatively, background subtraction techniques can be considered (see *e.g.* [140]). These require controlled lighting condition and can suffer from shadow effects.



(a) (b)

Figure 5.2: *Illustration of the different reconstructions obtained by shape from silhouettes (a) and shape from colour-consistency (b) in the case of a simple object made of two red and green triangles. In both cases the reconstruction is done from three cameras located at the positions C_1, C_2 and C_3, and the model obtained is represented by the union of the object and the hatched area. Shape from colour-consistency produces a more accurate reconstruction than shape from silhouettes because it exploits the additional colour information contained in the scene.*

5.3.2 Shape from photo-consistency

Contrary to shape from silhouette techniques which considered only a binarised version of the images obtained by segmentation, shape from photo-consistency techniques exploit the full photometric information contained in the images by introducing the notion of colour consistency [119] also called *photo-consistency* [82]. A shape is said to be *photo-consistent* with a set of images if, for each image in which a surface point is visible, the radiance leaving this point is equal to the radiance measured at the corresponding pixel. Thus, consistent voxels can be assumed to be surface voxels and attributed the colour of their projections; whereas inconsistent voxels can be assumed to correspond to empty space and can therefore be removed from the volume (see Fig. 5.2(b)). Starting with a 3D space with all voxels assumed opaque and applying the consistency test in order to carve away inconsistent voxels until all the visible voxels are colour-consistent leads to a reconstruction of the scene consistent with all the images. In analogy with the visual hull [84], Kutulakos and Seitz called the best reconstruction obtainable, which is consistent with the set of all the source images, the *photo-hull* [82]. Such a reconstruction is a more accurate approximation of the object geometry than the visual hull because photo-consistency allows the reconstruction of concavities whenever sufficient texture information is present on the object surface.

The determination of the visibility of the voxels is a fundamental problem. In the first implementation using colour-consistency, called Voxel Coloring [119], Seitz and Dyer define a constraint on the positions of the cameras called the *Ordinal Visibility Constraint* which allows the voxel space to be topologically sorted according to the distance from the cameras. Their approach guarantees that occluding voxels are visited before occluded voxels and thus allows a reconstruction via a single pass through this space. Their approach is efficient, but restricted to objects located outside the convex hull defined by the camera centres. Kutulakos and Seitz eliminate this limitation by proposing a multi-pass extension of Voxel Coloring called *Space Carving* [82]. In their implementation, they carry out near-to-far scans similar to the one in Voxel Coloring, but repeated along each axis of the 3D reference frame in both positive and negative direction, considering at each time only the cameras which are in front of the moving plane for consistency evaluation. Their approach allows arbitrary camera positions, but it is not optimal because it considers only a subset of the images for consistency evaluation, which may

lead to a failure to carve voxels inconsistent with the entire set of cameras but consistent with subsets of images.

Culbertson *et al.* address this problem by processing only the voxels whose visibility has changed at each iteration, until convergence [37]. Their approach leads to an optimal solution in the sense it is consistent with all images, but requires the use of complex data structures to compute the exact visibility of the voxels. In addition, it exhibits large run-time (up to 40 min) compared to Voxel Coloring (a few seconds or minutes). Alternatively, Eisert *et al.* [44] proposed a multi-hypothesis technique. In a first step, colour hypotheses are assigned to each voxel based on their projection in the set of images. In a second step, voxels located at the surface of the volume are checked and hypotheses inconsistent with images are removed. Voxels with no hypotheses remaining are carved away from the volume. The procedure is iterated until no further hypotheses can be removed, at which point there remain only voxels having a single hypothesis, which belong to the object surface and define a reconstruction of the object [44].

Almost all methods based on photo-consistency are based on the simplifying assumption that the scene is Lambertian [119, 82, 37, 44], *i.e.* the reflectance of a surface point is the same in all the directions. The advantage of making this assumption is that consistency takes a very simple form because consistent voxels are expected to have the same colour in each image. Under such an assumption consistency can be evaluated by simple thresholding of the standard deviation of the set of projection colours [119, 82, 37, 44]. There are several limitations to this thresholding approach. Firstly, the choice of the threshold affects directly the results obtained. A low threshold is very selective, and there is a risk to carve consistent voxels, while a high threshold may keep inconsistent voxels in the reconstruction. Slabaugh *et al.* relax this assumption in [126] by using an adaptive threshold, however this still requires the user to pre-define some thresholds. In other work, the hard limits imposed by a threshold have been replaced by some probabilistic measures of consistency [23, 16, 171]. Secondly, and more importantly, the Lambertian assumption is valid only for a restricted class of objects; most real objects are not Lambertian and are likely to be very poorly reconstructed with the previous algorithms. A method able to deal with specular highlights has been proposed in [170]. The method assumes that the light reflected by a surface is only modulated by the incident light, thus producing a set of colours which are collinear in the colour space when the viewing direction is varied. Under

this framework, photo-consistent voxels can be detected by evaluating the collinearity of the set of projection colours in the RGB space (a Lambertian surface correspond to the limit case where the line segment is restricted to a point). The method is able to reconstruct a broader class of objects than the previous methods, however it relies on the assumption that the scene is illuminated by light sources which have the same colour, and also the surface model is still limited to a certain class of objects. Bonfort and Sturm propose another method for reconstructing specular surfaces where consistency is determined in terms of the consistency of the set of normals computed at each voxel [17]. The method is however restricted to purely specular surfaces and requires the use of a calibrated pattern during reconstruction.

A number of extensions to the previous algorithms have been proposed. In [114], Prock and Dyer propose methods for improving the performance of voxel coloring algorithms. In particular, they show that the computation of the projection of voxels in images can be optimised by using hardware texture mapping. They also propose a coarse-to-fine approach based on an octree to optimise memory usage and processing time, which is normally an issue with volumetric methods. In [125], it is shown that the voxel space can be warped to an infinite domain thus allowing the reconstruction of objects located far away from the cameras as well as the background. One limitation of the methods reviewed so far is that they all rely on the assumption of accurately calibrated cameras; errors in calibration result in erroneous projection of voxels in the images, which in turn corrupt the photo-consistency measure. In order to address this problem, Kutulakos proposes to define photo-consistency up to specific image transformations that they call *suffle transformations* [81]. Saito and Kanade tackle the problem of reconstruction with uncalibrated cameras in [116]. As in [59], they select two views where the cameras are approximately pointing at orthogonal directions and are related by their fundamental matrix, in order to define a projective grid. They compute the projection of voxels by using the fundamental matrices relating novel views to the initial two views. Another extension is proposed in [39] for modelling scenes containing transparent objects.

5.4 Photometric methods

In contrast with previous methods which exploited the displacement of image features due to camera motion relatively to the object in order to reconstruct the geometry of the scene,

photometric methods infer 3D information from the radiance measured at each image pixel under different illumination conditions. With these methods the scene is viewed by a single camera. If we assume for simplicity that the camera is orthographic and that scene points (x, y, z) project to pixels (x, y) in the image, the reconstruction problem consists in assigning a depth $z = f(x, y)$ to each image point (this is easily generalised to projective cameras), thus producing a $2\frac{1}{2}$D reconstruction. Typically, rather than direct depth estimation the problem is formulated in terms of the estimation of surface gradient (p, q) or surface normal $(p, q, -1)^\top$, where p and q are defined by:

$$p = \frac{\partial f(x, y)}{\partial x} \quad \text{and} \quad q = \frac{\partial f(x, y)}{\partial y} \,. \tag{5.1}$$

Once the gradient has been estimated at each surface point, depth can be recovered by integration (see for example [145]). In order to be integrable, it is usually necessary to enforce the integrability constraint which guarantees that the mixed second partial derivatives are equal (see for example [55]). It can be observed that photometric methods are able to reconstruct only a surface patch where each point can be modelled by a surface height function f. In the case of more complex objects for which a full 3D model is needed, it may be necessary to reconstruct several surface patches and merge them together.

5.4.1 Shape from shading

A single image provides only one constraint on the radiance at each image pixel, however there are two unknowns p and q to estimate at each image point. This is clearly an ill-posed problem. It has been shown however that it is still possible to produce a reconstruction by imposing some additional constraints, for example on the smoothness of f. Such methods are called *shape from shading* and are due originally to Horn (see for example [75]). They are usually computationally intensive and lack robustness due to the necessity of introducing constraint in order to regularise the problem. A review of shape from shading methods can be found in [172].

5.4.2 Photometric stereo

Photometric stereo considers several images obtained by varying the illumination of the scene while keeping the camera at a fixed position. Unlike shape from shading, this problem is well-posed and does not require to impose additional constraints. The idea was first introduced by Woodham in [167]. In photometric stereo, the relation between the image intensity and the surface gradient represented by p and q, for given illumination conditions, is usually modelled by a reflectance map. With the knowledge of the reflectance map, each image defines one equation in the two unknowns p and q for each pixel. These equation are usually non-linear and therefore a unique solution cannot be guaranteed with only two views. In the case of a Lambertian surface, these equations become linear when expressed with respect to the unit surface normal at each pixel [167], and a linear solution can be computed from three images or more using least square techniques. Because three views or more lead to an over-constrained system of equations, it is possible to recover additional information such as the albedo at each surface point in the case of Lambertian surfaces. In comparison with conventional stereo methods which work well on rough surfaces with discontinuities in surface orientation, or textured surfaces with varying reflectance, photometric stereo is more efficient in the case of smooth surfaces with few discontinuities and uniform properties [167].

The implementation of photometric stereo is very simple in the case of Lambertian surfaces, however such surfaces are not representative of most real surfaces. Ikeuchi proposed an algorithm for the reconstruction of specular surfaces [76]. The method requires to replace point sources by area sources in order to be able to avoid localised specularities that could not be measured otherwise. The previous method is however limited to purely specular, *i.e.* mirror like, surfaces. In order to model a wider variety of surfaces, more complex reflectance models have been considered [102, 141]. In [102], a method is presented for the reconstruction of surfaces with hybrid reflectance models which are a combination of Lambertian and specular models. The method does not require any prior knowledge of the relative strength of Lambertian and specular components, and is able to estimate surface normal and also the reflectance parameters at each surface point. However it requires a large number of images in order to provide a sufficiently dense sampling of the photometric function. Alternatively, Tagare and de Figueiredo have considered a class of reflectance maps called *m-lobed reflectance maps* to

model real surfaces [141].

Unfortunately, a formal reflectance model is not applicable for all surfaces. In [168], Woodham measures empirically the reflectance properties of the surface. The reflectance map is stored in a look-up table built by observing a calibration sphere made of the same material as the object to reconstruct. This approach is able to model arbitrary types of surfaces, however the surface reconstructed must be made of the same material as the calibration object, have constant albedo, and both objects must be illuminated and viewed under identical conditions. In this work, Woodham also considered the use of multi-spectral images in order to acquire simultaneously all images. He uses three light sources equipped with red, green and blue filters to illuminate the scene which is captured with a 3CCD camera. In [32], photometric stereo has been generalised to colour images and showed to result in more accurate reconstruction compared to grey-level images because of the larger number of constraints provided by colour information.

So far the previous methods all considered light sources with known positions. In [73], Hayakawa proposed a method which does not require any *a priori* information about the light source positions and strengths. The algorithm uses Singular Value Decomposition (SVD) to factorise the matrix containing the image intensities for each frame into two components encapsulating respectively surface and light-source information; the method is similar to the factorisation method employed for structure from motion in [146]. The method is able to compute surface normals, surface reflectance, light direction and light source intensity. However, there exists an ambiguity in the reconstruction, which is represented by an arbitrary invertible 3×3 matrix. Hayakawa resolved the ambiguity by imposing an additional constraint on surface reflectance or light-source intensity. Belhumeur *et al.* characterise the ambiguity in the case of continuous Lambertian surfaces [15]. They show in this case that if $z = f(x, y)$ is the true surface, any surface $z' = \lambda f(x, y) + \mu x + \nu y$ with λ, μ, and ν real numbers ($\lambda \neq 0$) is an equally valid reconstruction; they call this ambiguous transformation a generalised bas-relief transformation. In the case where the surface albedo is constant or known in advance, or if all light sources have the same intensity, they show that the ambiguity reduces to a sign ambiguity (in-out ambiguity), which can be resolved by considering shadows (if present in the images). Drbohlav and Šára showed in [42] that the general ambiguity reduces to a two degree of freedom group of transformation in the case where the surface reflectance is the sum of a Lambertian and

specular component.

One limitation of photometric stereo is that it is based on a local shading model, *i.e.*, it assumed that the radiance at a surface patch is due only to the light internally generated at sources. Such a model is inaccurate because it ignores inter-reflection effects, *i.e.*, light generated by the reflection on other surface patches, or cast shadows, which are both global phenomena. In [103], Nayar *et al.* proposed a method able to deal with these effects. They start by generating a reconstruction using photometric stereo without taking into account inter-reflections, and then iteratively update the reconstruction by including the inter-reflections produced by the current reconstruction, until convergence. The approach is however limited to continuous surfaces and assumes a Lambertian surface model. In [168], it is showed that the over-constrained system of equation defined by at least three images of a surface with constant albedo can be used to form a confidence estimate. The confidence estimate measures the deviation from the local model and can be attributed to global phenomena such as inter-reflections or cast shadows which are not explained by the latter model. This provides a convenient mechanism for detecting such phenomena which corrupt the reconstruction.

5.5 Helmholtz Stereopsis

In contrast with previous methods which assumed the surface reflectance of the object reconstructed to be known in advance or to follow a particular parametric model, Helmholtz Stereopsis (HS) is able to reconstruct arbitrary surfaces, without making any assumption on their surface properties. The reflectance properties of a surface are measured by their *Bidirectional Reflectance Distribution Function (BRDF)*, which is defined as the ratio of the outgoing radiance to the incident irradiance at a given surface point [104]. HS exploits the symmetry of the BRDF with respect to the incoming and outgoing directions, which is known as *Helmholtz reciprocity*. This principle states that *the BRDF at a surface point remains unchanged when the viewpoint and the light source are interchanged*. The universality of this principle makes HS very attractive for reconstruction of surfaces - the only assumption made is that there are no inter-reflections. The idea of using Helmholtz reciprocity in computer vision first appeared in [92], and was later on implemented in [177, 178]. The method requires a camera and a point light source whose positions can be interchanged, thus producing reciprocal pairs of images.

The constraint derived enables estimation of the depth and normal at each pixel of a virtual camera sampling the 3D space from a minimum of three reciprocal pairs of images taken for different camera and light source configurations. This is only a necessary condition for points to be in correspondence, therefore the authors imposed an additional smoothness constraint which assumes the scene is made locally of fronto-parallel planes. We limit ourselves to a general description of the method for now; a more detailed description of the original algorithm as well as further developments will be presented in later chapters.

HS presents a number of advantages compared to other reconstruction techniques, namely [178]:

- It does not assume any model for the BRDF,
- It provides both depth and normal information, thus combining the advantages of conventional and photometric stereo,
- It is unaffected by lack of texture (unlike conventional stereo),
- It simplifies the detection of discontinuities (normally problematic with other techniques) because shadowed and half occluded regions are in correspondence in reciprocal pairs of images.

The following assumptions are implicit in HS:

- There are no self-occlusions,
- There are no self-shadows,
- There are no inter-reflections,
- The surface is locally smooth so that it can be represented locally by a reference plane,
- The BRDF is uniform over the area sampled by a camera pixel.

The original implementation of HS considered calibrated cameras and sources. This assumption has been relaxed in [179] where a novel matching constraint based on Helmholtz reciprocity is derived in the case of uncalibrated cameras and light sources with unknown strengths

and positions. The only assumption remaining is that the radiometric responses of the cameras are linear and equal and that the light sources are isotropic. The reconstruction obtained by enforcing this constraint presents a projective ambiguity, which can be resolved in a stratified manner by imposing additional geometric or photometric self-calibration constraints. The authors investigate the special case where the distance from the scene to the cameras and sources is large with respect to the scene relief. They show that in this case the photometric information allows the additional computation of the surface normals and the strength and direction of the light sources up to an arbitrary invertible transformation. They also observe that in this case the camera can be accurately modelled by an affine model, which allows reduction of the ambiguity in the reconstruction and all other previously computed information up to an arbitrary affine transformation, provided there is a minimum of four observed points and four cameras/sources pairs considered. The upgrade to metric follows from standard self-calibration techniques.

Other work showed that reconstruction is possible using HS with a single pair of reciprocal images [156, 180]. In [156], Tu and Mendonça reformulated the reconstruction problem in terms of finding an optimum path along epipolar lines using dynamic programming. The cost function minimised is derived from the Helmholtz reciprocity constraint and is therefore independent of the surface BRDF, which makes the method applicable with any type of surfaces. In addition, the cost function considered includes normal information, which imposes tighter constraints on the reconstruction than conventional dense stereo approaches based on dynamic programming. In another binocular implementation [180], Zickler *et al.* observed that the Helmholtz reciprocity constraint defines a first-order non-linear partial differential equation in the point coordinates and their first-order derivatives, for which they provide a solution in the simplified case of distant cameras and light sources, under scaled orthographic projection camera models. Their implementation proceeds in two steps. They first compute along each epipolar line a one-parameter family of solutions which is indexed by the choice of depth at the end-points of each line, and then impose a smoothness criterion across epipolar lines in order to select the correct solution for each epipolar line.

Jankó *et al.* [78] addressed the problem of radiometric calibration of the Helmholtz stereo setup. The radiometric calibration is necessary to compensate for the non-uniformity of the radiometric camera responses and the anisotropy of the light sources. They show that in the case of HS, it is sufficient to calibrate the ratio of the radiance due to the source over the pixel

sensitivity at each pixel in each image, and propose a method to compute these values from a minimum of two reciprocal pairs of images of an arbitrary planar surface. They report improvements by an order of magnitude in the surface normal estimation when radiometric calibration is performed. Other extensions of HS have been proposed in the context of registration of 3D models to a pair of reciprocal images [157] and computer graphics [121].

5.6 Conclusions

We have encountered very diverse image based object reconstruction techniques. These techniques can be classified for example according to the cue used to infer 3D information. For this reason, these techniques are grouped under the general category of *shape from X* techniques, where X represents the cue used for reconstruction. In conventional stereo, the main cue used is the disparity; volumetric methods have considered silhouettes or photo-consistency, while photometric methods are based on illumination or shading, and HS on Helmholtz reciprocity. We concentrated on the most popular techniques, however it is worth mentioning that the list of cues that can be used for reconstruction is not limited to these techniques. Other techniques are for example shape from focus/defocus, shape from texture, and shape from zoom. In addition, it is possible to combine different cues thus producing more efficient reconstruction techniques.

Another way to look at the reconstruction problem is to consider the class of objects to which the methods are applicable. It appears that the Lambertian assumption is predominant in computer vision because of its simplicity. It is at the basis of conventional stereo techniques as well as shape from photo-consistency. Even though the reconstruction of more general surfaces has been considered, in particular in the context of photometric stereo, these methods remain restricted to a certain class of surfaces following an assumed model or for which reflectance properties have been measured in advance. Shape from silhouette is one exception, however it has been observed that the reconstruction obtained by this method is limited to the visual hull, which is usually a coarse representation of the object. HS is another exception.

What is the best reconstruction technique? This depends on the equipment available, the time constraints (should the system be real-time?), the required degree of accuracy or flexibility, *etc*. If accuracy is the main concern however, it seems a good idea that the choice of the method

should be driven by the surface properties of the objects that we want to reconstruct, because deviations of the real surface properties from the model assumed by the reconstruction method will inevitably result in inaccuracies in the reconstruction of the scene geometry. Because one of our objectives is to improve the accuracy of the reconstruction of the widest class of objects possible, the rest of this thesis will concentrate on reconstruction using HS.

Chapter 6

Minimising a radiometric distance for accurate surface reconstruction with Helmholtz Stereopsis

6.1 Introduction

In the previous chapter, we reviewed the main image-based object reconstruction techniques. We observed that Helmholtz Stereopsis (HS) possesses some unique features which make the technique applicable to a wider class of objects than other techniques. In this chapter and the following, we continue the development of this technique, and propose a number of improvements and extensions aimed at improving the accuracy of the 3D model generated.

In this chapter, we concentrate on improving the accuracy of the surface normal estimation from a set of image correspondences using HS. As in most reconstruction techniques, two fundamental problems can be distinguished: the *correspondence* and the *reconstruction* problems. In the case of HS, the principle of Helmholtz reciprocity has been applied to formulate a matching constraint which is independent of the surface properties of the object reconstructed. An appropriate minimisation of this constraint results in a set of correspondences in sets of images, from which the depth and the surface normal can be reconstructed. The reconstruction of the normal, in particular, is of high importance because it has been shown to be less affected

by the smoothness assumptions made during reconstruction, compared to the depth estimate [177, 178]. For this reason, the final reconstruction is usually obtained from the integration of the normal field. Previous implementations of HS were limited to a linear least-square estimate obtained from Singular Value Decomposition (SVD) for the surface normal estimation. While the reconstruction problem appears as a more straightforward problem compared to the matching problem, in particular in the case of surfaces with arbitrary unknown surface reflectance properties, it remains however an essential part of the reconstruction technique and affects directly the final geometry of the reconstruction, and should therefore not be neglected. In this chapter, we carry out a deeper analysis of the normal reconstruction problem. In particular, after observing that the linear least squares solution minimises an algebraic distance, we propose an optimum solution based on a novel *radiometric* distance.

Linear algorithms have been extensively used in computer vision to solve a variety of problems such as camera calibration or scene reconstruction [72, 48]. These techniques proceed by defining a set of linear equations for which a solution is easily computed. In pratice, there exists no exact solution because all measurements are corrupted by noise; therefore an approximate solution is found by minimising an appropriate cost function. In the case of linear systems of equations, the solution is usually found by least squares techniques. The error defined by such a system of linear equations is sometimes called ”algebraic” because it measures how far the linear equations are from being satisfied in a purely mathematical sense. A popular algorithm for solving such problems is, for example, the Singular Value Decomposition (SVD) algorithm [112]. The reason why these algorithms are so popular is that there exists a linear (and therefore unique) solution and that this solution is usually computationally cheaper to compute than with more complicated methods. However one major criticism of such methods is that the algebraic distance usually lacks a physical meaning or interpretation.

Hartley [67, 70] and more recently Izquierdo and Guerra [77] analysed the reasons for the poor performance of the method minimising algebraic distances. Hartley showed that the poor performance can be attributed to the lack of numerical consideration when solving the system, more precisely to the poor conditioning of the set of equations resulting from the noise contaminating the input data. He observes that a major cause for the poor conditioning of the system of equations is the lack of homogeneity in the input data, and proposes a simple normalisation scheme based on translation and scaling of the input data in order to address the problem. The

concept of normalisation was originally introduced in the case of the computation of the fundamental matrix via the eight-point algorithm [67], and was later generalised to other problems such as camera calibration or estimation of the trifocal tensor [70]. Alternatively, Izquierdo and Guerra considered another class of normalisation transformations defined by diagonal matrices in order to improve the conditioning of the system - this presents some similarities with the standard technique of rescaling rows and/or columns of the equation matrix described in [60]. They also show that another cause of instability is the linear dependency between the rows of the equation matrix. In the same line of research, a variety of estimation techniques have been developed and applied to improve the solution of various problems in computer vision [80, 98, 85, 100, 99, 30, 31]. In all these works, normalisation has been shown to improve greatly the accuracy of the parameters estimated.

In spite of the improvement due to normalisation, methods minimising an algebraic distance are not as accurate as methods minimising a physically and statistically meaningful distance. The choice of the optimum distance is motivated by the type of measurements involved and how they are affected by noise. For example, in geometric problems such as camera calibration, homography estimation, fundamental matrix estimation or structure from motion, the distances minimised are naturally geometric distances. A popular choice of cost function in these cases is the reprojection error, which measures the distance between measurements and their reprojection [67, 70]. In the case of surface normal estimation using HS, the correspondence problem is assumed solved already, therefore the measurements affected by noise are the pixel intensities or radiance values at the matched points, and an optimum distance must therefore be defined in the space of radiances. In this thesis I develop a novel distance called the *radiometric distance*. It measures the modification to be made in each image in order to satisfy exactly the Helmholtz reciprocity constraint at the point considered; this yields a Maximum Likelihood (ML) surface normal estimate under standard Gaussian image noise conditions. The main disadvantage of considering such distances, rather than algebraic ones, is that non-linear minimisation techniques are usually required to compute the solution. Non-linear minimisation techniques are iterative and usually not as stable as linear techniques, in particular when a large number of variables is optimised. Fortunately, in the case of the defined radiometric distance, the total number of variables to optimise can be reduced to only two, in which case a solution can be computed at extremely low computational cost.

The chapter is structured as follows. We start by giving a brief overview of HS and describe the conventional linear least square solution for surface normal estimation; we refer to this solution as the *algebraic* solution. We then define a novel *radiometric* distance in Section 6.3. In the following section, we observe that an extension is required in order to support image saturations. Finally, we give some results and compare algebraic and radiometric solutions with both synthetic and real data, before concluding the chapter.

6.2 Overview of Helmholtz Stereopsis

Consider the configurations of object, light source and camera which are illustrated in Fig. 6.1. $\boldsymbol{O}_l$ and $\boldsymbol{O}_r$ are two points in space and $\boldsymbol{X}$ is a point on a surface. We denote by $d_l = \|\boldsymbol{O}_l - \boldsymbol{X}\|$ and $d_r = \|\boldsymbol{O}_r - \boldsymbol{X}\|$ the distance from the points $\boldsymbol{O}_l$ and $\boldsymbol{O}_r$ respectively to the surface point $\boldsymbol{X}$, and define $\boldsymbol{v}_l = \frac{1}{d_l}(\boldsymbol{O}_l - \boldsymbol{X})$ and $\boldsymbol{v}_r = \frac{1}{d_r}(\boldsymbol{O}_r - \boldsymbol{X})$, which represent the unit vectors pointing from the surface point $\boldsymbol{X}$ to $\boldsymbol{O}_l$ and $\boldsymbol{O}_r$ respectively. The surface normal at $\boldsymbol{X}$ is given by the unit vector $\boldsymbol{n}$. The Bidirectional Reflectance Distribution Function (BRDF) $f(\boldsymbol{X}, \boldsymbol{u}, \boldsymbol{v})$ of the surface point $\boldsymbol{X}$ is by definition the ratio of the outgoing radiance along the direction $\boldsymbol{v}$ to the incident irradiance along the direction $\boldsymbol{u}$. If we position an isotropic light source of intensity κ at $\boldsymbol{O}_l$ and a camera at $\boldsymbol{O}_r$, the pixel intensity[1] i_r observed by the camera is:

$$i_r = f(\boldsymbol{X}, \boldsymbol{v}_l, \boldsymbol{v}_r)\frac{\boldsymbol{v}_l \cdot \boldsymbol{n}}{d_l^2}\kappa\,. \tag{6.1}$$

If the positions of the light source and the camera are now interchanged[2], an analogous formula is obtained for the radiance i_l observed by the camera at position $\boldsymbol{O}_l$:

$$i_l = f(\boldsymbol{X}, \boldsymbol{v}_r, \boldsymbol{v}_l)\frac{\boldsymbol{v}_r \cdot \boldsymbol{n}}{d_r^2}\kappa\,. \tag{6.2}$$

The two images observed by such cameras form what is known as a reciprocal pair. The Helmholtz reciprocity principle imposes that $f(\boldsymbol{X}, \boldsymbol{v}_l, \boldsymbol{v}_r) = f(\boldsymbol{X}, \boldsymbol{v}_r, \boldsymbol{v}_l)$. Denoting $\boldsymbol{s}_l = \frac{1}{d_l^2}\boldsymbol{v}_l$ and $\boldsymbol{s}_r = \frac{1}{d_r^2}\boldsymbol{v}_r$, Eq. (6.1) and Eq. (6.2) can be combined to form the constraint [178]:

$$(i_l \boldsymbol{s}_l - i_r \boldsymbol{s}_r) \cdot \boldsymbol{n} = 0\,. \tag{6.3}$$

[1] We adopt the convention that the pixel intensity equals the scene radiance (or equivalently that they are proportional). This is usually a reasonable assumption for high quality cameras. If it is not the case, radiometric calibration can be performed in order to meet this requirement.

[2] Note that the same light source with the same intensity $\boldsymbol{\kappa}$ is used.

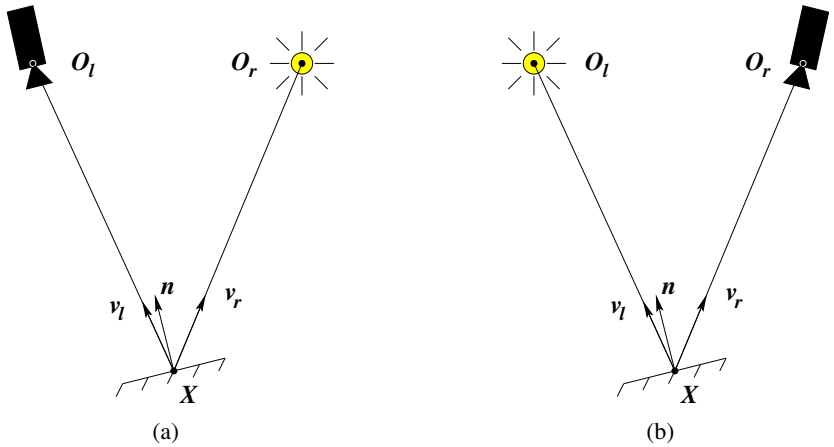


Figure 6.1: *A reciprocal pair of images. The position and orientation of the camera and light source are interchanged.*

Two such constraints provided by two reciprocal pairs are sufficient to compute the surface normal. If more constraints (one per reciprocal pair of images) are available, it is possible to define a multi-ocular matching constraint and thereby estimate both the depth of the surface point and its normal [92, 177, 178]. The remarkable feature of this constraint is that it uses only a non-parametric property of the BRDF (Helmholtz reciprocity) and does not make any use of the actual BRDF values, thus enabling the reconstruction of objects with arbitrary unknown surface properties. The implementation of this constraint is discussed in details in the rest of this section. We start by a general description of the algorithm, and then describe separately how the correspondence and the reconstruction problems are solved in more details.

6.2.1 Algorithm summary

HS requires to define a sampling of the 3D space around the object of interest. In [177, 178], the authors introduced a virtual camera in the scene in order to define such a sampling. Equivalently, the 3D space can be discretised regularly into voxels as in the case of other volumetric methods. This is effectively equivalent to the sampling proposed by [177, 178] in the case of an orthographic virtual camera. The bounding box of the volume thus defined must be chosen large enough to contain the object to reconstruct. The concept is illustrated in Fig. 6.2.

Like most volumetric methods, HS reasons directly in 3D in order to establish correspondences. In the case of HS, correspondences are found by hypothesising that some voxel contains an

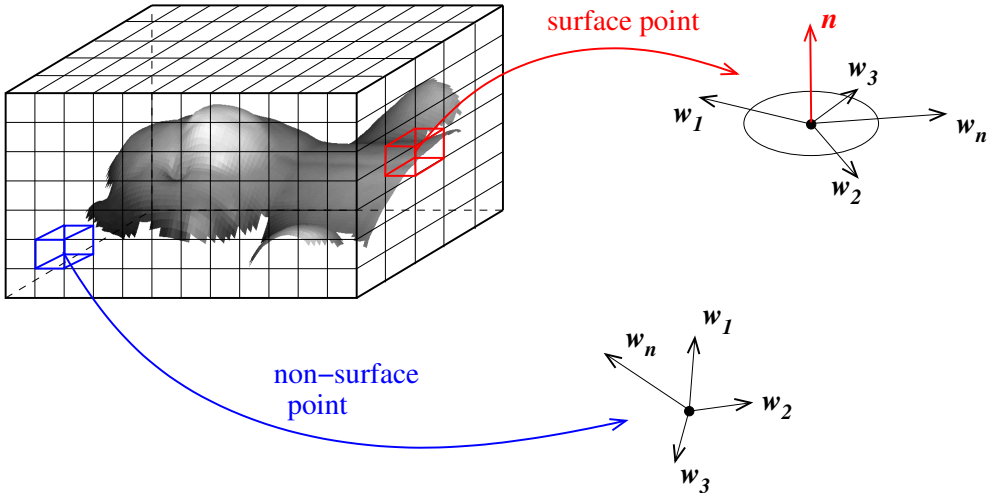


Figure 6.2: *Illustration of the HS reconstruction algorithm. The 3D space is discretised into voxels. Voxels are hypothesised to contain an object surface point, and we use the distributions of vectors* $\boldsymbol{w} = i_l \boldsymbol{s}_l - i_r \boldsymbol{s}_r$ *to test the validity of this assumption at each voxel. We show two examples. The blue voxel, which does not contain any surface point, yields a random distribution of vectors* $\boldsymbol{w}$*, while the red voxel, which contains a surface point, results in a set of coplanar vectors* $\boldsymbol{w}$*. This defines a method to identify surface points and also compute the surface normal at such points.*

object surface, and then testing the validity of the hypothesis based on the distribution of vectors $\boldsymbol{w} = i_l \boldsymbol{s}_l - i_r \boldsymbol{s}_r$ defined in Eq. (6.3) by each reciprocal pairs of images at the given voxel. In a nutshell, voxels containing a surface point are expected to produce a coplanar distribution of vectors $\boldsymbol{w}$, while voxels which do not contain any surface points are likely to yield a random distribution of vectors $\boldsymbol{w}$. The mechanism for discriminating surface from non-surface voxels is described in more details in Section 6.2.2.

Once surface voxels have been identified, this defines effectively a set of image point correspondences. For each surface voxel, the surface normal can then be identified by finding the normal to the distribution of vectors $\boldsymbol{w}$. We refer to this part of the algorithm as the reconstruction problem and describe it in details in Section 6.2.3.

6.2.2 Correspondence problem

In the case of HS, the standard solution to the correspondence problem [92, 177, 178] is based on SVD. We summarise it below. If $N \geq 3$ constraints defined in Eq. (6.3) (one for each

reciprocal pair) are stacked into a matrix, we obtain

$$W\boldsymbol{n} = \mathbf{0} \quad \text{with } W = \begin{bmatrix} (i_{l_1}\boldsymbol{s}_{l_1} - i_{r_1}\boldsymbol{s}_{r_1})^\top \\ (i_{l_2}\boldsymbol{s}_{l_2} - i_{r_2}\boldsymbol{s}_{r_2})^\top \\ \cdots \\ (i_{l_n}\boldsymbol{s}_{l_n} - i_{r_n}\boldsymbol{s}_{r_n})^\top \end{bmatrix}. \tag{6.4}$$

The main idea is to look at the distribution of row vectors in W in order to establish whether or not the point considered is a surface point. If the intensities used for constructing the matrix W come from a point which is located on a surface, these vectors are coplanar and the matrix W is expected to be of rank 2. If the point is not part of an object surface, the rows of the matrix W are likely to be random and W to be of rank 3. This is the ideal case. In practice, the problem is more complex because the measurements are corrupted by noise, and the rank 2 constraint is never going to be satisfied, in a purely mathematical sense, at surface points. For this reason, an alternative measure of rank has been proposed.

After applying SVD, W can be written:

$$W = UDV^\top \quad \text{with } D = \begin{bmatrix} \sigma_1 & & \\ & \sigma_2 & \\ & & \sigma_3 \end{bmatrix} \text{ and } \sigma_1 \geq \sigma_2 \geq \sigma_3 \geq 0\,. \tag{6.5}$$

The support measure is defined in terms of the second and third singular values σ_2 and σ_3 of W by

$$s = 1 - \frac{\sigma_3}{\sigma_2}\,. \tag{6.6}$$

A similar measure has been used in previous work [92, 177, 178]. The measure defined in Eq. (6.6) is strictly equivalent to the measure defined in [92, 177, 178], and has the advantage of normalising the value between 0 and 1, a value close to 1 corresponding to a high chance of the point being located on a surface. The general idea is that the three column vectors from the orthogonal matrix V represent the three principal directions of an ellipsoid, and the corresponding singular values σ_1, σ_2 and σ_3 represent the strength along each axis. As such, ideally at a surface point the ellipsoid should be flat, *i.e.* σ_3 should be zero. In practice, because of noise, the system has always rank 3 and we use the ratio defined in Eq. (6.6) to measure the non-flatness of the ellipsoid, *i.e.* how close numerically the matrix W is from being rank 2.

It is important to mention that Helmholtz reciprocity gives only a necessary condition for a correspondence to exist. This is not a sufficient conditions. One way of resolving this ambiguity is to impose an additional constraint on the surface. In [92, 177, 178], it has been assumed that the surface is locally constant. In this case the support measure s, *i.e.* measure of rank, is averaged over a rectangular window of fixed size centred at the point of interest. Correspondences are found by finding the window which maximises this value. In previous implementations, it has been assumed that the surface to reconstruct is a $2\frac{1}{2}$-D surface. Thus the search for correspondences is equivalent to finding the optimum depth (or elevation) along each vertical direction. In practice, the support measure is computed at the centre of each voxel of the grid, and only the one which maximises the support measure along each vertical line is retained. Typically the local depth constancy assumption results in a low resolution reconstruction. For this reason, the depth value is used only as a means to solve the correspondence problem. A more accurate estimate of the geometry is obtained from the computation of the normal, which is described next.

6.2.3 Reconstruction problem

Given some correspondences, previous approaches [92, 177, 178] have estimated the normal $\boldsymbol{n}$ at each point as the column vector of V corresponding to the smallest eigenvalue, from the SVD of W expressed in Eq. (6.5). At this stage, no additional smoothness constraint is required, therefore the normal $\boldsymbol{n}$ is computed only from the intensity values at the projection of the point considered (*i.e.* no windowing was applied). It has been observed in [92, 177, 178] that the normal estimate thus obtained is a more accurate estimate of the object geometry than the depth value obtained when solving the correspondence problem; in particular it preserves better the high frequency content of the surface variations (up to the normal sampling). This is attributed to the fact that no assumption was made about the local surface shape in this case. As in photometric methods, integration of the normal field has been used at the end of the reconstruction to compute an accurate 3D model of the object.

It can be shown (see for example [67]) that the solution obtained from SVD is the vector $\boldsymbol{n}$ which minimises

$$\|W \cdot \boldsymbol{n}\|^2 = \sum_j [(i_{l_j} \boldsymbol{s}_{l_j} - i_{r_j} \boldsymbol{s}_{r_j}) \cdot \boldsymbol{n}]^2 \quad \text{subject to } \|\boldsymbol{n}\| = 1\,. \tag{6.7}$$

This cost function can be re-written in the form

$$\sum_j d_{\text{alg}}(\boldsymbol{n}, i_{l_j}, i_{r_j}, \boldsymbol{s}_{l_j}, \boldsymbol{s}_{r_j})^2 \,, \tag{6.8}$$

where d_{alg} denotes the algebraic distance associated with a pair of reciprocal measurements i_{l_j} and i_{r_j} and a normal $\boldsymbol{n}$, which is defined by

$$d_{\text{alg}}(\boldsymbol{n}, i_{l_j}, i_{r_j}, \boldsymbol{s}_{l_j}, \boldsymbol{s}_{r_j})^2 = [(i_{l_j}\boldsymbol{s}_{l_j} - i_{r_j}\boldsymbol{s}_{r_j}) \cdot \boldsymbol{n}]^2 \,. \tag{6.9}$$

It can be observed that

$$d_{\text{alg}}(\boldsymbol{n}, i_{l_j}, i_{r_j}, \boldsymbol{s}_{l_j}, \boldsymbol{s}_{r_j})^2 = \|i_{l_j}\boldsymbol{s}_{l_j} - i_{r_j}\boldsymbol{s}_{r_j}\|^2 \cos^2\alpha_j \,, \tag{6.10}$$

where α_j denotes the angle between the vector $(i_{l_j}\boldsymbol{s}_{l_j} - i_{r_j}\boldsymbol{s}_{r_j})$ and the surface normal $\boldsymbol{n}$. $\cos^2\alpha_j$ represents clearly a physical quantity that we would like to minimise, however the physical meaning of the scaling factor $\|i_{l_j}\boldsymbol{s}_{l_j} - i_{r_j}\boldsymbol{s}_{r_j}\|^2$ is not so obvious. It is possible to eliminate the influence of this term by normalising the rows of W to one, however it may be the case that the weights introduced by this factor in the cost function defined in Eq. (6.8) play an important role by attenuating the effect of measurements corresponding to low intensities or cameras/light sources located far away from the scene point. The effect of this term is not very clear, and it is not very clear either whether it should be included or not. We will come back briefly to this problem in the results section. In any case, such a measure (whether normalised or not) does take into account the nature of the noise contaminating the measurements, and for this reason cannot be optimum. It has been considered in previous work mainly for its simplicity. In the next section, we investigate a novel measure which is optimum.

6.3 Surface reconstruction based on a radiometric distance

6.3.1 Definition of the radiometric distance

Since the fundamental entities observed (and likely to be affected by noise) are intensities or equivalently radiances, it seems a natural idea to perform the minimisation directly in the space of radiances. We search for the surface normal $\boldsymbol{n}$ and the pairs of estimated intensities

$\{\hat{i}_{l_j}, \hat{i}_{r_j}\}_j$ which minimise the following cost function:

$$\sum_j \left[(\hat{i}_{l_j} - i_{l_j})^2 + (\hat{i}_{r_j} - i_{r_j})^2\right] \quad \text{subject to } (\hat{i}_{l_j}\boldsymbol{s}_{l_j} - \hat{i}_{r_j}\boldsymbol{s}_{r_j}) \cdot \boldsymbol{n} = 0 \ \ \forall j\,. \tag{6.11}$$

Note that $\boldsymbol{s}_{l_j}$ and $\boldsymbol{s}_{r_j}$ are known in the previous equation because the cameras are calibrated. This cost function measures the corrections to be made in the intensities observed in each reciprocal pair of images in order to fulfil *exactly* the constraints in Eq. (6.4). After eliminating the constraints, the cost function can be written:

$$\sum_j \left[(\hat{i}_{l_j} - i_{l_j})^2 + \left(\frac{\boldsymbol{s}_{l_j} \cdot \boldsymbol{n}}{\boldsymbol{s}_{r_j} \cdot \boldsymbol{n}}\hat{i}_{l_j} - i_{r_j}\right)^2\right]\,, \tag{6.12}$$

where the variables to optimise are $\boldsymbol{n}$ and $\{\hat{i}_{l_j}\}_j$. This is *a priori* a complex minimisation problem involving $3 + N$ unknowns, where N is the number of reciprocal pairs of images.

It is shown in appendix E that this minimisation problem can be simplified to the search for the surface normal $\boldsymbol{n}$ which minimises the following cost function:

$$\sum_j \frac{[(i_{l_j}\boldsymbol{s}_{l_j} - i_{r_j}\boldsymbol{s}_{r_j}) \cdot \boldsymbol{n}]^2}{(\boldsymbol{s}_{l_j} \cdot \boldsymbol{n})^2 + (\boldsymbol{s}_{r_j} \cdot \boldsymbol{n})^2}\,. \tag{6.13}$$

By analogy with the previous section, we re-write the cost function in the form

$$\sum_j d_{\text{rad}}(\boldsymbol{n}, i_{l_j}, i_{r_j}, \boldsymbol{s}_{l_j}, \boldsymbol{s}_{r_j})^2\,, \tag{6.14}$$

where d_{rad} denotes the radiometric distance associated with a pair of reciprocal measurements i_{l_j} and i_{r_j} and a normal $\boldsymbol{n}$, which is defined by

$$d_{\text{rad}}(\boldsymbol{n}, i_{l_j}, i_{r_j}, \boldsymbol{s}_{l_j}, \boldsymbol{s}_{r_j})^2 = \frac{[(i_{l_j}\boldsymbol{s}_{l_j} - i_{r_j}\boldsymbol{s}_{r_j}) \cdot \boldsymbol{n}]^2}{(\boldsymbol{s}_{l_j} \cdot \boldsymbol{n})^2 + (\boldsymbol{s}_{r_j} \cdot \boldsymbol{n})^2}\,. \tag{6.15}$$

This is a simple minimisation problem with only two degrees of freedom ($\|\boldsymbol{n}\| = 1$). A visibility constraint must also be enforced. This constraint states that $\boldsymbol{w}_l \cdot \boldsymbol{n} > 0$ and $\boldsymbol{w}_r \cdot \boldsymbol{n} > 0$. Note that this does not take into account self-occlusions. It is possible to enforce this constraint during minimisation of the cost function, however it is simpler and usually sufficient to verify that the visibility constraint is satisfied after convergence of the search algorithm. Any non-linear iterative method can be used to carry out the optimisation, such as for example the Levenberg-Marquardt (LM) algorithm. The search can be initialised for example with the results of the SVD solution, or even by choosing an arbitrary normal satisfying the visibility constraint.

6.3.2 Comparison with the algebraic distance

From Eq. (6.9) and Eq. (6.15), it results that for a given reciprocal pair of images:

$$d_{\text{rad}}(\boldsymbol{n}, i_{l_j}, i_{r_j}, \boldsymbol{s}_{l_j}, \boldsymbol{s}_{r_j})^2 = \frac{1}{(\boldsymbol{s}_{l_j} \cdot \boldsymbol{n})^2 + (\boldsymbol{s}_{r_j} \cdot \boldsymbol{n})^2} d_{\text{alg}}(\boldsymbol{n}, i_{l_j}, i_{r_j}, \boldsymbol{s}_{l_j}, \boldsymbol{s}_{r_j})^2 \,. \tag{6.16}$$

It is difficult to give a simple interpretation of the multiplicative factor relating the two measures. However, it becomes apparent that the discrepancy between the two constraints is due *only* to the positions of the camera and light source relatively to the surface point, and does not depend on the surface albedo or reflectance property. This does not mean that the accuracy of the normal estimation does not depend on the latter properties, it probably does, however the relative performance of the two measures does not. In practice, this implies that the two measures are equivalent for surface patches equidistant from all camera and light sources and whose normal bisect all reciprocal pairs. This is approximately the case of horizontal surfaces located at the centre of the turntable in our experimental set up (see Section 6.5.2).

6.3.3 Maximum Likelihood estimate

We now justify statistically the cost function based on the radiometric distance which is defined in Eq. (6.14), and show its minimisation provides a Maximum Likelihood (ML) estimate of the surface normal under standard Gaussian noise assumptions. The demonstration is similar to the one given in [72] (pp 86–88) in the case of geometric distance for homography estimation.

We assume that the intensity measurement error follows a Gaussian distribution with zero mean and standard deviation σ at each pixel, and that these measurements are independent. Under this assumption, the Probability Density Function (PDF) of each measurement pixel intensity i is:

$$P(i) = \frac{1}{2\pi\sigma^2} \mathrm{e}^{-\frac{1}{2\sigma^2}(\bar{i}-i)^2} \,, \tag{6.17}$$

where $\bar{i}$ denotes the true intensity at the pixel considered. The true intensity values in the right image $\{\bar{i}_{r_j}\}_j$ are related to the true intensity values in the left image $\{\bar{i}_{l_j}\}_j$ by the true surface normal $\boldsymbol{n}$, such that

$$\bar{i}_{r_j} = \frac{\boldsymbol{s}_{l_j} \cdot \boldsymbol{n}}{\boldsymbol{s}_{r_j} \cdot \boldsymbol{n}} \bar{i}_{l_j} \,. \tag{6.18}$$

Therefore the PDF of the measurements given the true surface normal $\boldsymbol{n}$ and the true intensity values $\{\bar{i}_{l_j}\}_j$ in the left image is:

$$P(\{i_{l_j}, i_{r_j}\}_j | \boldsymbol{n}, \{\bar{i}_{l_j}\}_j) = \prod_j \frac{1}{2\pi\sigma^2} \mathrm{e}^{-\frac{1}{2\sigma^2}\left[(\bar{i}_{l_j} - i_{l_j})^2 + \left(\frac{\boldsymbol{s}_{l_j}\cdot\boldsymbol{n}}{\boldsymbol{s}_{r_j}\cdot\boldsymbol{n}}\bar{i}_{l_j} - i_{r_j}\right)^2\right]}. \tag{6.19}$$

We assume that the errors in the determination of $\boldsymbol{s}_{l_j}$ and $\boldsymbol{s}_{r_j}$ are negligible compared to the intensity measurement error; this is a reasonable assumption if the geometric calibration of the camera is very accurate, and the surface points can be localised accurately (this may require a very dense sampling of the 3D space in practice). If we write the log-likelihood, we obtain:

$$\log P(\{i_{l_j}, i_{r_j}\}_j | \boldsymbol{n}, \{\bar{i}_{l_j}\}_j) = -\frac{1}{2\sigma^2}\sum_j \left[(\bar{i}_{l_j} - i_{l_j})^2 + \left(\frac{\boldsymbol{s}_{l_j}\cdot\boldsymbol{n}}{\boldsymbol{s}_{r_j}\cdot\boldsymbol{n}}\bar{i}_{l_j} - i_{r_j}\right)^2\right] + N\log(\frac{1}{2\pi\sigma^2}). \tag{6.20}$$

The ML estimate of the surface normal $\boldsymbol{n}$ and the image intensities in the left image $\{\hat{i}_{l_j}\}_j$ are the values which maximise this log-likelihood. As for the normal estimation, this is equivalent to minimising the cost function defined in Eq. (6.12).

6.4 Treatment of image saturation

It has been mentioned earlier that one of the outstanding features of HS is that it does not make any assumption about the surface reflectance properties of the object. Specularities, which are traditionally problematic with the majority of image-based reconstruction algorithm, actually become features which help resolve the matching problem in the case of HS. Nevertheless, from a practical point of view, it is not always possible to capture all specularities due to the limited range of the camera sensor. This may result in some saturations of the pixel intensity measured, which corrupts the constraints because the intensities considered are not the ones physically expected. So far, very little attention has been given to this problem. Among all publications in the field of HS, only Tu and Mendonça reported a solution in [156] in the case of a binocular implementation. We propose a similar treatment of image saturation in the multi-ocular case, and adapt consequently the radiometric distance defined earlier. Even though image saturation is usually a relatively localised phenomenon in the image, if ignored it can result in some artefacts in the reconstruction.

The idea is very simple. When a saturation is observed in a reciprocal pair of images (usually the saturation is observed simultaneously in both images at reciprocal positions), it means that the normal approximately bisects the incident and emerging rays. We express this by the constraint:

$$(\boldsymbol{v}_l - \boldsymbol{v}_r) \cdot \boldsymbol{n} = 0\,. \tag{6.21}$$

This constraint replaces Eq. (6.3) when saturation is observed. In this case, the appropriate rows of W in Eq. (6.4) must be replaced by $(\boldsymbol{v}_l - \boldsymbol{v}_r)^\top$. As for the radiometric constraint defined in Eq. (6.15), it must be replaced by:

$$d_{\text{rad}}(\boldsymbol{n}, i_{l_j}, i_{r_j}, \boldsymbol{s}_{l_j}, \boldsymbol{s}_{r_j})^2 = \left[\left(\frac{\boldsymbol{s}_{l_j}}{\|\boldsymbol{s}_{l_j}\|} - \frac{\boldsymbol{s}_{r_j}}{\|\boldsymbol{s}_{r_j}\|}\right) \cdot \boldsymbol{n}\right]^2 . \tag{6.22}$$

This distance is similar to the cost function defined by Tu and Mendonça in [156] in the case of binocular HS.

6.5 Results

6.5.1 Synthetic data

The aim of the experiment is to compare the accuracy of the methods based on algebraic and radiometric distances, for surface normal estimation. In order to test the accuracy of the surface normal estimation independently of the correspondence problem, it is assumed that correspondences are known in advance. Two implementations are considered for the algebraic distance. The first defines rows for the matrix W as stated in Eq. (6.5) and is called *unnormalised algebraic*, while the second implementation normalises the rows to unit values and is called *normalised algebraic*. The issue of normalisation has been discussed earlier when treating the reconstruction problem.

A planar and uniform surface patch with ground truth normal $\boldsymbol{n}$ is generated randomly. N pairs of points $\boldsymbol{O}_l$ and $\boldsymbol{O}_r$ are generated randomly; they define N reciprocal pairs of images (here radiance values) of the surface patch. The points are constrained to be located on the same side of the patch such that the visibility constraint is satisfied for all reciprocal pairs. The distance from the points to the patch is also selected randomly within the interval $[\epsilon, 1]$ m where $\epsilon = 10^{-3}$ m, in order to avoid the configuration where the camera or light source is located on

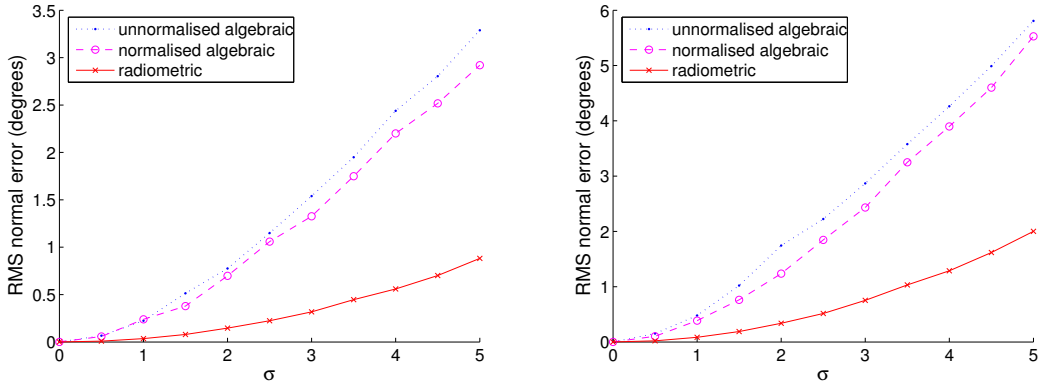
the surface point. In the implementation, the random positions are obtained by generating points with random spherical coordinates (r, θ, ϕ) in the respective intervals $[\epsilon, 1]$, $[0, \pi/2]$ rad and $[0, 2\pi]$ rad; in this parametrisation r, θ and ϕ denote respectively the radial, azimuth, and zenith coordinates. The radiance values generated are perturbed by a zero-mean Gaussian noise with standard deviation σ. The strength of the light source is constant and equal to $\kappa = 1,000$.

The BRDF of the surface patch has been modeled by the modified Phong reflectance model which is described in [87, 83]. It consists of the sum of a diffuse part and a specular part. We follow the formalism adopted in [83], and define:

$$f_{n,k_d,k_s}(\boldsymbol{X}, \boldsymbol{v}_l, \boldsymbol{v}_r) = k_d \frac{1}{\pi} + k_s \frac{n+2}{2\pi} \cos^n \alpha \,, \tag{6.23}$$

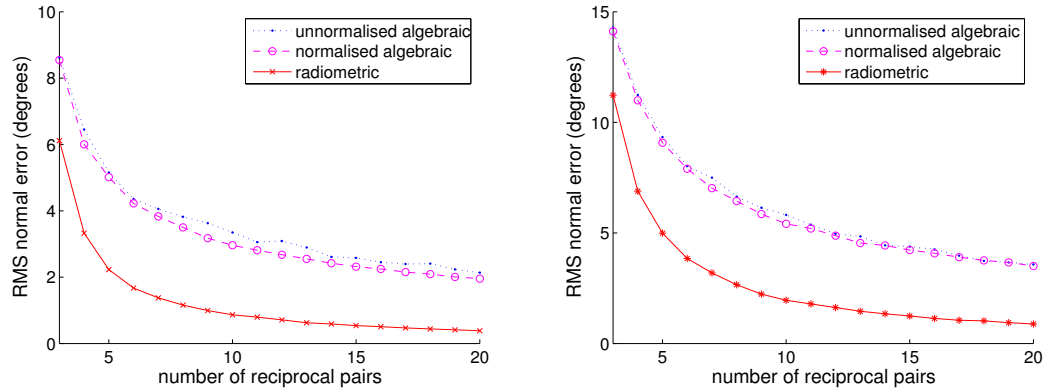
where α denotes the angle between the perfect specular reflective direction and the emerging direction. This model has three parameters n, k_d and k_s, which represent respectively the specular exponent (large values results in sharper specular reflections), the diffuse reflectivity and the specular reflectivity. We considered two different settings. The first one corresponds to the values $n = 40$, $k_d = 0.4$ and $k_s = 0.05$, while the second one corresponds to the values $n = 1$, $k_d = 1$ and $k_s = 0$. The second settings corresponds to a Lambertian model. The advantage of this model over the original model described in [108] is that it is physically plausible, *i.e.*, it produces BRDF values which do not violate the laws of physics, in particular the reciprocity principle on which HS is based. More complex physical-based models such as the Torrance and Sparrow model [36] could have been considered. However we found it sufficient to limit ourselves to this model. There are two main reasons for doing this. Firstly, the Phong model is the most commonly used shader in computer graphics. Secondly, it has been observed from Eq. (6.16) that the discrepancy between the two distances is not related to the BRDF, therefore in theory the relative performance of the two methods is expected to be similar regardless of the choice of BRDF model.

Two sets of experiments were carried out. In the first set, we consider a fixed number of 10 reciprocal pairs of images selected randomly as described earlier, and vary the standard deviation of the noise added to the measured radiance between 0 and 5, in order to study the influence of the noise level. The experiment is repeated 10,000 times, and the Root Mean Squared (RMS) angular error between the estimated normal and the true normal is computed for each method. The RMS angular error between the set of estimated normals $\hat{\boldsymbol{n}}_{ij}$ and true normals $\bar{\boldsymbol{n}}_{ij}$ is de-



(a) Lambertian (b) Phong

Figure 6.3: *Influence of the standard deviation of the Gaussian additive image noise on the accuracy of the normal reconstruction. 10 pairs of reciprocal image pairs were considered in all experiments. (a) shows the results in the case of a Lambertian surface, while (b) considers a Phong reflectance model with parameters* $k_d = 0.4$, $k_s = 0.05$ *and* $n = 40$. *RMS values computed from 10,000 experiments.*



(a) Lambertian (b) Phong

Figure 6.4: *Influence of the number of reciprocal image pairs considered on the accuracy of the normal reconstruction. The standard deviation of the Gaussian additive image noise is* $\sigma = 5$ *in all experiments. (a) shows the results in the case of a Lambertian surface, while (b) considers a Phong reflectance model with parameters* $k_d = 0.4$, $k_s = 0.05$ *and* $n = 40$. *RMS values computed from 10,000 experiments.*

fined by $\sqrt{\frac{1}{N}\sum_i \sum_j \theta(\bar{\boldsymbol{n}}_{ij}, \hat{\boldsymbol{n}}_{ij})^2}$, where $\theta(\bar{\boldsymbol{n}}_{ij}, \hat{\boldsymbol{n}}_{ij})$ denotes the angles between the vectors $\bar{\boldsymbol{n}}_{ij}$ and $\hat{\boldsymbol{n}}_{ij}$. The results can be found in Fig. 6.3. In the second set of experiments, the noise level is constant and equal to $\sigma = 5$, and the number of reciprocal pairs considered is allowed to vary between 3 (minimum number supported by the method) and 20. Again the experiment is repeated 10,000 times, and the Root Mean Squared (RMS) angular error between the estimated normal and the true normal is computed for each method. The results can be found in Fig. 6.4. It appears that in both cases, the method based on the radiometric distance is much more accurate than the methods based on the algebraic distance. The normalised implementation of the method based on the algebraic distance seems to be slightly more accurate than the unnormalised version, however the improvement is not very significant. The fact that the method based on the radiometric distance is the most accurate does not come as a surprise. It is supported by the fact that this method corresponds to the ML estimator.

6.5.2 Real data

The experimental setup (see illustration in Fig. 6.5) consists of a camera, a light source and a turn-table which performs the interchange of camera and light source positions. The camera and light source are positioned symmetrically with respect to the axis of rotation of the turntable, such that rotating the turntable by 180° is equivalent to interchanging camera and light source positions. Inaccuracies in the positioning typically introduce some errors in the measurements because the pairs of images acquired do not correspond exactly to reciprocal configurations. A 12 bit digital camera Vosskuhler CCD-1300 equipped with a 25 mm lens was used along with a halogen lamp equipped with a diaphragm and acting as a point light source. The resolution of the images produced by the camera is 1024×1280 pixels. The distance between the camera and the centre of the table is approximately 80 cm and the distance between the camera and the light source 60 cm. The geometric calibration of the camera was carried out by grabbing three images of the same planar calibration grid translated by known increments, thus forming a 3D calibration object. A standard calibration method based on SVD, followed by lens distortion calibration, has been used. We did not consider more sophisticated methods such as the ones described in the first part of the thesis because of the particular experimental setup constraints.

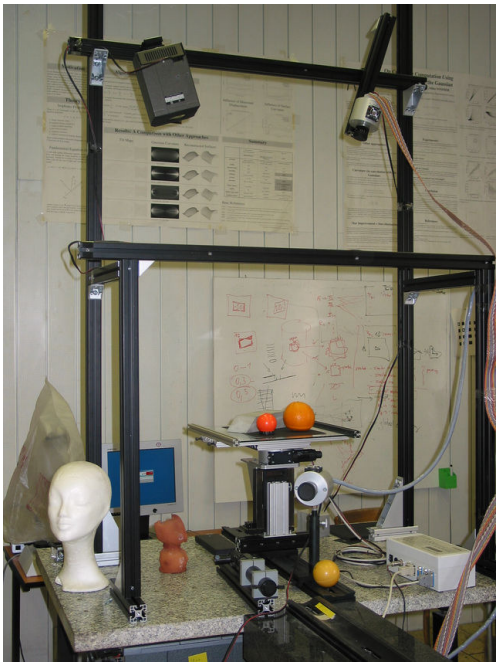


Figure 6.5: *Experimental setup used for reconstruction.*

Experiments were carried out with four different objects: a snooker ball (Fig. 6.7), two types of teapots (Fig. 6.8 and 6.9) and a doll (Fig. 6.10). The first three objects have specular surfaces, while the last object seems to be approximately Lambertian. The reconstruction procedure is the same for each object. Eight reciprocal pairs of images are generated by rotating the turntable by regular 22.5° increments. A set of images is shown for one of the objects in Fig. 6.6. A bounding box is defined for each object, in order to restrict the search for correspondences. The bounding box is discretised into square voxels of resolution $1\,\text{mm} \times 1\,\text{mm} \times 1\,\text{mm}$ in the case of the snooker ball and the doll, and $2\,\text{mm} \times 2\,\text{mm} \times 2\,\text{mm}$ for the teapots which are larger. The size of the window used during depth search is set to 5×5 pixels for all objects. In addition, some thresholding of the input images has been done before processing in order to eliminate the background. Such segmentation is rather crude and is usually not able to eliminate all background pixels; this is not a problem because the remaining points will be discarded automatically during reconstruction if they produce inconsistent measurements.

HS outputs two cues which can be used for reconstruction: the depth and the normal at each surface point. The depth map is shown in (b) of each figure; points are represented with a brightness proportional to the scene depth. The depth maps seem generally rather noisy and

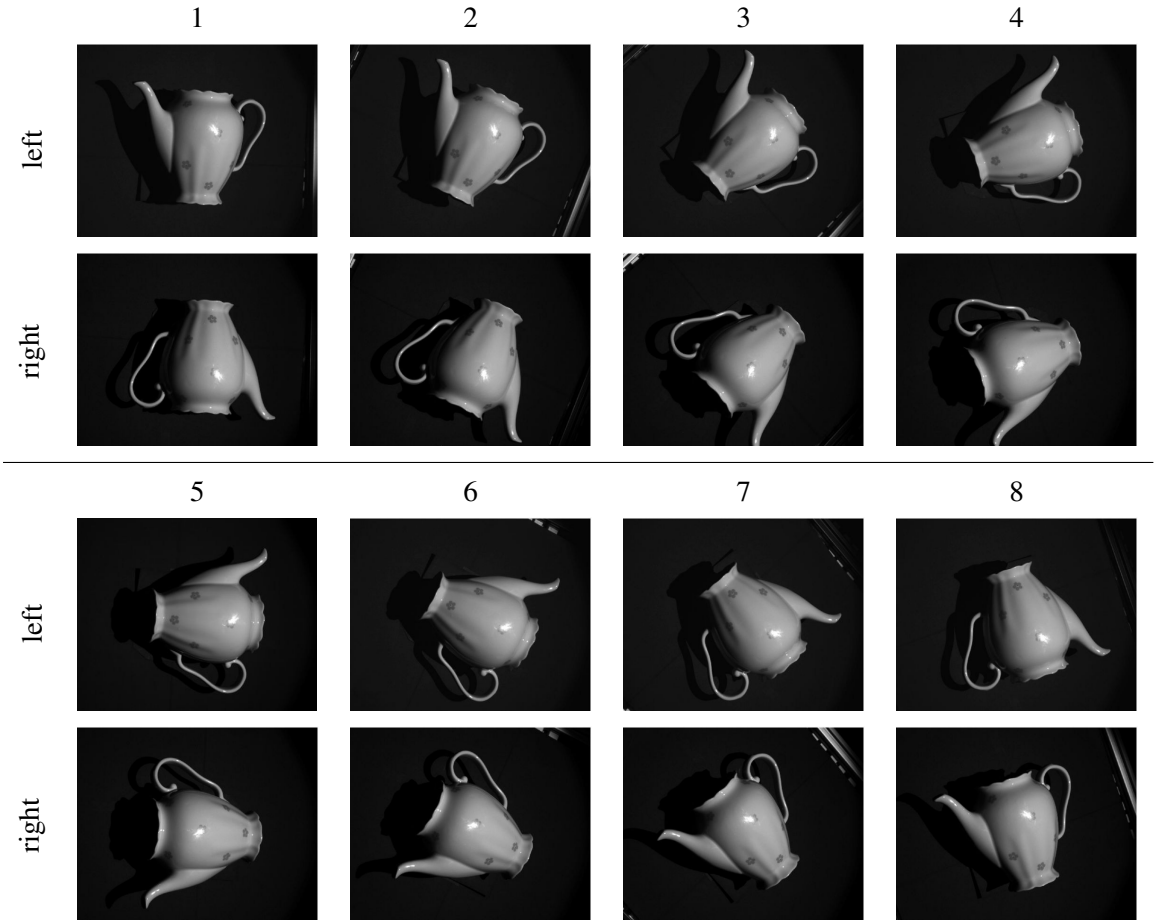


Figure 6.6: *The eight reciprocal pairs of images considered in the case of the object 'Teapot 1'.*

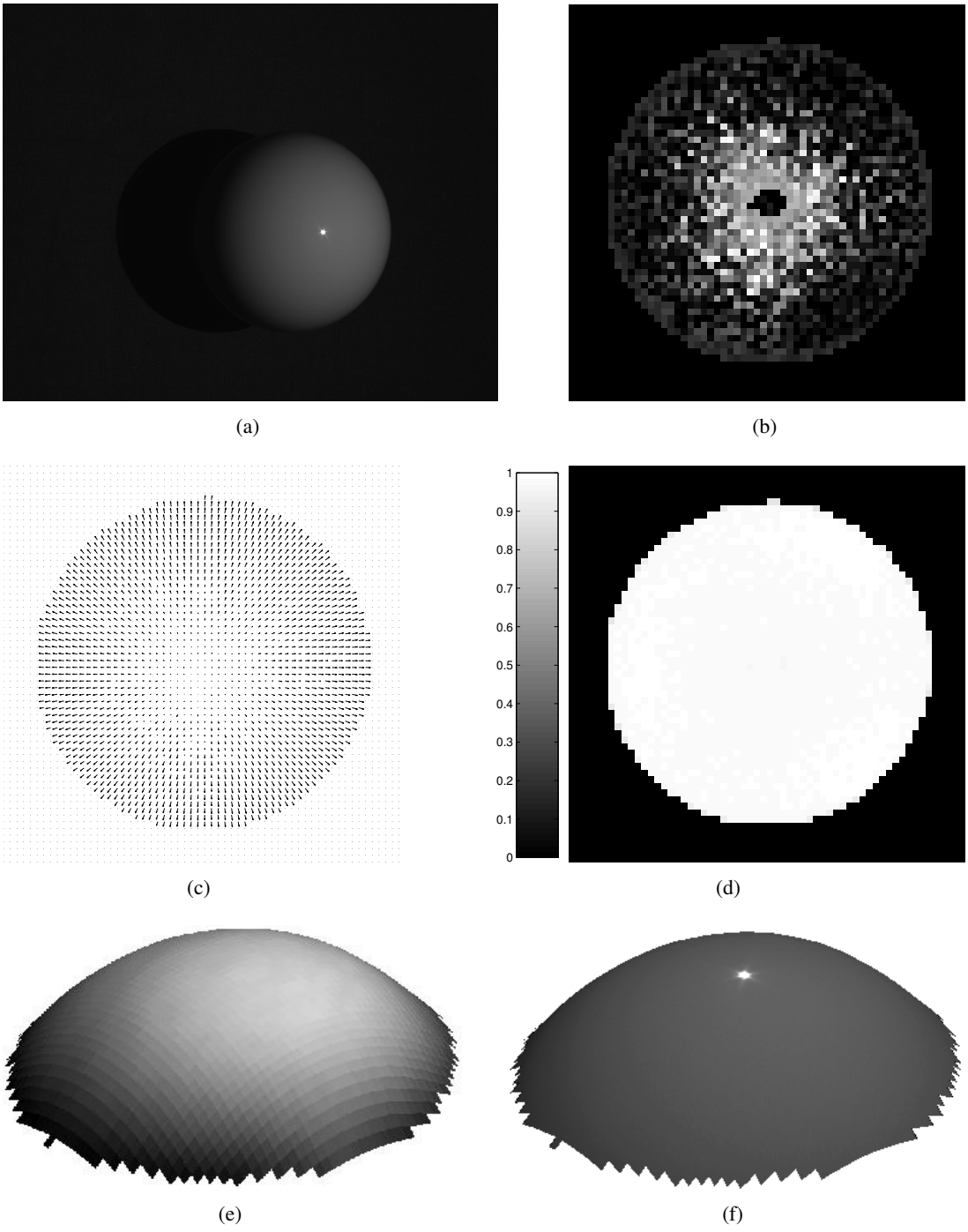


Figure 6.7: *Reconstruction of the object 'Snooker ball'. (a) shows one of the input images. (b) represents the depth map, (c) the normal field and (d) the support measure. (e) shows the 3D model obtained from integration of the normal field. (f) shows the same model with mapped texture.*

inaccurate. This is due to the fact that the matching constraint provided by imposing Helmholtz reciprocity defines only a necessary condition for finding correspondences. Even though a smoothness constraint has been enforced by maximising the support measure summed over a 5×5 window during depth search, this is not always sufficient to resolve totally the ambiguity. The use of larger windows would have resulted in less noisy depth map, but this would penalise the reconstruction of sharper surface variations, because of the low pass filtering effect of the smoothing. There is obviously a trade-off between filtering out the noise and preserving the surface variations. A needle map representation of the normal field is given in (c) of each figure. The normal was computed by minimising the cost function based on the radiometric distance. It can be observed that the normal field seems to have preserved the surface variations better than the depth map. This is because no smoothness assumption has been made at this stage. Naturally the normal field will be chosen as the main cue for inferring the 3D models of the objects.

Before presenting the final 3D models, a last intermediate result useful for reconstruction is presented in (d) of each figure. The result in question is the support measure associated with each normal. The support measure defined in Eq. (6.6) takes values between 0 and 1, the value of 1 representing the highest level of confidence. As expected, it can be observed that points located on the object surface are associated with high support measure (very close to 1), while background points have a very low support measure (close to 0). Note that the low support measure for the background is due to the threshold imposed earlier during background segmentation. From a theoretical point of view, background points could be reconstructed by the algorithm, and it may not be necessary to eliminate them. The problem is that there exist many occlusions/self-shadows at the vicinity of the object, which complicate considerably the reconstruction process. Even though methods for the detection of occlusions have been reported to be applicable in this case [178], this is not so straightforward to implement in practice, and no implementation has yet been reported in the literature. The ultimate goal is to produce a 3D model of the object. Such a model is obtained by integration of the normal field using the method reported in [145]. In our implementation, the support measure was treated as a confidence value for each normal, and was used to weight the associated normal during integration. The 3D model obtained is shown in (e) of each figure. (f) shows the same 3D model mapped with the texture from one of the input images.

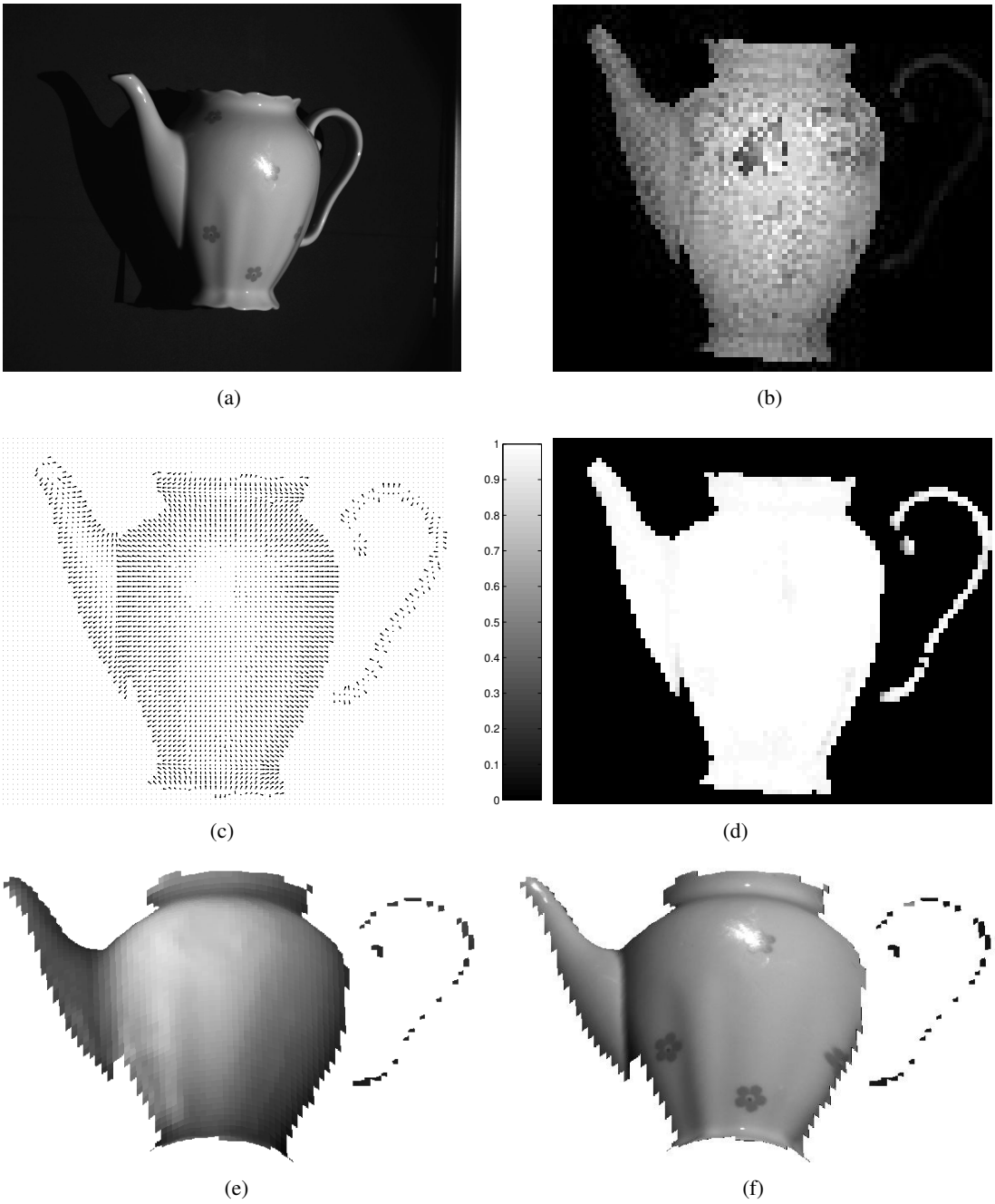


Figure 6.8: *Reconstruction of the object 'Teapot 1'. (a) shows one of the input images. (b) represents the depth map, (c) the normal field and (d) the support measure. (e) shows the 3D model obtained from integration of the normal field. (f) shows the same model with mapped texture.*

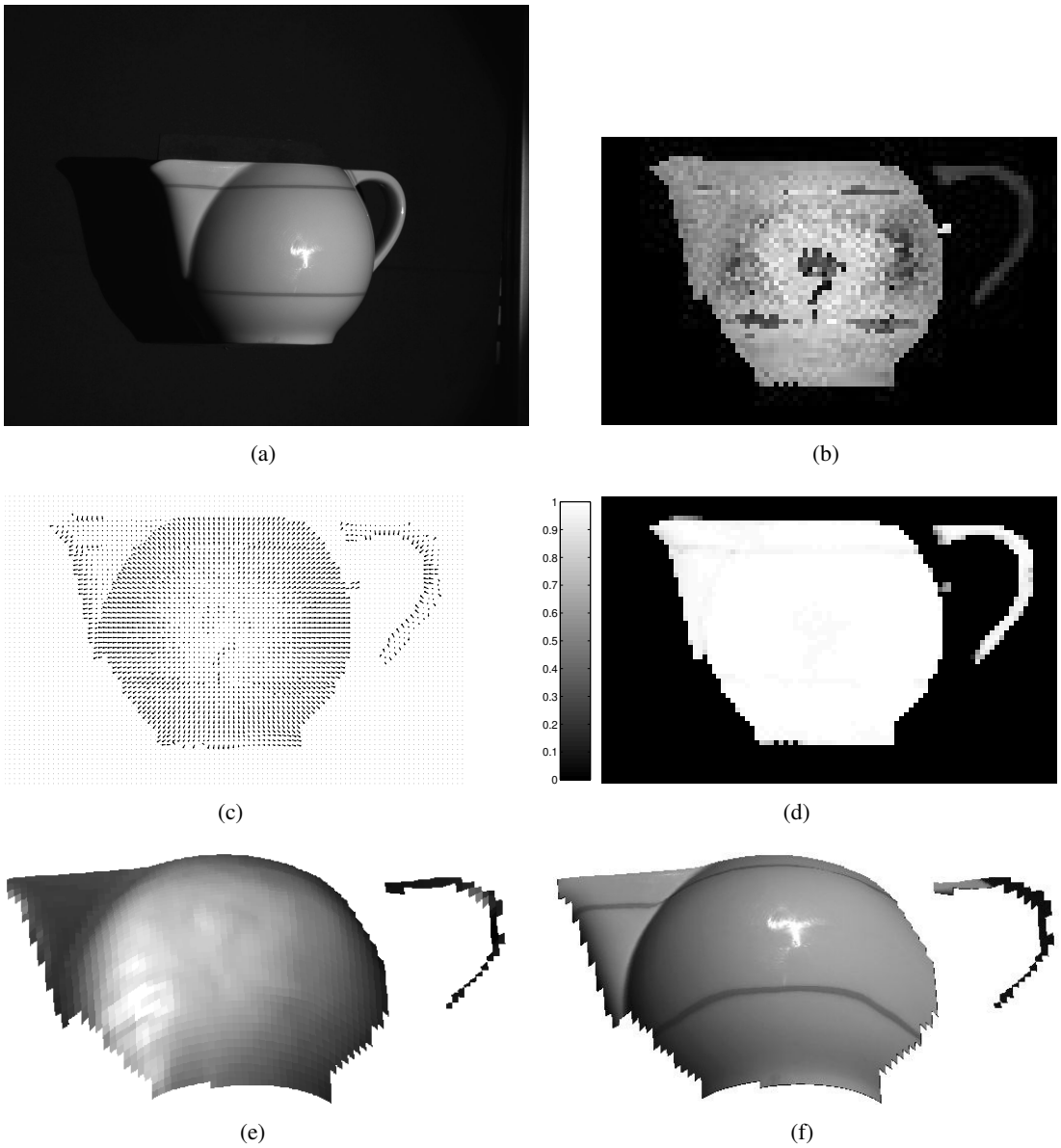


Figure 6.9: *Reconstruction of the object 'Teapot 2'. (a) shows one of the input images. (b) represents the depth map, (c) the normal field and (d) the support measure. (e) shows the 3D model obtained from integration of the normal field. (f) shows the same model with mapped texture.*

Experiments were carried out with both the algebraic and radiometric distances. However, the qualitative comparison carried out (in particular on the snooker ball for which the shape is known *a priori*) did not show any immediately visible improvement due to the use of the radiometric distance. For this reason, only results with the radiometric method have been reported here. It would have been interesting to measure quantitatively the improvement obtained by considering the radiometric constraint, however no ground truth was available for that. We believe that the similarity of performance of the two measures is due to the restriction in camera and light source placements relatively to the object which are imposed by the experimental setup. Indeed, given the relatively small size of the objects compared to the distance separating them from the camera and light source, this distance can be considered as approximately constant. Also, because of the visibility constraint imposed, there is not much scope for incident and emerging angle variations. The constancy of these terms mean that the multiplicative factors appearing in Eq. (6.16) do not exhibit large variations which normally penalise the algebraic distance. It results that for this particular set-up, the algebraic and radiometric distances are nearly equivalent. The benefit of using the radiometric distance is expected to be larger for setups allowing more flexibility in camera and light source placements.

The first three objects are particularly challenging to reconstruct because the surfaces are highly specular. HS seems to be performing very well on these objects as well as on the simpler Lambertian object. The reconstruction appears very smooth and visually correct. Only a few very small artefacts are visible at the location of the specularities for the first three objects (visible in the depth map, the normal field and also to a lesser extent in the final 3D model). These are caused by saturations due to specularities which are not as localised as expected in theory, and introduce a slight bias in the surface normal estimate. The extended saturations observed are due to the fact that the point light source assumption is not exactly satisfied in practice, and also because the surface is not an ideal specular reflector (mirror surface). The phenomenon remains very localised however, and does not affect very much the reconstruction because saturation does not usually occur simultaneously in all reciprocal pairs (except for horizontal surfaces located on the axis of rotation of the turntable).

One limitation of our implementation is that the reconstruction is restricted to object surfaces which are simultaneously visible in all reciprocal pairs of images. For example, only the top part of the snooker ball has been reconstructed, also the handle of the teapots appear as a

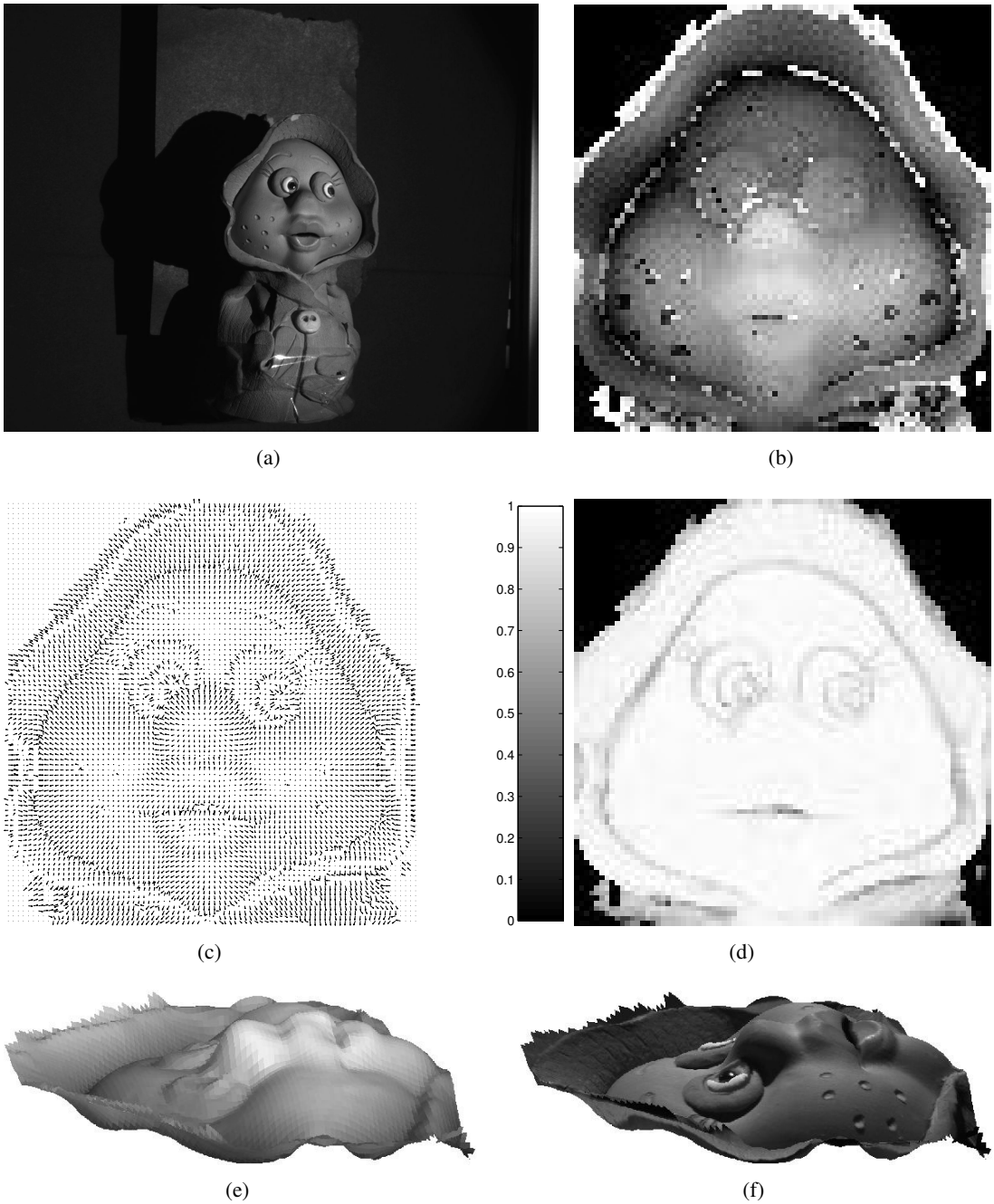


Figure 6.10: *Reconstruction of the object 'Doll'. (a) shows one of the input images. (b) represents the depth map, (c) the normal field and (d) the support measure. (e) shows the 3D model obtained from integration of the normal field. (f) shows the same model with mapped texture.*

separate components not connected to the rest of the reconstruction. This is mainly attributed to our experimental setup which does not allow arbitrary camera and light source placements. A more flexible implementation would allow more general camera and light source placements. The problem with such an implementation is that there exist many occlusions that corrupt the reconstruction, and an efficient mechanism to discard them is needed. This can done rather simply by observing that a point is occluded in one image if it is in a shadowed area in the reciprocal image, therefore occlusion detection is simplified to shadow detection. In such an implementation, any point visible in at least three reciprocal pairs of images (the minimum requirement to compute the support measure) could be reconstructed. Finally, it can be noticed in the reconstruction of the doll, that some outliers are present at the object boundary. These are due to some occlusions. These may have been eliminated if a more efficient background segmentation had been applied, or alternatively if a segmentation had been carried out later on on support measure values (it can be observed that outliers have lower support measure than surface points).

6.6 Conclusions

In this chapter, we concentrated on the reconstruction problem in the case of HS. A novel method for surface normal reconstruction has been presented. The method is based on the minimisation of a cost function which consists of squared radiometric distances summed over all reciprocal pair of images in which the surface is visible. Physically, the cost function represents the modification to be applied to the intensities of the projection of surface points in order to satisfy exactly the Helmholtz reciprocity principle. The normal found by this method has been shown to be a ML estimate under standard Gaussian assumption. Such a solution can be computed at low computational cost because of the small number of optimisation variables involved. The case of image saturations due to specularities has also been considered and successfully integrated in our reconstruction algorithm.

In the case of synthetic data, it has been verified experimentally that the radiometric cost function results in a significant improvement in the accuracy of the normal estimation compared to the algebraic method based on SVD. Experiments carried out with real data showed that the method is able to produce realistic 3D models of a variety of objects which are a priori difficult

to reconstruct because of their surface properties, however the improvement resulting from the use of the method has not been quantified because of the absence of ground truth for these objects.

The radiometric distance offers an optimum solution to the surface normal estimation problem, however the correspondence problem still relies on the use of algebraic solution provided by SVD. While it appears to be sufficiently accurate, we believe that there exists scope for improvement in solving the correspondence problem in a more efficient manner, and hope our work on surface normal estimation will inspire the development of similar methods in this field.

Chapter 7

Generalisation of Helmholtz Stereopsis to rough and textured surfaces

7.1 Introduction

We continue the work on Helmholtz Stereopsis (HS). In this chapter, we concentrate on extending the class of surfaces to which the method is applicable. More specifically, we generalise the method to the reconstruction of textured and rough surfaces. All implementations of HS presented so far [92, 177, 178, 179, 156, 180] considered the reconstruction of smooth uniform surfaces. This was also the case of the objects reconstructed in the previous chapter of this thesis. In reality, however, many objects do not satisfy this assumption. We distinguish two main classes of common objects which violate this assumption: i) rough objects (*i.e.* locally non-convex) and ii) textured objects. We argue that the standard version of HS which constructs constraints based on single pixel measurements in images can fail on such objects. In the case of textured surfaces, the violation is due to the high frequency variations of the surface scattering properties which cannot be captured by the finite sensor elements. In the case of rough surfaces, the constraint is corrupted by inter-reflections occurring within the non-convex geometry of the surface. In both cases, the reasons for the violation of the constraint are intimately related to the definition of the Bidirectional Reflectance Distribution Function (BRDF).

This chapter is structured as follows. We start by analysing the physical reasons for the failure

of the standard HS constraint on both types of surfaces. Then a novel method which is able to produce correct unbiased constraints for both types of surfaces is proposed - the definition is supported by recent work in the field of physics and remote sensing. It is important to note that the solution proposed here addresses only the problem of local inter-reflections which are encountered for example in rough objects. Global inter-reflections can occur at a large scale and are usually very difficult to take into consideration. This is validated on a simple test object. Then the implementation in the context of HS is presented. Finally some experimental results with real objects are presented, as well as a comparison with the standard HS which uses raw pixel measurements.

7.2 Problem with rough and strongly textured surfaces

In this section, we explain and illustrate on elementary examples that the HS constraint defined in Eq. (6.3) is affected by the presence of texture and inter-reflections occurring in rough surfaces when single pixel measurements are used to construct it.

7.2.1 Original Helmholtz Stereopsis constraint formulation

The HS constraint expressed in Eq. (6.3), results directly from the reciprocity of the BRDF (see derivation in previous chapter). Therefore the question of the validity of the constraint concerns actually the validity of the reciprocity of the BRDF associated with the intensity measurements. In order to clarify this aspect, it is necessary to go back to the original definition of the BRDF.

The BRDF was originally defined by Nicodemus *et al.* in [104] as a means of characterising the geometric reflecting properties of a surface. Let us consider a surface point $\boldsymbol{x}_r$ (see Fig. 7.1). In order to avoid dealing with microscopic representations, which complicate considerably the parametrisation of the problem, the surface is represented locally by a reference plane. It is assumed that a relatively large area $\mathcal{A}_i$ of the surface is illuminated along the direction represented by the vector $\boldsymbol{v}_i$ by a well collimated beam with uniform irradiance $\mathrm{d}E_i(\boldsymbol{v}_i)$, and that the point $\boldsymbol{x}_r$ is located well within the area $\mathcal{A}_i$. Let us denote by $\mathrm{d}L_r(\boldsymbol{x}_r, \boldsymbol{v}_r)$ the resulting radiance reflected at the point $\boldsymbol{x}_r$ in the direction represented by the vector $\boldsymbol{v}_r$. Because of some physical phenomena occurring at the surface of the material, the radiance emanating

from this point can be considered as the sum of contributions of elements of area located in the neighbourhood of the point $\boldsymbol{x}_r$ - unless the surface is perfectly smooth and opaque in which case only the point $\boldsymbol{x}_r$ contributes. In order to take into account these phenomena, the area $\mathcal{A}_i$ is chosen large enough such that all points susceptible to contribute to the reflected radiance $\mathrm{d}L_r(\boldsymbol{x}_r, \boldsymbol{v}_r)$ are included. The Bidirectional Reflectance Distribution Function (BRDF) is then defined as the ratio of the outgoing radiance to the incoming irradiance, *i.e.*:

$$f_r(\boldsymbol{x}_r, \boldsymbol{v}_i, \boldsymbol{v}_r) = \frac{\mathrm{d}L_r(\boldsymbol{x}_r, \boldsymbol{v}_r)}{\mathrm{d}E_i(\boldsymbol{v}_i)} \,. \tag{7.1}$$

It is clear from the definition that this is a purely theoretical concept involving infinitesimal elements which cannot be measured in reality. In particular, the reflected radiance should be confined within a solid angle element, if exact BRDF measurements were to be made.

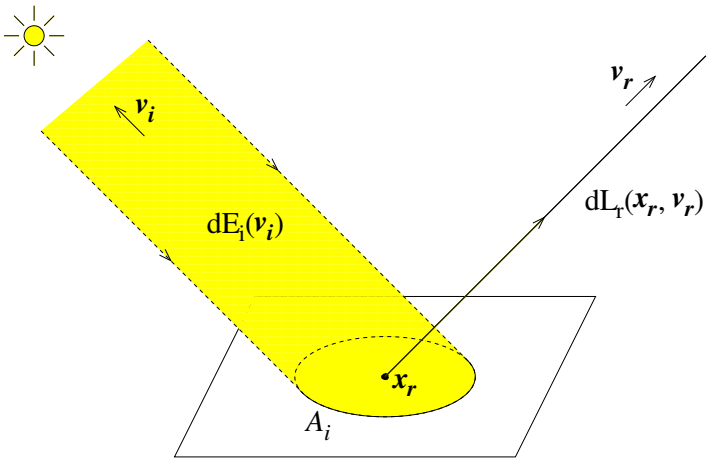


Figure 7.1: *Reflectance geometry for theoretical definition of BRDF.*

In practice, a major source of limitation is due to the resolution of the sensor. The consequence is that radiance measurements are actually average values of the radiance emanating from the surface area corresponding to the projection of the sensor element (see Fig. 7.2). If we denote by $\mathcal{A}_r$ the area of the projection of the sensor element observing $\boldsymbol{x}_r$ onto the surface (actually its reference plane), the actual BRDF measured can be expressed mathematically as:

$$f_{\text{sensor}}(\boldsymbol{x}_r, \boldsymbol{v}_i, \boldsymbol{v}_r) = \frac{\frac{1}{\mathcal{A}_r}\int_{\boldsymbol{x}_r \in \mathcal{A}_r} \mathrm{d}L_r(\boldsymbol{x}_r, \boldsymbol{v}_r)}{\mathrm{d}E_i(\boldsymbol{v}_i)} = \frac{\overline{\mathrm{d}L_r(\boldsymbol{x}_r, \boldsymbol{v}_r)}}{\mathrm{d}E_i(\boldsymbol{v}_i)} \,, \tag{7.2}$$

where the bar symbol over a variable denotes its mean value. In the previous HS constraint formulated in Eq. (6.3), the image brightness measured at single pixel locations and denoted

by i_l or i_r, actually correspond to such average values $\overline{\mathrm{d}L_r(\boldsymbol{x}_r, \boldsymbol{v}_r)}$. In the case of smooth uniform (*i.e.* non-textured at the scale of observation) surfaces, the scattering properties of the surface can be considered statistically uniform and isotropic across the reference plane. It results that the reflected radiance $\mathrm{d}L_r(\boldsymbol{x}_r, \boldsymbol{v}_r)$ is approximately constant over the area $\mathcal{A}_r$ covered by the sensor, and therefore $\overline{\mathrm{d}L_r(\boldsymbol{x}_r, \boldsymbol{v}_r)} \approx \mathrm{d}L_r(\boldsymbol{x}_r, \boldsymbol{v}_r)$. Thus in the case of such surfaces, Helmholtz reciprocity is satisfied and the constraint in Eq. (6.3) is valid. We explain next the reasons why this is usually not the case with rough or textured objects.

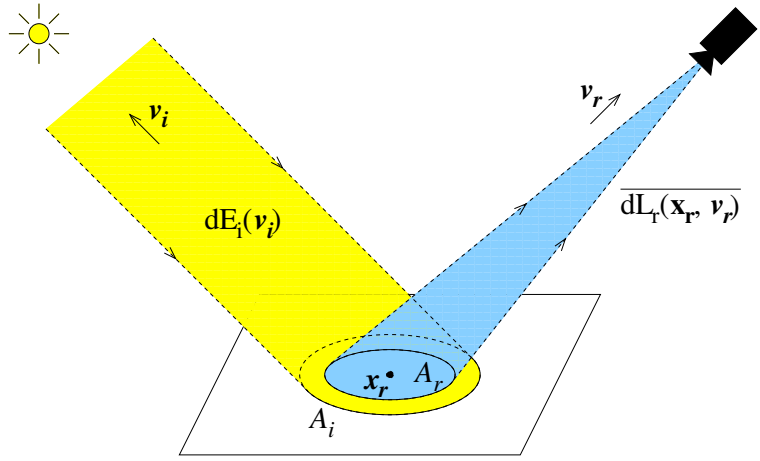


Figure 7.2: *Reflectance geometry for BRDF measurement with a finite size sensor element.*

7.2.2 Textured surfaces

The notion of texture is related to the scale of observation; for example a sheet of paper is usually considered non-textured at macroscale, although it is textured when observed at a microscopic scale. Here we refer to textured surfaces as surfaces which appear textured at the scale of observation. At this scale, such surfaces usually have statistically non-uniform properties. This poses some problems when carrying out single pixel measurements because the portions of the surface covered by the pixel projections vary as the camera changes its position and orientation in space. Practically this means that it is not possible at this scale to carry out statistically meaningful measurements of the surface radiance emanating from the surface. This is illustrated on a simple example in Fig. 7.3. Suppose for simplicity that the surface observed is Lambertian and the variable surface albedo ρ is either 0 (shown in black) or 1 (shown in white). A camera in a position according to Fig. 7.3(b) perceives a patch of albedo $\rho = 1$ while

a camera in configuration according to Fig. 7.3(a) would see $\rho \approx 1/3$. The configurations in Fig. 7.3(a) and Fig. 7.3(b) are in reciprocal positions yet yield differing observed intensities, hence the principle of reciprocity, at the pixel scale, is violated.

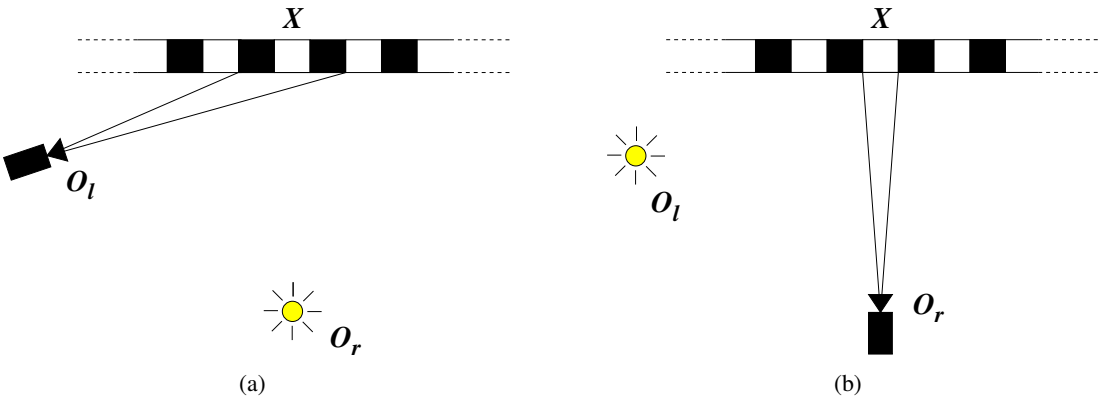


Figure 7.3: *Illustration of the failure of reciprocity in the case of textured surfaces. The portion of the surface viewed by a finite size sensor element depends on the camera position and orientation. This results in non-reciprocal measurements.*

7.2.3 Rough Surfaces

So far, it has been implicitly assumed that surfaces reflect the light coming from the light source directly into the camera. This is known as a *local shading model* [54] (p 77). Even though this is not the most accurate model of the physics of light reflection, this proved sufficient in the case of smooth convex objects. Concave surfaces require a significantly more complex description because the light emitted by a light source may be reflected several times from surface to surface before reaching the camera. This phenomenon is called *inter-reflection*. It can occur *a priori* with any surface presenting some concavities. We consider an important class of surfaces accommodating such phenomena: rough surfaces. Such surfaces are microscopically non-convex and usually present strong inter-reflections.

Let us illustrate the problem on the simple non-convex scene depicted in Fig. 7.4. The scene consists of two planar patches, one of which (denoted by $\boldsymbol{M}$) is a perfect mirror. We consider a camera and a light source and acquire a reciprocal pair of intensity measurements. If we first ignore the mirror, the intensity i_l and i_r measured are reciprocal (we assume the surface is non-textured and therefore reciprocity holds). If we now introduce the mirror into the scene, an

inter-reflection occurs. It can be observed that this inter-reflection contributes to the intensity i_l observed in the left image (see Fig. 7.4(a)), whereas it does not contribute to the intensity i_r observed in the right image because the ray from the interreflection reaches the camera at a different pixel (see Fig. 7.4(b)). Therefore, the measurements are no longer reciprocal because of the inter-reflections. It should be noted that it was not necessary here to consider measurements over finite extent sensor elements in order to prove the non-reciprocity of the measurements. In practice, the averaging of BRDF over the area observed by the sensor would show a similar effect, unless the size of the concavity is smaller than the area subtended by the sensor, in which case all inter-reflections would be captured by the sensor, and reciprocity would be maintained.

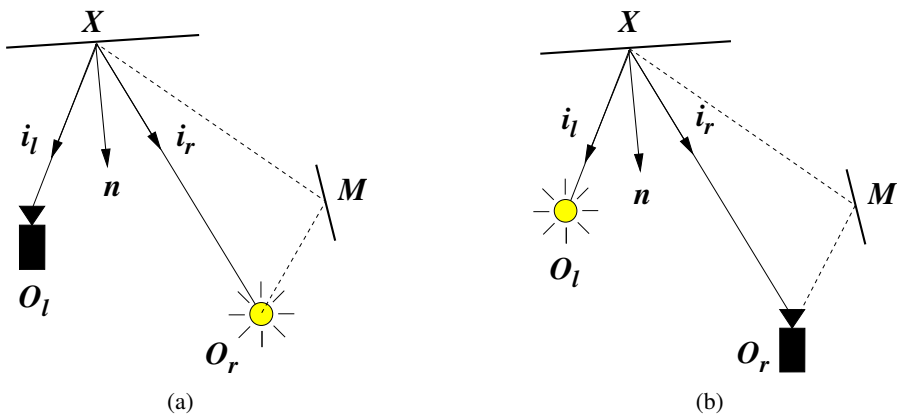


Figure 7.4: *Illustration of the failure of reciprocity in the case of non-convex surfaces. The patch denoted by M is a perfect mirror. The solid line represents the optical path followed by the ray of light responsible for the formation of the image of the point X, if the mirror is not taken into account. The dashed line represents an inter-reflection caused by the introduction of the mirror into the scene. The inter-reflection contributes only to the image of X by the left camera in this case because the inter-reflection in the right image is measured at a different pixel, and reciprocity becomes violated. Similar effects occur in rough surfaces.*

7.3 Novel Helmholtz Stereopsis constraint for rough and textured surfaces

In this section, we formulate a novel HS constraint which does not suffer from the limitations of the previous one, and demonstrate its validity on a simple test example.

7.3.1 Definition of the novel Helmholtz Stereopsis constraint

It is clear from the previous section that the idea of carrying out individual pixel measurements must be abandoned in the case of textured or rough surfaces. The solution proposed is based on carrying radiance measurements over extended areas corresponding to the projection of the same surface region. Let us denote by $\mathcal{A}_r$ a surface region containing the surface point $\boldsymbol{x}_r$ at which we would like to measure the reflectance properties. We define the BRDF as the ratio of the average radiance emanating from this region to the incoming irradiance (see Eq. (7.2)). If the projection of the region $\mathcal{A}_r$ in the left and right images are denoted respectively $\mathcal{P}_l$ and $\mathcal{P}_r$, the average radiances can be approximated by the average pixel intensities computed over $\mathcal{P}_l$ in the left image and $\mathcal{P}_r$ in the right image:

$$I_l = \frac{1}{\mathcal{P}_l} \sum_{\mathcal{P}_l} i_l \quad \text{and} \quad I_r = \frac{1}{\mathcal{P}_r} \sum_{\mathcal{P}_r} i_r \,. \tag{7.3}$$

Note that by abuse of notation, the same symbols have been used to denote regions and their areas. The remaining question now is how to chose the area $\mathcal{A}_r$ in order to guarantee reciprocal measurements.

In the case of smooth textured surfaces, the new definition guarantees reciprocal measurements as long as the image regions $\mathcal{P}_l$ and $\mathcal{P}_r$ backproject exactly onto the same surface patch $\mathcal{A}_r$. The proof is straightforward. Every optical path passing through a point in the surface neighbourhood being reciprocal, the average intensity values I_l and I_r are also reciprocal because they correspond to the integration of all paths through the region. In practice, it is not possible to average the intensity values over areas which correspond strictly to the same area, because of the finite resolution of the sensor elements. However, if the size of the averaging area is large enough with respect to the size of the sensor element, the error due to the finite resolution of the sensor becomes negligible.

In the case of rough surfaces (*i.e.* locally non-convex), it has been proved recently in [128, 129, 130, 41] that such a definition guarantees reciprocity. The macro-shape of a rough surface can be represented locally by a reference plane (see Fig. 7.5). The main idea of the proof is that, if the surface exhibits a reciprocal behaviour at a microscopic level, then it can be easily shown, at least within the scope of geometric optics, that any optical path passing through the structure is reciprocal. As a result, after summing all possible paths, our previous definition of BRDF is

reciprocal, up to boundary effects caused by optical paths for which the incident ray enters the surface outside the patch and leaves inside it (or similarly, when the incident ray enters inside the point neighbourhood and leaves outside). In practice, the impact of boundary effects can be decreased by averaging over more extended surface point neighbourhoods.

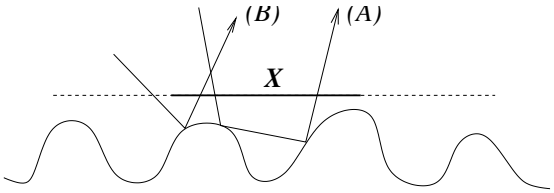


Figure 7.5: *A rough surface can be represented locally by a reference plane (represented here by a dashed line). An extended neighbourhood is considered on the reference plane (represented by a solid line). Two optical paths are shown. The path (A) enters and leaves inside the neighbourhood defined. This is not the case of the optical path (B), which enters outside of the neighbourhood and leaves inside it, and therefore contributes to the boundary effects. If the neighbourhood is chosen large enough, the boundary effects due to local inter-reflections become negligible.*

We can therefore define the following HS constraint which is applicable to both textured and rough surfaces:

$$(I_l \boldsymbol{s}_l - I_r \boldsymbol{s}_r) \cdot \boldsymbol{n} = 0\,. \tag{7.4}$$

Note that the concept of rough or textured surfaces depends on the scale at which the surface is viewed. For example, a sheet of white paper is a smooth surface at a macroscale, while it is rough at a microscale. Clearly the averaging region should be adapted to the scale of the texture or structure pattern of the surface. We will come back to the problem of the choice of the scale in Section 7.4. Before that, we validate the novel constraint defined in Eq. (7.4) on a simple test object.

7.3.2 Experimental validation

A simple experiment was conducted with a concave object exhibiting strong inter-reflections. The object consists of a spherical cap obtained by sectioning a white ping-pong ball[1]. The reference plane chosen is the one corresponding to the plane of the cut. In this plane, we

[1] The concavity is not a hemisphere because the plane of the cut does not pass through the centre of the ping-pong ball.

consider the point $\boldsymbol{X}$ located at the centre of the circle defined by the cross section, and choose the region bounded by the circle as the extended neighbourhood for this point. In this particular case, there exists no boundary effects, because the scene consists of only one concavity, and all the optical paths entering or exiting the concavity must pass inside the extended neighbourhood defined. The outer part of the ping-pong ball is coated with some clay in order to ensure that there are no transparency effect perturbing the experiment. Note that this object is locally smooth, however it exhibits strong inter-reflections at the scale of the whole concavity. This test object is very simple however very interesting because it allows us to isolate a single concavity, and test the new principle on this concavity, without being affected by boundary effects. More complex examples of rough surfaces will be seen in the Section 7.5.2.

The objective of the experiment is two-fold. Firstly we would like to show that pixel based radiance measurements are affected by inter-reflections, secondly we would like to verify that even though it is not possible to reconstruct accurately the microstructure of the object (here the inner part of the ping-pong ball), it is possible to reconstruct the macrostructure of the object (here the orientation of the ping-pong ball section) by considering intensities averaged over the area covered by the projection of the section in each image. The experimental set-up described in Section 6.5.2 of the last chapter was used to acquire reciprocal pairs of images of the object. In total, five different sets were acquired, each set corresponding to a different inclination angles of the ping-pong ball section and containing eight reciprocal pairs of images. The inclination angle is measured with respect to the vertical direction (see Fig. 7.6). The values of the inclination angle considered are given in the first row of Table 7.1. We show in Fig. 7.7 the images corresponding to the case where the inclination angle is $\alpha \approx 45°$.

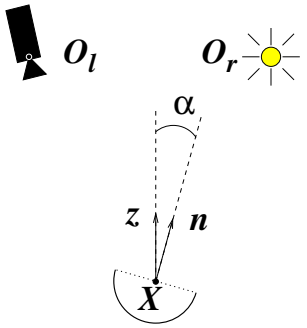


Figure 7.6: *Experimental setup for the ping-pong ball section. The normal $\boldsymbol{n}$ of the ball is inclined by an angle α with respect to the vertical direction $\boldsymbol{z}$.*

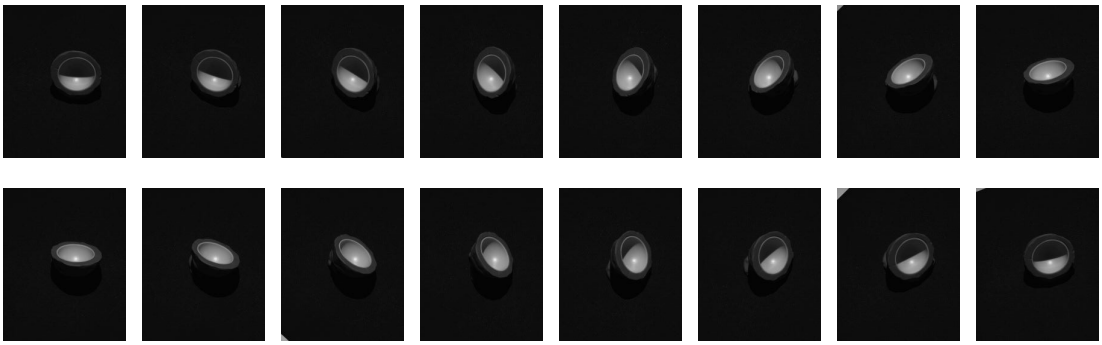
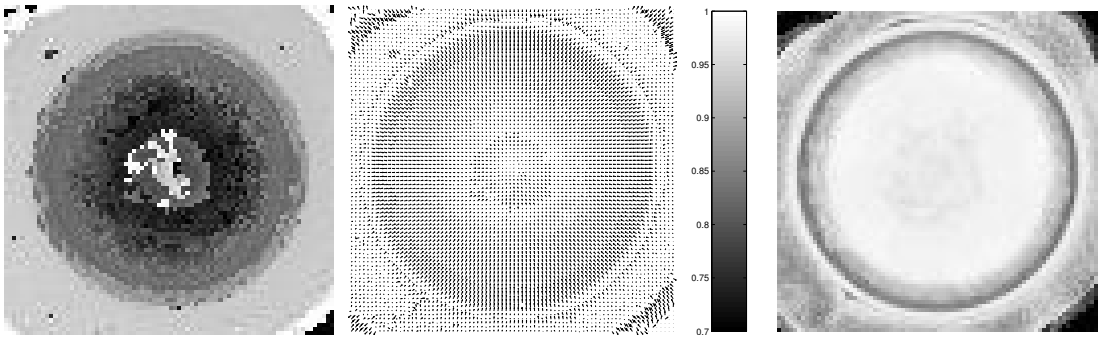


Figure 7.7: *One set of eight reciprocal pairs of images of a ping-pong ball section (inclination angle $\alpha \approx 45$). The bottom row images are obtained by interchanging the position of the light source and camera with respect to the top row image. The outer shell of grey values is the clay that is holding the half ping-pong ball and also ensures that there are no transparency effect perturbing the experiment.*

Limitation of the pixel based constraint

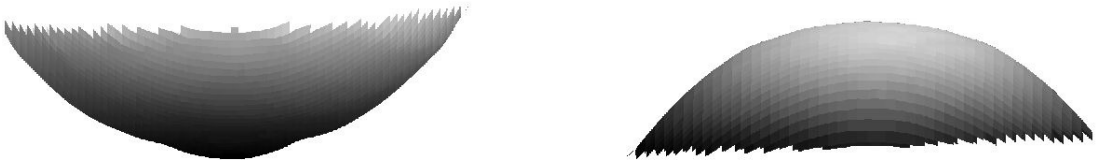
We first tried to reconstruct the entire surface of the ball section by applying the standard HS algorithm, which considered point-based intensity measurements. We chose the set of images corresponding to the smallest inclination angle ($2.9\,^\circ$) in order to minimise the occlusions. For this specific inclination, each point in the concavity is visible in all eight reciprocal pairs of images, therefore the whole concavity can be reconstructed. We applied the method described in the previous chapter. The results obtained for the depth map, normal field and support measure, are shown in Fig. 7.8. Even though all points within the concavity are associated with a high support measure, it can be observed that the points located at the bottom of the concavity present large depth and normal errors.

The 3D model obtained by integrating the normal field is shown in Fig. 7.9(a). Note that we applied some manual segmentation based on the support measure in order to filter out background points which did not present any interest for the experiment. Some artefacts are clearly visible at the bottom of the 3D model. These artefacts are probably due to inter-reflection effects which are stronger in the central part of the concavity. For comparison, the 3D model obtained in the case of the snooker ball is given in Fig. 7.9(b). This object is highly similar except that it is convex instead of concave. The snooker ball does not present the artefacts visible in the case of concave surfaces. This experiments suggests that the constraint based on single pixel radiance measurements is corrupted by inter-reflections occurring in concavities.



(a) (b) (c)

Figure 7.8: *Reconstruction of the half ping-pong ball. (a) represents the depth map, (b) the normal field and (c) the support measure.*



(a) (b)

Figure 7.9: *Reconstruction of the inner part (a) and outer part (b) of a spherical cap. (a) corresponds to the ping-pong ball section considered in this section, while (b) corresponds to the snooker ball considered in the previous chapter (see Fig. 6.7). Both models were obtained from integration of the normal field. Similar view points are considered for both objects. It is clearly visible that (a), which is concave, is not as well reconstructed as (b), which is convex. The errors in the first case are due to inter-reflections which corrupt pixel based intensity measurements.*

Further experiments need to be made in order to quantify the phenomenon.

Validity of the constraint based on extended regions

In this case, we are interested in reconstructing the macrostructure of the ping-pong ball, which is represented by the orientation of its cross section. For each set of images, we compute the average intensities I_l and I_r over the area covered by the projection of the ping-pong ball section in each reciprocal pair of images. The constraints defined in Eq. (7.4) can then be formed for each reciprocal pair of images, and the normal $\boldsymbol{n}$ can be computed by using one of the methods defined in the previous chapter. In this case, we used the radiometric method (see Section 6.3).

Table 7.1: *Comparison between the inclination angles estimated by imposing the novel constraint based on radiance measurements over extended regions (α_{ext}) with the ground truth values (α_{GT}). δ is the angular difference between the two normals. Note that δ is not equal to the difference between α_{ext} and α_{GT} because the two normals are usually not located in the same vertical plane. All values are in degrees.*

Set	1	2	3	4	5
α_{GT}	2.9	17.4	36.1	45.0	54.5
α_{ext}	3.7	15.1	37.4	46.7	56.7
δ	3.8	2.8	1.3	2.6	2.6

Table 7.2: *Root Mean Squared (RMS) and maximum deviation angle of the vectors $(I_l \boldsymbol{s}_l - I_r \boldsymbol{s}_r)$ from the plane orthogonal to the normal (eight vectors were used to compute the deviation). All values are in degrees.*

Set	1	2	3	4	5
RMS	0.25	0.22	0.15	0.14	0.14
max	0.37	0.34	0.29	0.30	0.24

In order to evaluate the accuracy of the normal estimation, the results of the reconstruction are compared with the ground truth normal obtained by performing conventional stereo on the outlines of the cut of the ball. The results are shown in Table 7.1. They seem to exhibit a relatively good agreement. We also evaluate the consistency of the set of constraints in Eq. (7.4) formed by all the reciprocal pairs. Consistent measurements should lead to coplanar vectors $(I_l \boldsymbol{s}_l - I_r \boldsymbol{s}_r)$. We therefore measure the angular deviation of these vectors from the plane perpendicular to the recovered normal $\boldsymbol{n}$. The Root Mean Squared (RMS) and the maximum deviation are shown in Table 7.2. The sets of constraints appear consistent for all orientation of the ball. This evidence supports the theory that the constraint can be used to determine the macro-structure of the scene.

7.4 Implementation

The implementation of the previous constraint to rough and textured surface reconstruction requires to construct consistent measurements of surface radiance. Theoretically, these measurements are computed by averaging intensity values over image regions corresponding to the projection of the same physical surface patch, as described in Eq. (7.3). Such a construction is

non-trivial because the image areas over which the intensity measurements should be averaged depend on the local reference plane orientation and also on the local scale of the structure or texture sub-elements, which are both unknown *a priori*. We propose two different algorithms to address these issues.

7.4.1 Extended HS algorithm

The first algorithm proposed is called *extended HS*. This algorithm uses simple isotropic filtering of the images by an appropriate convolution kernel in order to approximate the average radiance values. In the implementation, although Eq. (7.3) suggests simple averaging, a Gaussian convolution kernel was used in order to down-weight the contribution of the most distant points in the neighbourhood. The choice of the parameters of the convolution kernel (size and standard deviation) are dictated by the scale of the surface structure or texture sub-elements. Currently these parameters are set empirically. The main advantage of this implementation is that it is simple and leads to a negligible increase in the run-time compared to the standard implementation based on single pixel measurements, because the image convolutions can be done as a pre-processing step.

A limitation of the algorithm is that it implicitly assumes an approximately uniform scale for the texture or structure pattern. In practice, the method has been observed to be fairly insensitive to the choice of the size of the convolution kernel, as long as it is sufficiently large to capture the texture or structure variations. For this reason, the previous assumption is not a problem for a large number of objects, such as the ones considered in the experiments described in the next section. A more sophisticated implementation, able to cope with large variations in texture or structure scale, would determine automatically the scale of the texture or structure sub-elements, and adjust it locally at each image neighbourhood. In the case of the determination of the texture scale, it has been shown that the polarity provides a useful statistic [54] (p 196). Similar techniques may be applicable to rough surfaces. An alternative method which works well with both types of surfaces is presented in the next section.

7.4.2 Adaptive HS algorithm

The second algorithm proposed is called *adaptive HS*. The algorithm tries to dynamically improve the averaging in Eq. (7.3). The surface neighbourhood is represented by a disc whose orientation (represented by the normal $\boldsymbol{n}$) and scale (represented by the radius r) must be determined. The main idea of the algorithm is that the support measure should be optimum at the correct scale and surface orientation. We therefore try to find $\boldsymbol{n}$ and r which optimise the support measure associated with the surface patch.

The solution proposed is iterative and requires initialisation of the surface normal. Such initialisation is provided for example by the results of the extended HS algorithm described earlier, or by choosing any normal satisfying the visibility constraint. At each iteration, the current normal estimate is used to compute the exact projection of a disc centred at the depth provided by the previous algorithm. A number of hypotheses are made concerning the radius of the patch and only the one resulting in the best support measure is retained (*i.e.* the one leading to the largest support measure defined in Eq. (6.6)). Rewriting Eq. (7.4) with the intensities averaged over the projection of the disc with optimum radius, a refined normal is then computed, and the optimum radius can be re-estimated for the same series of hypotheses. We iterate the procedure until the change in the orientation of the normal estimated is less than a certain threshold (0.1° in our implementation) or the maximum number of iterations is exceeded (10 in our implementation). The algorithm is summarised in Algorithm 4.

In terms of run-time, the adaptive HS algorithm is slower than the extended HS algorithm because it is iterative and also because the computation of the projection of a disc is more computer intensive. The adaptive HS algorithm is however expected to give more accurate results because it averages the intensities over areas corresponding to the projection of the same surface point neighbourhood and also optimises the scale at each surface point.

7.5 Results

In this section we demonstrate the applicability of the method to textured and rough surfaces. The experimental setup and methodology are the ones described in the previous chapter. The reconstruction method is also similar to the one described in the previous chapter, except that

Algorithm 4 Adaptive HS algorithm

The objective is to compute the normal $\boldsymbol{n}$ and radius r of the circular surface patch with highest support measure s. The parameter ϵ represents the tolerance in angular change in surface normal orientation, and i_{max} is the maximum number of iterations.

1. Initialisation: $i \leftarrow 0$, $s \leftarrow -\infty$, $\boldsymbol{n} \leftarrow \boldsymbol{n}_0$, $r \leftarrow 0$, where $\boldsymbol{n}_0$ is the normal provided by the extended HS algorithm, if available, or otherwise any value satisfying the visibility constraint.

2. Do:

   (a) Assume that the orientation of the patch is $\boldsymbol{n}$, and compute the average image intensities I_l and I_r for different radius hypotheses r_k,

   (b) Form the constraints in Eq. (7.4) and compute the normal $\boldsymbol{n}_k$and the support measure s_k associated with each radius r_k using for example the method described in Section 6.3,

   (c) Find the parameter k_{opt} which leads to the highest support measure (*i.e.* largest value defined in Eq. (6.6)),

   (d) If $s_{k_{\text{opt}}} \leq s$, exit the loop,

   (e) Otherwise set δ to the absolute value of the angle between $\boldsymbol{n}$ and $\boldsymbol{n}_{k_{\text{opt}}}$,

   (f) Update: $i \leftarrow i + 1$, $s \leftarrow s_{k_{\text{opt}}}$, $\boldsymbol{n} \leftarrow \boldsymbol{n}_{k_{\text{opt}}}$, $r \leftarrow r_{k_{\text{opt}}}$,

   while $\delta > \epsilon$ and $i < i_{max}$.

3. Return $\boldsymbol{n}$.

(a) (b)

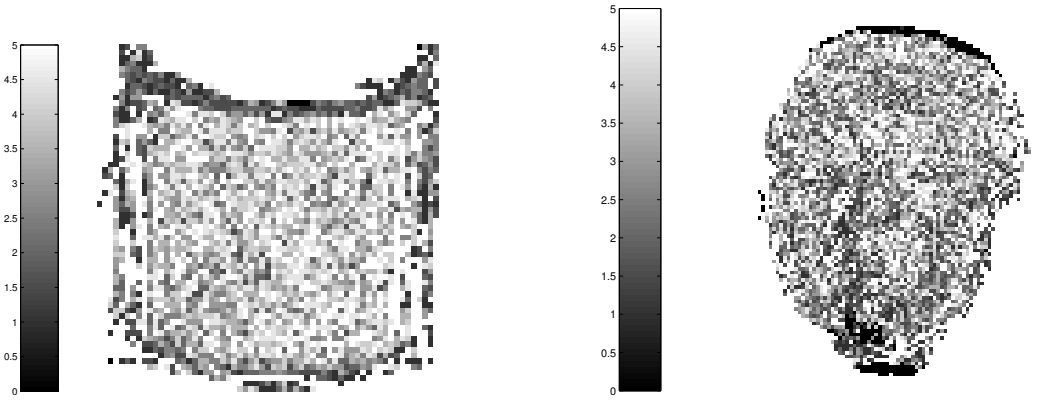
Figure 7.10: *Images of the textured objects used for reconstruction: (a) is a mug and (b) the head of a polystyrene mannequin.*

consistent radiance measurements provided either by the extended HS or adaptive HS algorithms are used. These two techniques are compared with the standard HS algorithm described in the previous chapter.

7.5.1 Textured surfaces

Two textured objects were considered (see Fig. 7.10). The first one is a mug with some blue dot patterns painted on a white background surface. The second object is a polystyrene mannequin head, in this case the texture comes from the material itself. Both surfaces are specular, which makes the reconstruction difficult with standard techniques.

For both objects, a bounding box has been defined and the 3D space has been discretised into square voxels at a resolution of $2\,\text{mm} \times 2\,\text{mm} \times 2\,\text{mm}$. A window of size 5×5 pixels is used to resolve the matching ambiguity during depth search. In the case of the extended HS algorithm, a Gaussian convolution kernel of size 21×21 pixels with standard deviation 4 pixels was used. The choice of the size of the kernel is dictated by the scale of the texture at the surface of the objects. In this case, the scale has been selected empirically. We have observed that the choice of this parameter does not need to be very accurate. Close values will most likely lead to the same results as long as the scale is large enough to capture the texture sub-elements. Unnecessarily large scales are however not recommended because they would result in a decrease in the resolution of the reconstruction. In the case of the adaptive algorithm, the scale, which is represented by the radius of the disc projected, is allowed to take



(a) (b)

Figure 7.11: *The grey level of a pixel encodes the radius (in mm) of the disc representing the surface patch at each point, after convergence of the adaptive HS algorithm, for the two textured objects.*

arbitrary values within the interval $[0, 5]$ mm sampled at a resolution of 0.5 mm. The algorithm selects automatically the best value within the interval for each surface point. The optimum scale found after convergence at each surface point can be found in Fig. 7.11.

The results of the reconstruction are shown in Fig. 7.12 and Fig. 7.13. Qualitatively, it can be observed that the depth map and the normal field are less noisy and apparently more accurate in the case of the extended and adaptive HS algorithms, compared to the standard HS algorithm. Regarding the support measure, although it is clear that points located at the surface of the object have high values, it is difficult to say which method leads to the highest value by simple visual observation. More is said on this topic in the next paragraph. It can be observed that most background points are eliminated (zero support measure). This is due to the input image thresholding which has been applied during reconstruction and also the requirement for the surface points to be visible simultaneously in all reciprocal pairs of images. In spite of this filtering, there remains a number of background points, in particular in the case of the mug. In the final reconstruction, these outliers were eliminated based on the support measure values. Such segmentation was done manually, although the task could certainly be automated in the future. We did not consider doing such optimisation in the current implementation, because our objective is to demonstrate the feasibility of HS in the case of textured and rough surfaces, therefore it is not desirable to add other potential sources of errors in the analysis. The 3D models were then produced by integration of the normal fields weighted by the support measure at each point, as described in the previous chapter. Background points are set to zero support,

Table 7.3: *Comparison of the RMS support measure obtained by the different reconstruction algorithms for the textured objects considered.*

	Mug	Mannequin head
standard HS	0.9455	0.9438
extended HS	0.9869	0.9883
adaptive HS	0.9912	0.9908

which has the effect of eliminating them from the reconstruction. It can be observed that the 3D model obtained with the extended and adaptive HS have a smoother appearance than the one obtained from the standard HS algorithm. We also show the 3D model obtained by the adaptive HS algorithm texture mapped with one input image in Fig. 7.14. The results seem realistic and able to capture accurately the 3D shape of the objects.

Quantitatively, we use the Root Mean Squared (RMS) support measure in order to define a measure of the consistency of the normals obtained with the intensity measurements. The RMS support measure is computed only over the points which belong to the object surface, hence the necessity of an accurate segmentation from the background if we want the measure to be reliable. Denoting by N the number of surface points and by s_{ij} the support measure at the surface point parameterised by i and j, the RMS support measure is defined by: $\sqrt{\frac{1}{N}\sum_i \sum_j s_{ij}^2}$. The values obtained for the different algorithms are presented in Table 7.3. The quantitative results confirm that the support measure is increased with the two methods considering extended regions (extended and adaptive HS algorithms) compared to the standard HS algorithm. Such an increase is important because it means that these methods are able to produce more consistent models than the standard one. The extended HS and adaptive HS algorithms give very close results. As expected, the adaptive HS leads to the highest values, because it is the only one to optimise the scale and orientation of the patch locally at each surface point.

7.5.2 Rough surfaces

We now consider the reconstruction of two objects with rough surfaces (see Fig. 7.15). The first one is a teddy bear, and the second one is a piece of corrugated cardboard. Both surfaces are highly anisotropic and exhibit strong inter-reflection effects, making the reconstruction again very challenging by state of the art techniques.

The reconstruction is made at a resolution of $2\,\text{mm} \times 2\,\text{mm} \times 2\,\text{mm}$ for the teddy bear and

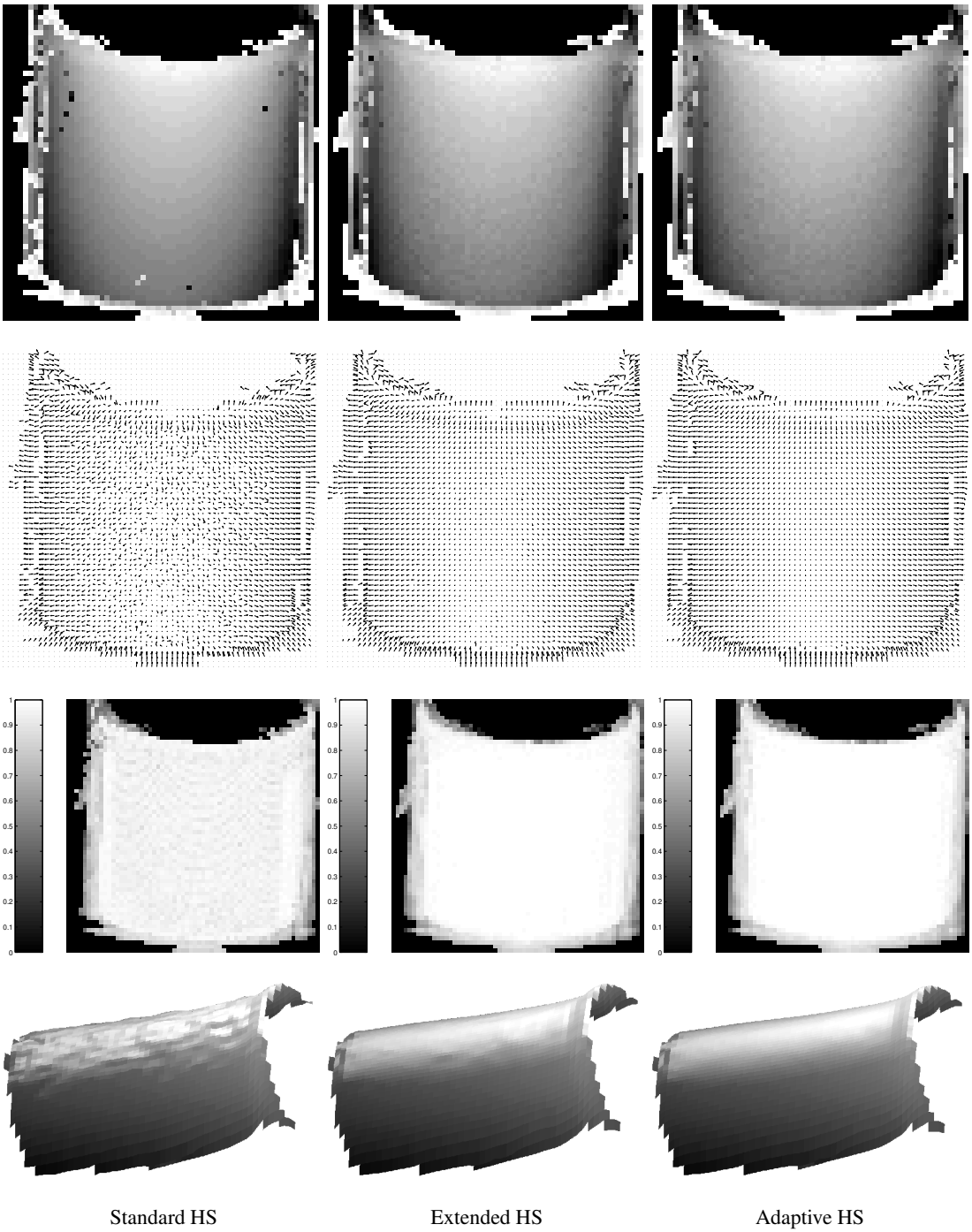


Figure 7.12: *Reconstruction of the object 'Mug'. The left, middle and right columns correspond respectively to the standard, extended and adaptive HS algorithms. From top to bottom, the rows represent the depth map, the normal field, the support measure, and the 3D model obtained from integration of the normal field.*

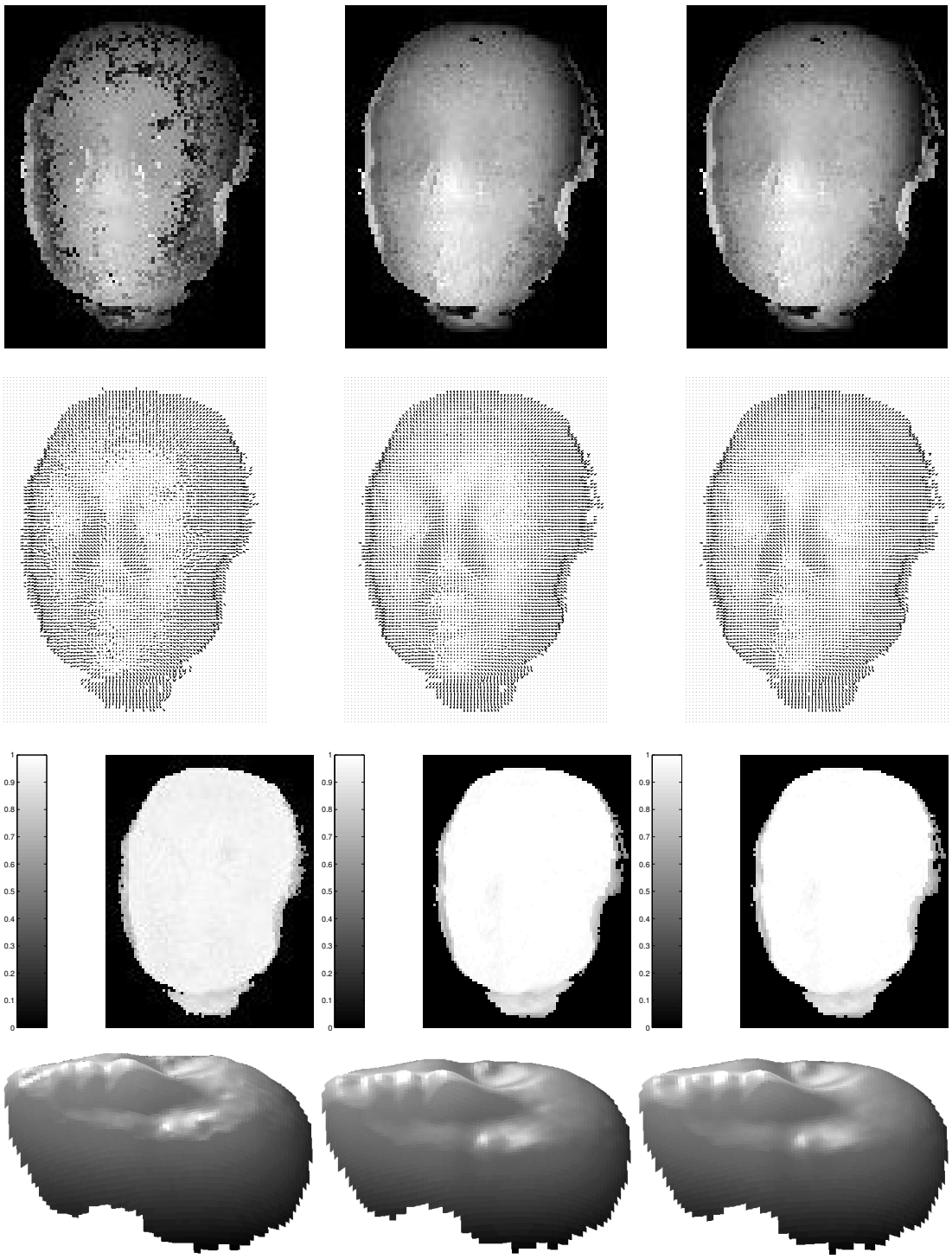
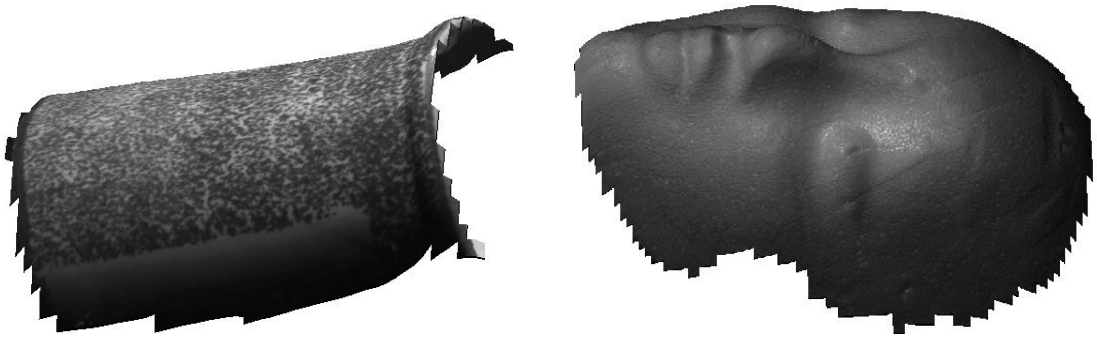
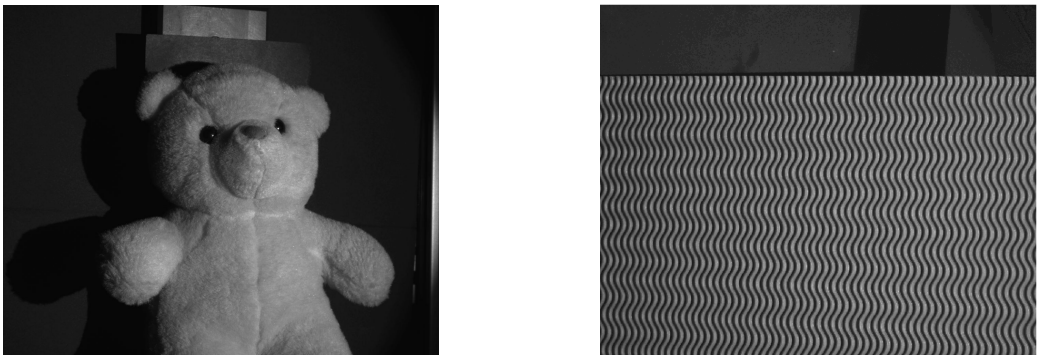


Figure 7.13: *Reconstruction of the object 'Mannequin head'. The left, middle and right columns correspond respectively to the standard, extended and adaptive HS algorithms. From top to bottom, the rows represent the depth map, the normal field, the support measure, and the 3D model obtained from integration of the normal field.*



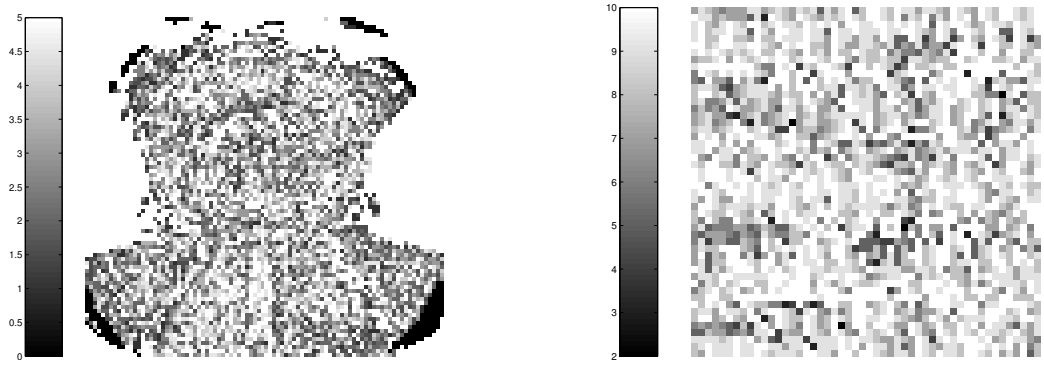
(a) (b)

Figure 7.14: *Texture mapped 3D models obtained by the adaptive HS algorithm for the two textured objects.*



(a) (b)

Figure 7.15: *Images of the rough objects used for reconstruction: (a) is a teddy bear and (b) a sheet of corrugated cardboard.*



(a) (b)

Figure 7.16: *The grey level of a pixel encodes the radius (in mm) of the disc representing the surface patch at each point, after convergence of the adaptive HS algorithm, for the two rough objects.*

$1\,\text{mm} \times 1\,\text{mm} \times 1\,\text{mm}$ for the corrugated sheet. As previously, a window of size 5×5 pixels is used to resolve the matching ambiguity during depth search. In the case of the teddy bear, we used a convolution kernel of size 21×21 pixels with standard deviation 4 pixels for the extended HS algorithm, and allowed the radius of the disc representing the surface patch to take arbitrary values within the interval $[0, 5\,\text{mm}]$ sampled at a resolution of 0.5 mm for the adaptive HS algorithm. The size of the structure elements is much larger in the case of the corrugated sheet, therefore it is necessary to define larger extended regions for averaging. The size of the convolution kernel was set to 101×101 pixels with standard deviation 20 pixels and the possible values for the radius to the interval $[2, 10\,\text{mm}]$ sampled at a resolution of 1 mm. Again it was observed that the scale did not matter much as long as it was large enough to cover the concavities defining the surface structure and make the boundary effects negligible. The optimum scale found after convergence at each surface point is shown in Fig. 7.16.

Fig. 7.17 and Fig. 7.18 show the results of the reconstruction for the three different algorithms. Similarly to the previous section, we can observe that the depth maps and normal fields are more noisy in the case of the standard HS algorithm. This is considerably improved by the algorithm considering extended regions. It is interesting to note that in the case of the corrugated sheet, the microstructure, *i.e.* here the undulations of the structure, can be reconstructed. We provide a reconstruction of a smaller area of the sheet at a finer resolution in Fig. 7.19.

It can be verified that the support measure at surface points is not as high when single pixel measurements are considered, in particular in the case of the corrugated sheet. The darker

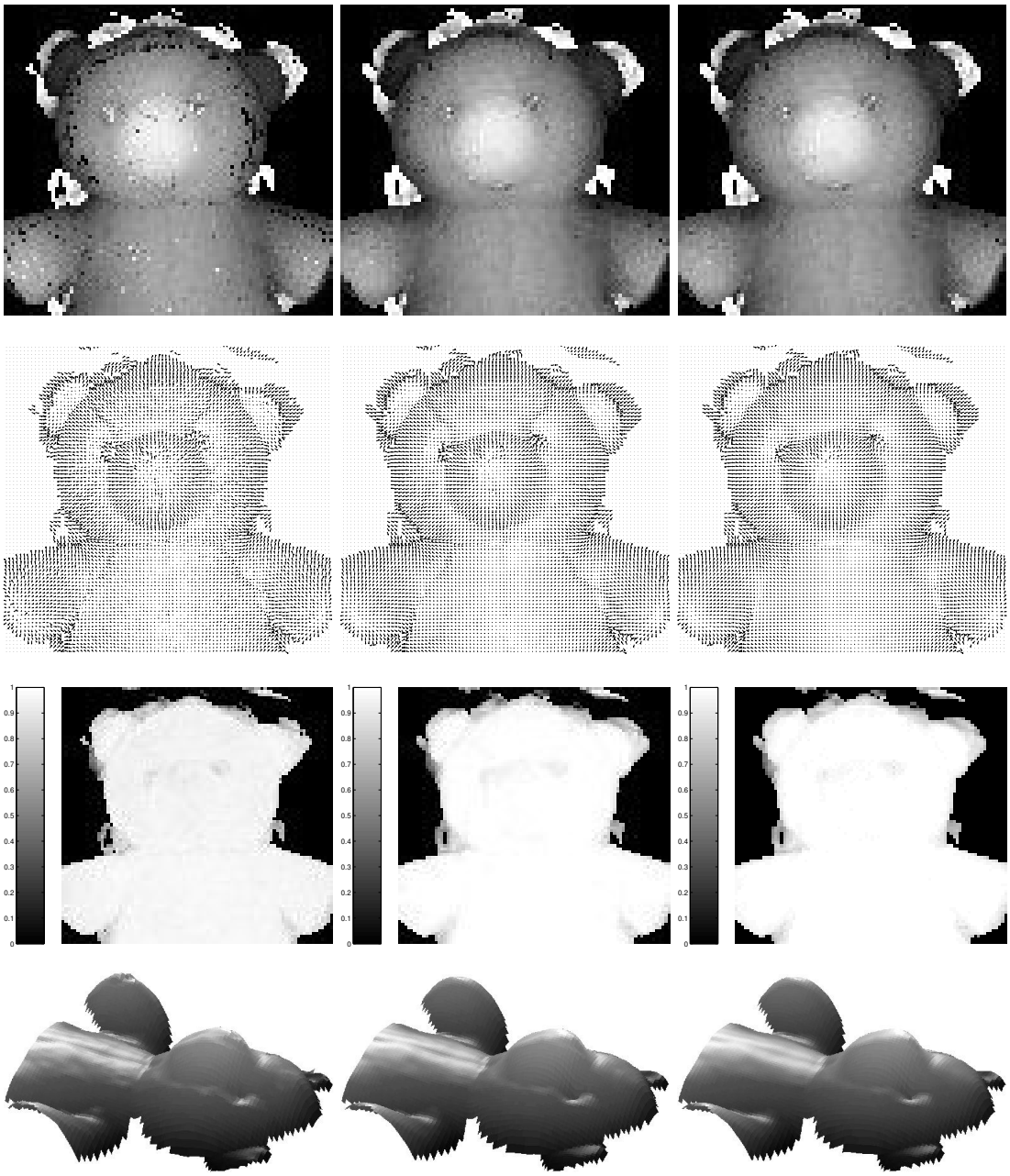


Figure 7.17: *Reconstruction of the object 'Teddy bear'. The left, middle and right columns correspond respectively to the standard, extended and adaptive HS algorithms. From top to bottom, the rows represent the depth map, the normal field, the support measure, and the 3D model obtained from integration of the normal field.*

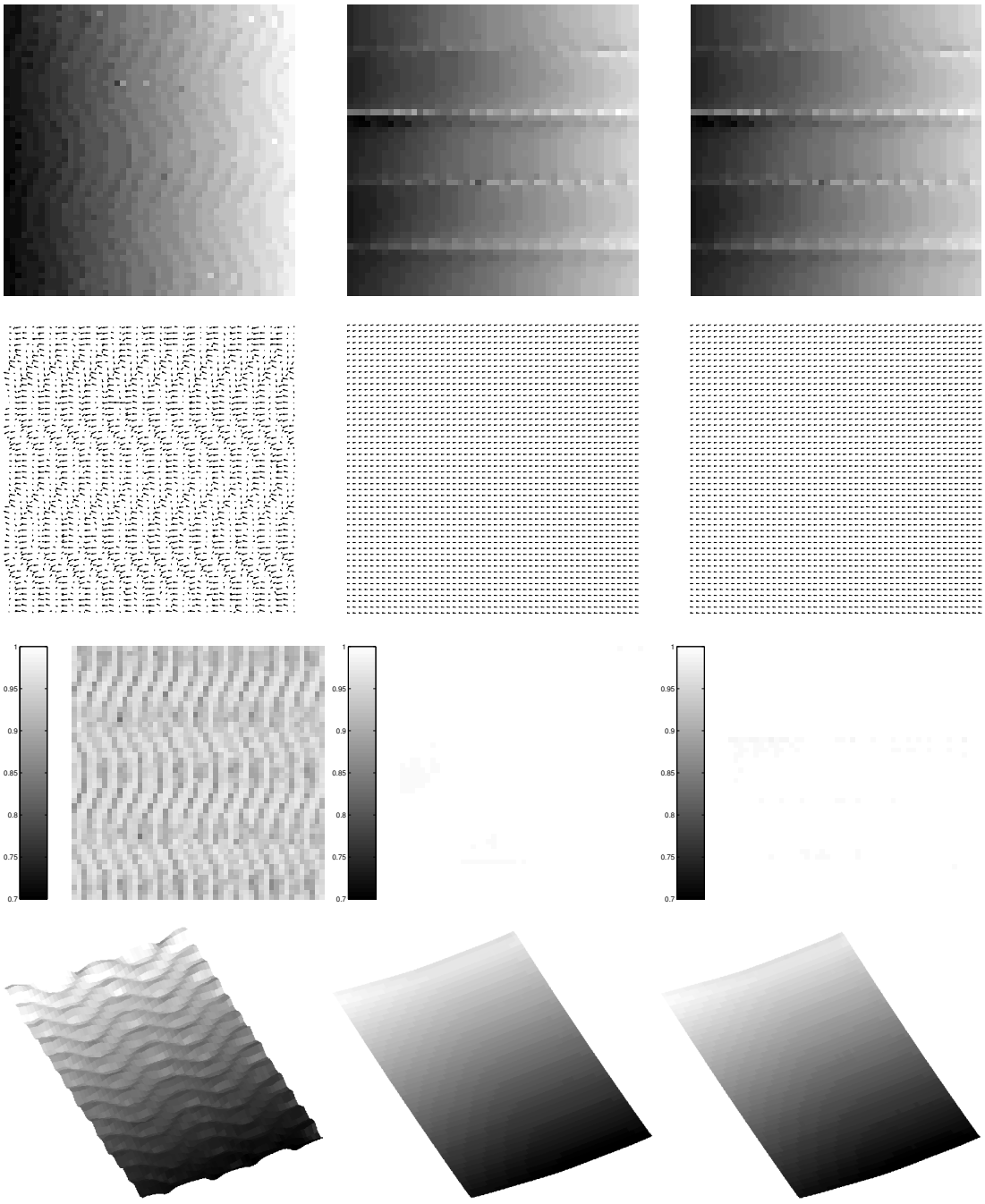


Figure 7.18: *Reconstruction of the object 'corrugated sheet'. The left, middle and right columns correspond respectively to the standard, extended and adaptive HS algorithms. From top to bottom, the rows represent the depth map, the normal field, the support measure, and the 3D model obtained from integration of the normal field. Note that the images representing the support measure in the case of the extended and adaptive HS algorithms are not missing or corrupted; they appear invisible because the support measure is very high at every point.*

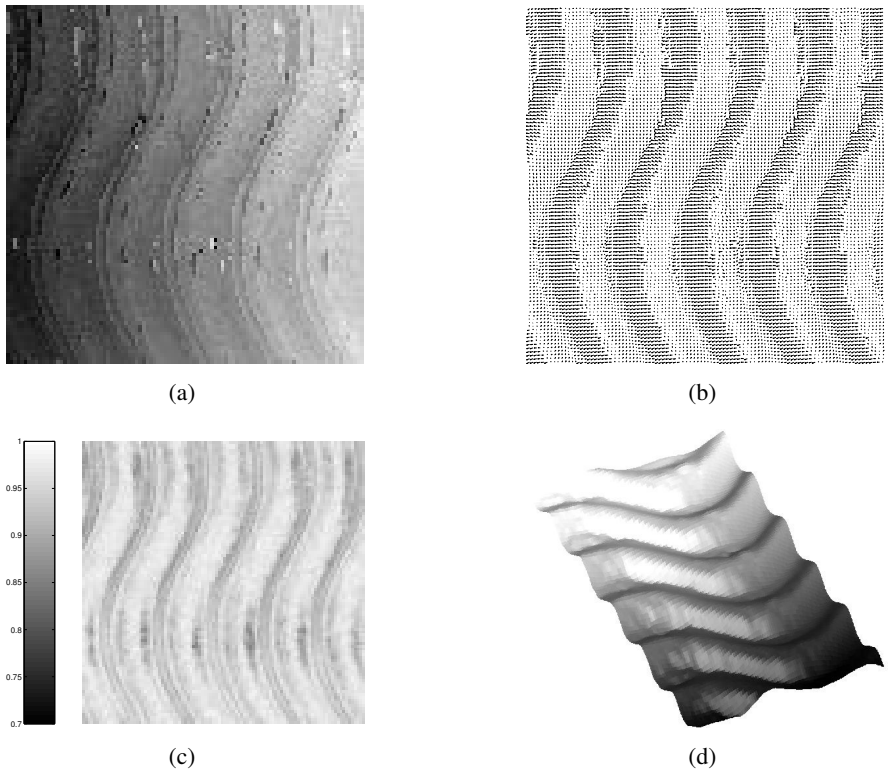


Figure 7.19: *Reconstruction of the object 'corrugated sheet' at a smaller scale using the standard HS algorithm. (a) represents the depth map, (b) the normal field, (c) the support measure, and (d) the 3D model obtained from integration of the normal field.*

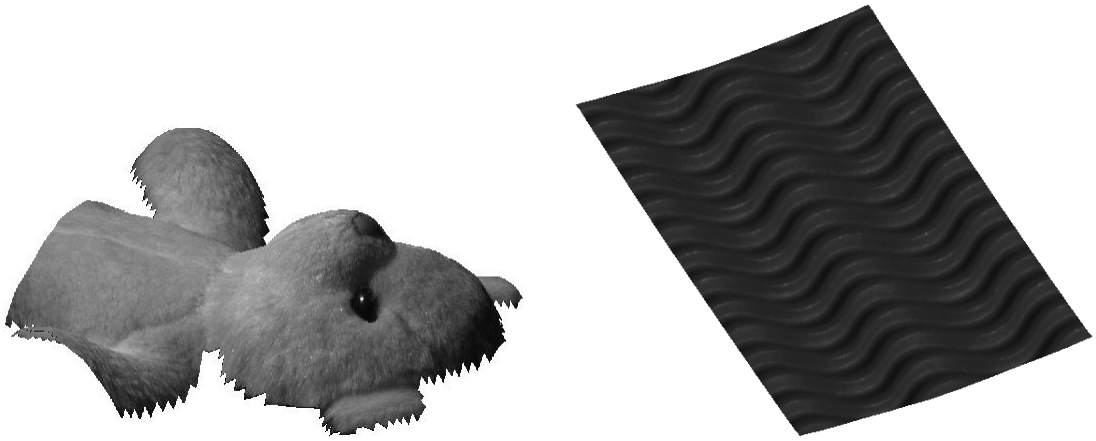
Table 7.4: *Comparison of the RMS support measure obtained by the different reconstruction algorithms for the rough objects considered.*

	Teddy bear	corrugated sheet
standard HS	0.9506	0.9367
extended HS	0.9861	0.9970
adaptive HS	0.9900	0.9973

region of lower support measure suggest the location of regions most affected by the inter-reflection effects. This tendency, visually observed, is confirmed by the RMS support measure computed over the surface points of both objects, after appropriate segmentation of the background, which are shown in Table 7.4. The adaptive algorithm leads to the highest support measure among all algorithms. In the case of the teddy bear, the 3D model obtained with the extended or adaptive algorithms have a smoother appearance, which is still able to capture fine details such as the seam on its belly. In the case of the corrugated sheet, it is possible to reconstruct both the microstructure and the macrostructure of the object. Note however that the reconstruction of the microstructure has several disadvantages. Firstly it has a high memory requirement and run-time because of the necessity to sample the surface at a very fine resolution. Such reconstruction would not be practical if a large area was to be reconstructed at such a resolution. Secondly the reconstruction at a microscopic scale is necessarily inaccurate because of the inter-reflection effects. This is suggested by the lower support measure observed in this case. We show the 3D models obtained by the adaptive HS algorithm after texture mapping in Fig. 7.20.

7.6 Conclusions

Rough and highly textured surfaces are often encountered in reality. The ability to reconstruct their shape is important in computer vision. In this work, we explicitly addressed the problem of reconstructing such surfaces by Helmholtz Stereopsis (HS). We observed that radiometric constraints constructed from single pixel measurements are necessarily biased when inter-reflections or strong texture are present. We showed that a solution is to construct consistent measurements from image regions corresponding to the projections of the same bounded surface patch instead. An experiment on a hemispherical concavity revealed good agreement of the results with the theory. It is important to note that solution proposed addresses the problem



(a) (b)

Figure 7.20: *Texture mapped 3D models obtained by the adaptive HS algorithm for the two rough objects.*

of local inter-reflections only; global inter-reflections effects are very challenging and usually very difficult to take into account.

Two different HS algorithms generalised to highly textured and rough surfaces were proposed. The first algorithm, called extended HS, approximates the average image radiances by pre-processing each input image using isotropic filtering. This is equivalent to running the standard HS algorithm on the pre-convolved input images. As such, consistent measurements can be obtained without significant increase in the run-time of the standard HS algorithm. The other algorithm, called adaptive HS, finds the optimum scale and refines the normal obtained by the extended HS algorithm, by iteratively averaging the intensities over areas corresponding to the projection of the same surface point neighbourhood.

The experiments on objects exhibiting rough surface properties or strong texture showed that the novel formulation usually results in an increase in the quality of both the depth map and the normal field reconstructed, compared with the standard HS algorithm. It also resulted in a significant improvement in the consistency of the radiometric constraints used to validate the hypotheses on surface geometry, and produced realistic 3D models with smoother geometries. It is important to mention that in certain cases, for example if the scale of the texture pattern or structure pattern defining the rough surface is large with respect to the camera resolution, then the standard HS algorithm is usually able to produce visually accurate reconstructions. In this case, it is therefore usually possible to obtain a reconstruction of the scene both at a

microscopic scale (with the standard HS algorithm) and a macroscopic scale (with the novel algorithm). The reconstruction obtained by the standard HS algorithm has two disadvantages however: i) it has a high computational cost because of the fine resolution required and ii) it may be inaccurate because of the limitations inherent to the standard HS constraint mentioned earlier.

Part III

Epilogue

Chapter 8

Conclusions and future work

8.1 Conclusions

In this thesis, we have considered the problem of improving the accuracy of object reconstruction from images. Our contributions were made in two main areas of computer vision which are camera calibration and Helmholtz Stereopsis (HS).

In the case of camera calibration, we have concentrated on using invariants in order to increase the accuracy. Invariants allow more accurate determination of the camera parameters because they define constraints on subsets of the camera parameters, that can be used to generate new data without increasing arbitrarily the dimensionality of the problem. We have considered two main situations.

The first situation corresponds to a translating camera. In this case, we have developed a novel calibration method which is based on Points at Infinity (PI) representing directions present in the scene. The method uses the invariance properties to translation motion, of the projection of these points, called the Vanishing Points (VPs), in order to decouple the translation parameters from the other parameters, thereby generating two simpler sub-problems with constant number of unknowns. Our method differs significantly from other VP-based methods, because it does not require to observe parallel sets of lines in the scene. This is a considerable advantage in terms of flexibility, in addition to the improvements in calibration accuracy that were observed.

The second situation that we considered is the case of a zooming camera moving freely in 3D

space. For this purpose, we have introduced the novel concept of the Normalised Image of the Absolute Conic (NIAC), which comes as a generalisation of the Image of the Absolute Conic (IAC) to zooming cameras. The NIAC can be considered as a particular instance of the set of all possible IAC representing the different possible zooming factors. It is an imaginary object which cannot be observed directly. We proposed several algorithms for its determination. The method requires three or four views of a planar grid, depending on the camera model adopted. It decouples the camera parameters into three sub-sets with constant number of parameters: the first one containing the intrinsic parameters independent to zooming, the second one containing the remaining intrinsic parameters (focal lengths) for each view, and a third one containing the extrinsic parameters. The different algorithms proposed accommodate the different types of cameras (zero or non-zero skew). Experiments with synthetic and real data showed the novel method is more accurate than other plane based calibration methods which do not consider such invariance properties.

In the case of reconstruction using HS, we have proposed several contributions which increase the accuracy of the standard implementation, and also open up the possibility of reconstructing a wider class of objects.

The first main contribution we have proposed in this field is a method to reconstruct optimally the surface normal at each surface point. This replaces the standard solution based on Singular Value Decomposition (SVD) which had an algebraic basis and lacked of physical meaning. Our method is based on the minimisation of a novel distance that we have called the radiometric distance. Effectively, minimising the radiometric distance is equivalent to minimising the modification in intensities to be applied in each image in order to satisfy exactly the HS constraint at each surface point. The solution is simple, and it has been shown that it provides a Maximum Likelihood (ML) estimate in the case of standard Gaussian additive noise conditions. In addition, we addressed the problem of image saturations due to specular highlights. Experiments with synthetic data confirmed the superiority of the radiometric constraint over the algebraic one. In the case of experiments with real data, the novel measure proved able to reconstruct accurately the object geometry, although the improvement was not as obvious compared to the algebraic solution, in the case of our particular experimental setup.

Our second contribution in HS is to show that the standard HS implementation based on indi-

vidual pixel measurements is biased in the case of rough or textured surfaces. We proposed an alternative measure which can be used for the reconstruction of both types of surfaces and is compatible with the most recent definition of Bidirectional Reflectance Distribution Function (BRDF) for locally non-convex surfaces. We demonstrated the applicability of the novel constraint defined on several real test objects.

Throughout the second part of the thesis, a variety of real objects were considered. Some were textured, others almost uniform. Regarding their surface properties, some were smooth, others were rough. Almost all objects were specular. In all cases, the reconstruction obtained appeared accurate and realistic. It is important to mention again that no assumption regarding the surface properties has been made at any stage of the reconstruction. This illustrates the great potential of HS for the reconstruction of surfaces.

8.2 Future work

In the stratified representation of invariants shown for camera calibration, the NIAC occupied the highest position after the IAC and the VP. One limitation of the zooming camera calibration method based on the NIAC is that it assumes that the principal point remains fixed while zooming. This is not the case of all camera technologies. However, we have observed that the assumption is reasonable if the aim of camera calibration is to compute the overall projection matrix rather than the individual camera parameters. If these parameters must be computed separately and accurately, it may be necessary to consider other methods, depending on the particular camera behaviour.

One can wonder how far it is possible to go in the hierarchy of invariants. In particular, it could be tempting to try and incorporate the coordinates of the principal point into a novel invariant, thus extending the invariance properties to another level. In the context of plane-based camera calibration, including the coordinates of the principal points makes camera calibration more complicated because it can no longer be performed from several views of a single planar calibration plane (n images of a single plane provide only $2n$ constraints on the intrinsic parameters, therefore there can be at most only one variable parameter among all intrinsic parameters). This implies that if such an invariant was to be considered, a more complex cal-

ibration grid should also be used, which would make calibration more complicated to use in practice.

One possible line of research would be to define invariants which incorporate lens distortion. In this thesis, we have considered only invariants in the case of an undistorted pin-hole camera. If the camera deviates from such a model, the distortion has first been corrected by applying other methods, and then our methods have been applied to the undistorted images. A more efficient framework would define invariants which incorporate the lens distortion. In [155], Tsai has already considered an invariant to the radial distortion but only in the case of camera calibration from a single image. In the case of multiple images separated by a translation, a free motion, or a free motion plus zooming, new invariants must be defined.

In the field of HS, we have proposed an optimum solution to the normal reconstruction problem. However we believe that there remains scope for improvement in solving the correspondence problem more accurately. The current solution to this problem is still based on SVD, and although it lead to good results, it seems reasonable to think that this aspect of reconstruction could benefit from more sophisticated techniques, which find correspondences in a statistically optimum manner.

If we consider the applicability of the method, the range of objects covered is quite large. In addition to smooth uniform objects, we have extended the applicability of the method to objects with textured or rough surfaces. The solution proposed takes into account local inter-reflections, however global inter-reflections remain problematic, because of the large scale at which such effects can occur and the difficulty to detect them. Another class of objects that cannot be reconstructed by HS is the case of transparent objects.

Another line of research that we have already started exploring is the development of volumetric implementations of HS. The current multi-ocular implementations of HS treat the reconstruction problem independently along each line of the grid. We believe that HS could benefit significantly from the use of methods which solve simultaneously the correspondence and reconstruction problem in 3D. In this framework, the problem could be formulated as finding a surface which maximises the support measure over its surface. This would allow to eliminate the fronto-parallel assumption imposed during computation of the surface depth, because the ambiguity could now be resolved by imposing more appropriate constraints to the surface evo-

lution in space. In addition, such a procedure would use optimally the double output resulting from the HS constraint (surface normal orientation and depth at each point). So far only normal orientation has been considered for the generation of the final 3D model.

Appendices

Appendix A

Camera position estimation

This chapter describes different methods for the computation of the camera position in the case that the intrinsic and orientation parameters have already been estimated. This problem has been labelled Problem 2 in Section 3.3. Two methods are presented: a linear and a non-linear method. Typically the linear method is computed first because of its simplicity, and is followed by the non-linear method which gives a refined solution.

A.1 Linear solution

Once the intrinsic parameters and orientation are known, the position (or translation vector $\boldsymbol{t}$) can be computed from point correspondences. Since $\boldsymbol{P}_i$ are real points in the scene which are not located at infinity, their last coordinate is non-zero, and the homogeneous coordinates can be scaled so that the fourth coordinate is unity. Under this assumption, Eq. (2.7) can be written in the form

$$\boldsymbol{p}_i \sim K[R|\mathbf{0}]\boldsymbol{P}_i + K\boldsymbol{t}\,.$$

The two terms in the previous equation are equal up to a scale factor, that is their cross product is zero, and it follows that

$$\boldsymbol{p}_i \times (K[R|\mathbf{0}]\boldsymbol{P}_i + K\boldsymbol{t}) = [\boldsymbol{p}_i]_\times (K[R|\mathbf{0}]\boldsymbol{P}_i + K\boldsymbol{t}) = \mathbf{0}\,.$$

In the previous expression, for convenience, the cross product has been expressed in terms of the skew symmetric matrix, which for the vector $\boldsymbol{p}_i = (p_{i_1}, p_{i_2}, p_{i_3})^\top$ is defined by

$$[\boldsymbol{p}_i]_\times = \begin{bmatrix} 0 & -p_{i_3} & p_{i_2} \\ p_{i_3} & 0 & -p_{i_1} \\ -p_{i_2} & p_{i_1} & 0 \end{bmatrix} .$$

The previous equation can be written in the form

$$B_i \boldsymbol{t} = \boldsymbol{c}_i \quad \text{with} \quad B_i = [\boldsymbol{p}_i]_\times K \quad \text{and} \quad \boldsymbol{c}_i = -[\boldsymbol{p}_i]_\times K[R|\mathbf{0}]\boldsymbol{P}_i \,. \tag{A.1}$$

This defines a system of three equations. However, since the skew symmetric matrix has rank 2, the equations are not linearly independent, thus the third equation can for example be omitted. From a set of n point correspondences, a $2n \times 3$ matrix B and a $2n$ vector $\boldsymbol{c}$ are obtained by stacking up the matrices B_i and vectors $\boldsymbol{c}_i$ respectively for each correspondence. The vector $\boldsymbol{t}$ is then computed by solving the linear system $B\boldsymbol{t} = \boldsymbol{c}$.

As the system has 3 unknowns, and each point correspondence leads to two equations, a minimum solution is obtained from $1\frac{1}{2}$ points. The 3×3 matrix B has rank 3, so it is invertible and $\boldsymbol{t} = B^{-1}\boldsymbol{c}$. If only one point is available the camera calibration is still possible but up to an overall scale factor.

In practice, the system is usually over-determined, and an approximate solution $\boldsymbol{t}$ that minimises $\|B\boldsymbol{t} - \boldsymbol{c}\|$ is sought. The matrix B having full-rank, a least-square solution is obtained by computing its pseudo-inverse [72]. The procedure is described in Algorithm 5.

Algorithm 5 Basic linear computation of $\boldsymbol{t}$

1. For each world to image point correspondences $\boldsymbol{P}_i$ and $\boldsymbol{p}_i$, compute the matrix B_i and the vector $\boldsymbol{c}_i$ from equation (A.1).
2. Assemble all the matrices B_i into a single matrix B, and all the vectors $\boldsymbol{c}_i$ into a single vector $\boldsymbol{c}$.
3. Compute the pseudo-inverse of B, by the formula $B^+ = (B^\top B)^{-1} B^\top$.
4. Obtain $\boldsymbol{t} = B^+ \boldsymbol{c}$.

A.2 Minimisation of a geometric distance

In order to refine the solution obtained previously, we search for the translation vector $\boldsymbol{t}$ which minimises the sum of squared geometric errors defined by the distance between the projection of 3D points and the corresponding image points in the image plane

$$d'_{\text{geom}}(\boldsymbol{p}_i, K[R|\boldsymbol{t}]\boldsymbol{P}_i) = \|\boldsymbol{p}_i - K[R|\boldsymbol{t}]\boldsymbol{P}_i\| \,. \tag{A.2}$$

A suitable algorithm to solve such a non-linear minimisation problem is for example the Levenberg-Marquardt (LM) algorithm [112].

Appendix B

Approximation of the variance of the geometric distance

In this section, error propagation is used to compute the variance of the geometric distance $d_{\text{geom}}(\boldsymbol{v}, \boldsymbol{l})$ defined in Eq. (3.7). For simplicity, it is assumed that the uncertainty in $d_{\text{geom}}(\boldsymbol{v}, \boldsymbol{l})$ results only from the uncertainty in the measure of the image line coordinates $\boldsymbol{l}$, *i.e.* the coordinates of the associated Vanishing Point (VP) $\boldsymbol{v}$ is assumed to be known exactly. The motivations for such an approximation are explained in Section 3.3.3.

We start by making some statistical considerations regarding the distribution of the end points of image lines, in order to compute the covariance matrix of their coordinates. It is assumed that the coordinates of end points $\boldsymbol{p}_i = (x_i, y_i, 1)^\top$ follow a Gaussian distribution with mean $\bar{\boldsymbol{p}}_i = (\bar{x}_i, \bar{y}_i, 1)^\top$ and standard deviation σ for each coordinates, which are assumed independent. Under these assumptions, a line $\boldsymbol{l} \sim \boldsymbol{p}_i \times \boldsymbol{p}_j$ with end points $\boldsymbol{p}_i = (x_i, y_i, 1)^\top$ and $\boldsymbol{p}_j = (x_j, y_j, 1)^\top$ depends on the distribution of the vector $(x_i, x_j, y_i, y_j)^\top$, which has mean $(\bar{x}_i, \bar{x}_j, \bar{y}_i, \bar{y}_j)^\top$ and covariance matrix $\sigma^2 I$, where the matrix I is the identity matrix. The function which maps the vector $(x_i, x_j, y_i, y_j)^\top$ to the coordinates of the image line $\boldsymbol{l} = (y_i - y_j, x_j - x_i, x_i y_j - x_j y_i)^\top$, has a Jacobian matrix J_0 evaluated at $(\bar{x}_i, \bar{x}_j, \bar{y}_i, \bar{y}_j)^\top$ which is equal to:

$$J_0 = \begin{bmatrix} 0 & 0 & 1 & -1 \\ -1 & 1 & 0 & 0 \\ \bar{y}_j & -\bar{y}_i & -\bar{x}_j & \bar{x}_i \end{bmatrix}.$$

It results from error propagation (see for example [72], pp 123–125) that the coordinates of the line form a random variable $\boldsymbol{l}$ with mean $\bar{\boldsymbol{l}} = (\bar{y}_i - \bar{y}_j, \bar{x}_j - \bar{x}_i, \bar{x}_i\bar{y}_j - \bar{x}_j\bar{y}_i)^\top$ and covariance matrix

$$\Sigma = \sigma^2 J_0 I J_0^\top = \sigma^2 \begin{bmatrix} 2 & 0 & -\bar{x}_i - \bar{x}_j \\ 0 & 2 & -\bar{y}_i - \bar{y}_j \\ -\bar{x}_i - \bar{x}_j & -\bar{y}_i - \bar{y}_j & \bar{x}_i^2 + \bar{x}_j^2 + \bar{y}_i^2 + \bar{y}_j^2 \end{bmatrix}.$$

Now the function which maps the coordinates of the image line $\boldsymbol{l} = (a, b, c)^\top$ to the geometric distance $d_{\text{geom}}(\boldsymbol{v}, \boldsymbol{l})$, given a known VP $\boldsymbol{v} = (u, v, w)^\top$, is considered. The Jacobian matrix of this function, evaluated at $\bar{\boldsymbol{l}} = (\bar{y}_i - \bar{y}_j, \bar{x}_j - \bar{x}_i, \bar{x}_i\bar{y}_j - \bar{x}_j\bar{y}_i)^\top$, is defined by $\boldsymbol{J} = [\frac{\partial d_{\text{geom}}}{\partial a}, \frac{\partial d_{\text{geom}}}{\partial b}, \frac{\partial d_{\text{geom}}}{\partial c}]$. The partial derivatives are given by:

$$\begin{aligned}
\frac{\partial d_{\text{geom}}}{\partial a} &= \frac{1}{\sqrt{a^2+b^2}} \frac{\partial}{\partial a}(a\frac{u}{w} + b\frac{v}{w} + c) + (a\frac{u}{w} + b\frac{v}{w} + c)\frac{\partial}{\partial a}\frac{1}{\sqrt{a^2+b^2}} \\
&= \frac{1}{\sqrt{a^2+b^2}} \frac{u}{w} - \frac{1}{2}(a\frac{u}{w} + b\frac{v}{w} + c)\frac{2a}{\sqrt{a^2+b^2}(a^2+b^2)} \\
&= \frac{1}{\sqrt{a^2+b^2}} \left[\frac{u}{w} - \frac{a(a\frac{u}{w} + b\frac{v}{w} + c)}{a^2+b^2}\right], \\
\frac{\partial d_{\text{geom}}}{\partial b} &= \frac{1}{\sqrt{a^2+b^2}} \left[\frac{v}{w} - \frac{b(a\frac{u}{w} + b\frac{v}{w} + c)}{a^2+b^2}\right], \\
\frac{\partial d_{\text{geom}}}{\partial c} &= \frac{1}{\sqrt{a^2+b^2}} \frac{\partial}{\partial c}(a\frac{u}{w} + b\frac{v}{w} + c) + (a\frac{u}{w} + b\frac{v}{w} + c)\frac{\partial}{\partial c}\frac{1}{\sqrt{a^2+b^2}} \\
&= \frac{1}{\sqrt{a^2+b^2}}.
\end{aligned}$$

It should be noted that the expression of the second partial derivative can be deduced from the expression of the first one by symmetry, interchanging a and b, and u and v. Applying error propagation one more time, it is obtained that d_{geom} is a random variable with variance $\sigma^2_{\text{geom}} = \boldsymbol{J}\Sigma\boldsymbol{J}^\top$. This proves the result stated in Section 3.3.3.

Appendix C

Equation of the IAC

We consider the general model defined in Eq. (2.5) for the calibration matrix K, *i.e.*

$$K = \begin{bmatrix} f & -f\cot\theta & u_0 \\ & \frac{fr}{\sin\theta} & v_0 \\ & & 1 \end{bmatrix} . \tag{C.1}$$

In matrix form, the Image of the Absolute Conic (IAC) is represented algebraically by the equation

$$\boldsymbol{p}^\top \omega \boldsymbol{p} = 0 \,, \tag{C.2}$$

where $\omega = K^{-\top}K^{-1}$ is the conic coefficient matrix.

We have

$$K^{-1} = \frac{1}{f}\begin{bmatrix} 1 & \frac{\cos\theta}{r} & -u_0 - \frac{v_0\cos\theta}{r} \\ & \frac{\sin\theta}{r} & -\frac{v_0\sin\theta}{r} \\ & & f \end{bmatrix} , \tag{C.3}$$

and it follows that

$$\omega \sim K^{-\top}K^{-1} \tag{C.4}$$

$$\sim \begin{bmatrix} 1 & & \\ \frac{\cos\theta}{r} & \frac{\sin\theta}{r} & \\ -u_0 - \frac{v_0\cos\theta}{r} & -\frac{v_0\sin\theta}{r} & f \end{bmatrix} \begin{bmatrix} 1 & \frac{\cos\theta}{r} & -u_0 - \frac{v_0\cos\theta}{r} \\ & \frac{\sin\theta}{r} & -\frac{v_0\sin\theta}{r} \\ & & f \end{bmatrix} \tag{C.5}$$

$$= \begin{bmatrix} 1 & \frac{\cos\theta}{r} & -u_0 - \frac{v_0\cos\theta}{r} \\ \frac{\cos\theta}{r} & \frac{1}{r^2} & -\frac{v_0}{r^2} - \frac{u_0\cos\theta}{r} \\ -u_0 - \frac{v_0\cos\theta}{r} & -\frac{v_0}{r^2} - \frac{u_0\cos\theta}{r} & u_0^2 + 2u_0v_0\frac{\cos\theta}{r} + \frac{v_0^2}{r^2} + f^2 \end{bmatrix}. \tag{C.6}$$

Substituting the expression of ω in Eq. (C.2) and noting $\boldsymbol{p} = [u, v, 1]^\top$, the following equation is obtained after simplification:

$$(u-u_0)^2 + \frac{1}{r^2}(v-v_0)^2 + 2\frac{cos\theta}{r}(u-u_0)(v-v_0) = -f^2\,. \tag{C.7}$$

Appendix D

Equation of the perpendicular bissector to a chord on the IAC

We consider a pair of images of circular points $\boldsymbol{P} = \mathsf{H}\boldsymbol{I} = \boldsymbol{h}_1 + \mathrm{i}\boldsymbol{h}_2$ and $\boldsymbol{Q} = \mathsf{H}\boldsymbol{J} = \boldsymbol{h}_1 - \mathrm{i}\boldsymbol{h}_2$ on the the Image of the Absolute Conic (IAC), with $\boldsymbol{h}_1 = [h_{11}, h_{21}, h_{31}]^\top$ and $\boldsymbol{h}_2 = [h_{12}, h_{22}, h_{32}]^\top$. We want to compute the equation of the perpendicular bisector to this chord after the image transformation T has been applied.

We first observe that the mid-point of $[\boldsymbol{PQ}]$ is the point:

$$\boldsymbol{M} \sim \frac{1}{h_{31}^2 + h_{32}^2}(h_{31}\boldsymbol{h}_1 + h_{32}\boldsymbol{h}_2) = \begin{bmatrix} m_1 \\ m_2 \\ 1 \end{bmatrix}, \tag{D.1}$$

and the direction of the line $(\boldsymbol{PQ})$ is represented by the Point at Infinity (PI):

$$\boldsymbol{D} \sim h_{32}\boldsymbol{h}_1 - h_{31}\boldsymbol{h}_2 = \begin{bmatrix} d_1 \\ d_2 \\ 0 \end{bmatrix}. \tag{D.2}$$

This can be verified by noting that the points $\boldsymbol{M}$, $\boldsymbol{D}$, $\boldsymbol{P}$ and $\boldsymbol{Q}$ are aligned (they are linear combinations of the base vectors $\boldsymbol{h}_1$ and $\boldsymbol{h}_2$) and that they are harmonic (their cross ratio is -1). It should be noted that the denominator in the expression of M is non-zero if and only if the optical axis of the camera is not orthogonal to the image plane. Four new parameters m_1, m_2, d_1 and d_2 have been introduced in the two previous equations.

After transformation by T, $\boldsymbol{M}$ and $\boldsymbol{D}$ are mapped respectively into

$$\boldsymbol{M}' = T \begin{bmatrix} m_1 \\ m_2 \\ 1 \end{bmatrix} = \begin{bmatrix} m_1 + t_1 m_2 + t_2 \\ t_3 m_2 + t_4 \\ 1 \end{bmatrix}, \tag{D.3}$$

and

$$\boldsymbol{D}' = T \begin{bmatrix} d_1 \\ d_2 \\ 0 \end{bmatrix} = \begin{bmatrix} d_1 + t_1 d_2 \\ t_3 d_2 \\ 0 \end{bmatrix}. \tag{D.4}$$

Noting that a normal vector to $(\boldsymbol{P}'\boldsymbol{Q}')$ is $\boldsymbol{N} = [-t_3 d_2, d_1 + t_1 d_2, 0]^\top$, we conclude that the perpendicular bisector to the chord is represented by the equation

$$\boldsymbol{l} = \begin{bmatrix} -(d_1 + t_1 d_2) \\ -t_3 d_2 \\ (m_1 + t_1 m_2 + t_2)(d_1 + t_1 d_2) + (t_3 m_2 + t_4) t_3 d_2 \end{bmatrix}. \tag{D.5}$$

Appendix E

Simplification of the cost function based on the radiometric distance

We would like to compute the values $\boldsymbol{n}$ and $\{\hat{i}_{l_j}\}_j$ which minimise the cost function

$$F(\boldsymbol{n}, \{\hat{i}_{l_j}\}_j) = \sum_j \left[(\hat{i}_{l_j} - i_{l_j})^2 + \left(\frac{\boldsymbol{s}_{l_j} \cdot \boldsymbol{n}}{\boldsymbol{s}_{r_j} \cdot \boldsymbol{n}} \hat{i}_{l_j} - i_{r_j} \right)^2 \right] . \tag{E.1}$$

We start by writing the partial derivatives of F with respect to $\hat{i}_{l_j}$:

$$\forall j, \quad \frac{\partial F(\boldsymbol{n}, \{\hat{i}_{l_j}\}_j)}{\partial \hat{i}_{l_j}} = 2(\hat{i}_{l_j} - i_{l_j}) + 2 \frac{\boldsymbol{s}_{l_j} \cdot \boldsymbol{n}}{\boldsymbol{s}_{r_j} \cdot \boldsymbol{n}} \left(\frac{\boldsymbol{s}_{l_j} \cdot \boldsymbol{n}}{\boldsymbol{s}_{r_j} \cdot \boldsymbol{n}} \hat{i}_{l_j} - i_{r_j} \right) , \tag{E.2}$$

and observe that they are equal to zero at the optimum value, which leads to the constraints:

$$\forall j, \quad \hat{i}_{l_j} + \left(\frac{\boldsymbol{s}_{l_j} \cdot \boldsymbol{n}}{\boldsymbol{s}_{r_j} \cdot \boldsymbol{n}} \right)^2 \hat{i}_{l_j} = i_{l_j} + \frac{\boldsymbol{s}_{l_j} \cdot \boldsymbol{n}}{\boldsymbol{s}_{r_j} \cdot \boldsymbol{n}} i_{r_j} . \tag{E.3}$$

From each constraint, we can deduce the expression of $\hat{i}_{l_j}$ at the optimum:

$$\forall j, \quad \hat{i}_{l_j} = \left(i_{l_j} + \frac{\boldsymbol{s}_{l_j} \cdot \boldsymbol{n}}{\boldsymbol{s}_{r_j} \cdot \boldsymbol{n}} i_{r_j} \right) \frac{(\boldsymbol{s}_{r_j} \cdot \boldsymbol{n})^2}{(\boldsymbol{s}_{l_j} \cdot \boldsymbol{n})^2 + (\boldsymbol{s}_{r_j} \cdot \boldsymbol{n})^2} , \tag{E.4}$$

which after simplification can be written

$$\forall j, \quad \hat{i}_{l_j} = \frac{i_{l_j} (\boldsymbol{s}_{r_j} \cdot \boldsymbol{n})^2 + i_{r_j} (\boldsymbol{s}_{l_j} \cdot \boldsymbol{n})(\boldsymbol{s}_{r_j} \cdot \boldsymbol{n})}{(\boldsymbol{s}_{l_j} \cdot \boldsymbol{n})^2 + (\boldsymbol{s}_{r_j} \cdot \boldsymbol{n})^2} . \tag{E.5}$$

We also compute the two following terms which will be useful next:

$$\forall j, \quad \begin{cases} \hat{i}_{l_j} - i_{l_j} &= -\frac{(\boldsymbol{s}_{l_j} \cdot \boldsymbol{n})[(i_{l_j} \boldsymbol{s}_{l_j} - i_{r_j} \boldsymbol{s}_{r_j}) \cdot \boldsymbol{n}]}{(\boldsymbol{s}_{l_j} \cdot \boldsymbol{n})^2 + (\boldsymbol{s}_{r_j} \cdot \boldsymbol{n})^2} , \\ \frac{\boldsymbol{s}_{l_j} \cdot \boldsymbol{n}}{\boldsymbol{s}_{r_j} \cdot \boldsymbol{n}} \hat{i}_{l_j} - i_{r_j} &= \frac{(\boldsymbol{s}_{r_j} \cdot \boldsymbol{n})[(i_{l_j} \boldsymbol{s}_{l_j} - i_{r_j} \boldsymbol{s}_{r_j}) \cdot \boldsymbol{n}]}{(\boldsymbol{s}_{l_j} \cdot \boldsymbol{n})^2 + (\boldsymbol{s}_{r_j} \cdot \boldsymbol{n})^2} . \end{cases} \tag{E.6}$$

In addition, the partial derivative of F with respect to the surface normal $\boldsymbol{n}$ is

$$\frac{\partial F(\boldsymbol{n}, \{\hat{i}_{l_j}\}_j)}{\partial \boldsymbol{n}} = 2 \sum_j \left[\left(\frac{\boldsymbol{s}_{l_j} \cdot \boldsymbol{n}}{\boldsymbol{s}_{r_j} \cdot \boldsymbol{n}} \hat{i}_{l_j} - i_{r_j} \right) \hat{i}_{l_j} \frac{\partial}{\partial \boldsymbol{n}} \left(\frac{\boldsymbol{s}_{l_j} \cdot \boldsymbol{n}}{\boldsymbol{s}_{r_j} \cdot \boldsymbol{n}} \right) \right] . \tag{E.7}$$

which, after observing that $\frac{\partial (\boldsymbol{s} \cdot \boldsymbol{n})}{\partial \boldsymbol{n}} = \boldsymbol{s}$, simplifies to:

$$\frac{\partial F(\boldsymbol{n}, \{\hat{i}_{l_j}\}_j)}{\partial \boldsymbol{n}} = 2 \sum_j \left[\left(\frac{\boldsymbol{s}_{l_j} \cdot \boldsymbol{n}}{\boldsymbol{s}_{r_j} \cdot \boldsymbol{n}} \hat{i}_{l_j} - i_{r_j} \right) \hat{i}_{l_j} \frac{(\boldsymbol{s}_{r_j} \cdot \boldsymbol{n}) \boldsymbol{s}_{l_j} - (\boldsymbol{s}_{l_j} \cdot \boldsymbol{n}) \boldsymbol{s}_{r_j}}{(\boldsymbol{s}_{r_j} \cdot \boldsymbol{n})^2} \right] . \tag{E.8}$$

All variables $\hat{i}_{l_j}$ can be eliminated from this expression by substituting the terms defined in the system of equations (E.6), which results in:

$$\begin{aligned} \frac{\partial F(\boldsymbol{n}, \{\hat{i}_{l_j}\}_j)}{\partial \boldsymbol{n}} = 2 \sum_j \Bigg[& \frac{(\boldsymbol{s}_{r_j} \cdot \boldsymbol{n})[(i_{l_j} \boldsymbol{s}_{l_j} - i_{r_j} \boldsymbol{s}_{r_j}) \cdot \boldsymbol{n}]}{(\boldsymbol{s}_{l_j} \cdot \boldsymbol{n})^2 + (\boldsymbol{s}_{r_j} \cdot \boldsymbol{n})^2} \cdots \\ & \left(i_{l_j} - \frac{(\boldsymbol{s}_{l_j} \cdot \boldsymbol{n})[(i_{l_j} \boldsymbol{s}_{l_j} - i_{r_j} \boldsymbol{s}_{r_j}) \cdot \boldsymbol{n}]}{(\boldsymbol{s}_{l_j} \cdot \boldsymbol{n})^2 + (\boldsymbol{s}_{r_j} \cdot \boldsymbol{n})^2} \right) \frac{(\boldsymbol{s}_{r_j} \cdot \boldsymbol{n}) \boldsymbol{s}_{l_j} - (\boldsymbol{s}_{l_j} \cdot \boldsymbol{n}) \boldsymbol{s}_{r_j}}{(\boldsymbol{s}_{r_j} \cdot \boldsymbol{n})^2} \Bigg] . \end{aligned} \tag{E.9}$$

After simplification, we finally obtain:

$$\frac{\partial F(\boldsymbol{n}, \{\hat{i}_{l_j}\}_j)}{\partial \boldsymbol{n}} = 2 \sum_j \frac{[(i_{l_j} \boldsymbol{s}_{l_j} - i_{r_j} \boldsymbol{s}_{r_j}) \cdot \boldsymbol{n}][(i_{l_j} \boldsymbol{s}_{r_j} + i_{r_j} \boldsymbol{s}_{l_j}) \cdot \boldsymbol{n}][(\boldsymbol{s}_{r_j} \cdot \boldsymbol{n}) \boldsymbol{s}_{l_j} - (\boldsymbol{s}_{l_j} \cdot \boldsymbol{n}) \boldsymbol{s}_{r_j}]}{[(\boldsymbol{s}_{l_j} \cdot \boldsymbol{n})^2 + (\boldsymbol{s}_{r_j} \cdot \boldsymbol{n})^2]^2} . \tag{E.10}$$

At the optimum, $\boldsymbol{n}$ must satisfy the constraint $\frac{\partial F(\boldsymbol{n}, \{\hat{i}_{l_j}\}_j)}{\partial \boldsymbol{n}} = \boldsymbol{0}$. This defines a system of three non-linear equations with unknowns the three components of $\boldsymbol{n}$ (there are actually only two unknowns since the scale of $\boldsymbol{n}$ can be arbitrary). We could solve directly this system.

Alternatively we can define an auxiliary cost function G which depends only on $\boldsymbol{n}$, by substituting the terms defined in the system of equations (E.6), into the original cost function F, which leads to:

$$G(\boldsymbol{n}) = \sum_j \frac{[(i_{l_j} \boldsymbol{s}_{l_j} - i_{r_j} \boldsymbol{s}_{r_j}) \cdot \boldsymbol{n}]^2}{(\boldsymbol{s}_{l_j} \cdot \boldsymbol{n})^2 + (\boldsymbol{s}_{r_j} \cdot \boldsymbol{n})^2} . \tag{E.11}$$

It is easy to verify that $\frac{\partial F(\boldsymbol{n}, \{\hat{i}_{l_j}\}_j)}{\partial \boldsymbol{n}} = \frac{\partial G(\boldsymbol{n})}{\partial \boldsymbol{n}}$, thus F and G have the same minimum. Therefore, the solution to the original problem can be found by minimising G. We found this preferable to solving the non-linear system of equations defined in $\frac{\partial F(\boldsymbol{n}, \{\hat{i}_{l_j}\}_j)}{\partial \boldsymbol{n}} = \boldsymbol{0}$, because the equations involved are simpler, and also because it is more similar to the problems solved in other chapters, which means that similar methods can be applied. Such a solution can be computed by using a non-linear minimisation technique such as the Levenberg-Marquardt (LM) algorithm.

Bibliography

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

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

[3] M. Agrawal and L. S. Davis. Camera calibration using spheres: a semi-definite programming approach. In *Proc. IEEE International Conference on Computer Vision*, volume 2, pages 782–789, October 2003.

[4] A. S. Aguado, E. Montiel, and M. S. Nixon. Invariant characterization of the Hough transform for pose estimation of arbitrary shapes. In *Proc. British Machine Vision Conference*, volume 2, pages 785–794, 2000.

[5] M. T. Ahmed and A. A. Farag. A neural approach to zoom-lens camera calibration from data with outliers. *Image and Vision Computing*, 20(9-10):619–630, August 2002.

[6] J. Y. Aloimonos. Perspective approximations. *Image and Vision Computing*, 8(3):179–192, August 1990.

[7] M. Armstrong, A. Zisserman, and R. I. Hartley. Self-calibration from image triplets. In *Proc. European Conference on Computer Vision*, pages 3–16, 1996.

[8] K. S. Arun, T. S. Huang, and S. D. Blostein. Least-squares fitting of two 3-d point sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(5):698–700, 1987.

[9] A. Bartoli, R. Hartley, and F. Kahl. Motion from 3d line correspondences: Linear and non-linear solutions. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 477–484, 2003.

[10] A. Basu and K. Ravi. Active camera calibration using pan, tilt and roll. *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics*, 27(3):559–566, 1997.

[11] A. Baumberg. Reliable feature matching across widely separated views. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 774–781, 2000.

[12] P. Beardsley and D. Murray. Camera calibration using vanishing points. In *Proc. British Machine Vision Conference*, pages 416–425, 1992.

[13] P. Beardsley, D. Murray, and A. Zisserman. Camera calibration using multiple images. In *Proc. European Conference on Computer Vision*, pages 312–320, 1992.

[14] P. A. Beardsley, P. H. S. Torr, and A. Zisserman. 3D model acquisition from extended image sequences. In *Proc. European Conference on Computer Vision*, volume II, pages 683–695, 1996.

[15] P. N. Belhumeur, D. J. Kriegman, and A. L. Yuille. The bas-relief ambiguity. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 1060–1066, 1997.

[16] R. Bhotika, D. J. Fleet, and K. N. Kutulakos. A probabilistic theory of occupancy and emptiness. In *Proc. European Conference on Computer Vision*, volume III, pages 112–132, 2002.

[17] T. Bonfort and P. Sturm. Voxel carving for specular surfaces. In *Proc. IEEE International Conference on Computer Vision*, October 2003.

[18] N. A. Borghese and P. Cerveri. Calibrating a video camera pair with a rigid bar. *Pattern Recognition*, 33(1):81–95, 2000.

[19] M. Born and E. Wolf. *Principles of Optics*. Pergamon Press, third edition, 1965.

[20] S. Bougnoux. From projective to Euclidean space under any practical situation, a criticism of self-calibration. In *Proc. IEEE International Conference on Computer Vision*, pages 790–796, 1998.

[21] J.-Y. Bouguet. Camera calibration toolbox for matlab®. Intel Corp., `http://www.vision.caltech.edu/bouguetj/calib_doc/`, August 2005.

[22] E. Boyer and J.-S. Franco. A hybrid approach for computing visual hulls of complex objects. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 695–701, June 2003.

[23] A. Broadhurst, T. Drummond, and R. Cipolla. A probabilistic framework for the Space Carving algorithm. In *Proc. IEEE International Conference on Computer Vision*, pages 388–393, July 2001.

[24] D. C. Brown. Close-range camera calibration. *Photogrammetric Engineering*, 37(8):855–866, 1971.

[25] T. Buchanan. The twisted cubic and camera calibration. *Computer Vision, Graphics, and Image Processing*, 42:130–132, 1988.

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

[27] S. Chaudhuri and A. Rajagopalan. *Depth from defocus: a real aperture imaging approach*. Springer Verlag, 1999.

[28] W. Chen and B. C. Jiang. 3-D camera calibration using vanishing point concept. *Pattern Recognition*, 24(1):57–67, 1991.

[29] Y.-S. Chen, S.-W. Shih, Y.-P. Hung, and C.-S. Fuh. Simple and efficient method of calibrating a motorized zoom lens. *Image and Vision Computing*, 19(14):1099–1110, December 2001.

[30] W. Chojnacki, M. J. Brooks, and A. van den Hengel. Rationalising the renormalisation method of kanatani. *Journal of Mathematical Imaging and Vision*, 14(1):21–38, 2001.

[31] W. Chojnacki, M. J. Brooks, A. van den Hengel, and D. Gawley. Revisiting hartley's normalized eight-point algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(9):1172–1177, 2003.

[32] P. H. Christensen and L. G. Shapiro. Three-dimensional shape from color photometric stereo. *International Journal of Computer Vision*, 13(2):213–227, 1994.

[33] R. Cipolla, D. P. Robertson, and E. G. Boyer. Photobuilder - 3D models of architectural scenes from uncalibrated images. In *Proc. IEEE International Conference on Multimedia Computing and Systems*, volume 1, pages 25–31, 1999.

[34] R. Collins and Y. Tsin. Calibration of an outdoor active camera system. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 528–534, 1999.

[35] R. T. Collins. A space-sweep approach to true multi-image matching. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, page 358, 1996.

[36] R. L. Cook and K. E. Torrance. A reflectance model for computer graphics. *ACM Transactions on Graphics*, 1(1):7–24, 1982.

[37] W. B. Culbertson, T. Malzbender, and G. G. Slabaugh. Generalized voxel coloring. In *Proc. IEEE International Conference on Computer Vision: International Workshop on Vision Algorithms*, pages 100–115, 1999.

[38] K. Daniilidis and J. Ernst. Active intrinsic calibration using vanishing points. *Pattern Recognition Letters*, 17:1179–1189, 1996.

[39] J. S. De Bonet and P. A. Viola. Roxels: Responsibility weighted 3D volume reconstruction. In *Proc. IEEE International Conference on Computer Vision*, volume 1, pages 418–425, 1999.

[40] F. Devernay and O. D. Faugeras. Straight lines have to be straight. *Machine Vision and Applications*, 13(1):14–24, 2001.

[41] O. Drbohlav. Helmholtz stereopsis on non-uniform and non-smooth surfaces. Private communication, September 2003.

[42] O. Drbohlav and R. Šára. Specularities reduce ambiguity of uncalibrated photometric stereo. In *Proc. European Conference on Computer Vision*, volume II, pages 46–62, 2002.

[43] T. Echigo. A camera calibration technique using three sets of parallel lines. *Machine Vision and Applications*, 3(3):159–167, 1990.

[44] P. Eisert, E. Steinbach, and B. Girod. Multi-hypothesis, volumetric reconstruction of 3-d objects from multiple calibrated camera views. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 3509–3512, 1999.

[45] O. Faugeras, B. Hotz, H. Mathieu, T. Viéville, Z. Zhang, P. Fua, E. Théron, L. Moll, G. Berry, J. Vuillemin, P. Bertin, and C. Proy. Real-time correlation-based stereo : algorithm, implementations and applications. Technical Report RR-2013, INRIA, 1993.

[46] O. D. Faugeras. *Three-Dimensional Computer Vision: A Geometric Viewpoint*. The MIT Press, 1993.

[47] O. D. Faugeras. Stratification of three-dimensional vision: projective, affine, and metric representations. *Journal of Optical Society of America A*, 12(3):465–484, March 1995.

[48] O. D. Faugeras and Q.-T. Luong. *The geometry of multiple images*. The MIT Press, 2001.

[49] O. D. Faugeras, Q.-T. Luong, and S. J. Maybank. Camera self-calibration: Theory and experiments. In *Proc. European Conference on Computer Vision*, pages 321–334, 1992.

[50] O. D. Faugeras and S. J. Maybank. Motion from point matches: Multiplicity of solutions. *International Journal of Computer Vision*, 4(3):225–246, 1990.

[51] O. D. Faugeras and G. Toscani. The calibration problem for stereo. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 15–20, 1986.

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

[53] A. W. Fitzgibbon and A. Zisserman. Automatic camera recovery for closed or open image sequences. In *Proc. European Conference on Computer Vision*, volume I, pages 311–326, 1998.

[54] D. A. Forsyth and J. Ponce. *Computer Vision, A Modern Approach*. Prentice Hall, 2003.

[55] R. T. Frankot and R. Chellappa. A method for enforcing integrability in shape from shading algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(4):439–451, 1988.

[56] A. Fusiello, V. Roberto, and E. Trucco. Symmetric stereo with multiple windowing. *International Journal of Pattern Recognition and Artificial Intelligence*, 14(8):1053–1066, 2000.

[57] A. Fusiello, E. Trucco, and A. Verri. A compact algorithm for rectification of stereo pairs. *Machine Vision and Applications*, 12(1):16–22, 2000.

[58] S. Ganapathy. Decomposition of transformation matrices for robot vision. *Pattern Recognition Letters*, 2(6):401–412, December 1984.

[59] B. García and P. Brunet. 3D reconstruction with projective octrees and epipolar geometry. In *Proc. IEEE International Conference on Computer Vision*, pages 1067–1072, 1998.

[60] G. H. Golub and C. F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, 1983.

[61] P. Gurdjos, A. Crouzil, and R. Payrissat. Another way of looking at plane-based calibration: the centre circle constraint. In *Proc. European Conference on Computer Vision*, volume IV, pages 252–266, 2002.

[62] P. Gurdjos and R. Payrissat. Plane-based calibration of a camera with varying focal length: the centre line constraint. In *Proc. British Machine Vision Conference*, volume 2, pages 623–632, 2001.

[63] E. L. Hall, J. B. K. Tio, C. A. McPherson, and F. A. Sadjadi. Measuring curved surfaces for robot vision. *Computer*, 15(12):42–54, December 1982.

[64] C. Harris and M. Stephens. A combined corner and edge detector. In *Alvey Vision Conference*, pages 147–151, University of Manchester, 1988.

[65] R. Hartley, R. Gupta, and T. Chang. Stereo from uncalibrated cameras. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 761–764, 1992.

[66] R. I. Hartley. Self-calibration from multiple views with a rotating camera. In *Proc. European Conference on Computer Vision*, volume 1, pages 471–478, 1994.

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

[68] R. I. Hartley. Lines and points in three views and the trifocal tensor. *International Journal of Computer Vision*, 22(2):125–140, 1997.

[69] R. I. Hartley. Computation of the quadrifocal tensor. In *Proc. European Conference on Computer Vision*, volume I, pages 20–35, 1998.

[70] R. I. Hartley. Minimizing algebraic error in geometric estimation problems. In *Proc. IEEE International Conference on Computer Vision*, pages 469–476, January 1998.

[71] R. I. Hartley and P. Sturm. Triangulation. *Computer Vision and Image Understanding*, 68(2):146–157, 1997.

[72] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.

[73] H. Hayakawa. Photometric stereo under a light source with arbitrary motion. *Journal of Optical Society of America A*, 11(11):3079–3089, 1994.

[74] D. J. Heeger and A. D. Jepson. Subspace methods for recovering rigid motion I: Algorithm and implementation. *International Journal of Computer Vision*, 7(2):95–117, 1992.

[75] B. K. P. Horn. *Robot vision*. MIT Press, 1986.

[76] K. Ikeuchi. Determining surface orientations of specular surfaces by using the photometric stereo method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 3(6):661–669, 1981.

[77] E. Izquierdo and V. Guerra. Estimating the essential matrix by efficient linear techniques. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(9):925–935, 2003.

[78] Z. Jankó, O. Drbohlav, and R. Šára. Radiometric calibration of a Helmholtz stereo rig. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 166–171, 2004.

[79] T. Kanade and M. Okutomi. A stereo matching algorithm with an adaptive window: Theory and experiment. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(9):920–932, 1994.

[80] K. Kanatani. *Statistical Optimization for Geometric Computation: Theory and Practice*. Elsevier Science Inc., New York, NY, USA, 1996.

[81] K. N. Kutulakos. Approximate n-view stereo. In *Proc. European Conference on Computer Vision*, volume I, pages 67–83, 2000.

[82] K. N. Kutulakos and S. M. Seitz. A theory of shape by space carving. *Int. J. Computer Vision*, 38(3):199–218, July 2000.

[83] E. P. Lafortune and Y. D. Willems. Using the modified phong reflectance model for physically based rendering. Technical Report CW 197, Department of Computing Science, K.U. Leuven, November, 1994.

[84] A. Laurentini. The visual hull concept for silhouette-based image understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(2):150–162, 1994.

[85] Y. Leedan and P. Meer. Heteroscedastic regression in computer vision: Problems with bilinear constraint. *International Journal of Computer Vision*, 37(2):127–150, 2000.

[86] R. K. Lenz and R. Y. Tsai. Techniques for calibration of the scale factor and image center for high-accuracy 3-D machine vision metrology. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(5):713–720, September 1988.

[87] R. R. Lewis. Making shaders more physically plausible. *Computer Graphics Forum*, 13(2):109–120, 1994.

[88] D. Liebowitz, A. Criminisi, and A. Zisserman. Creating architectural models from images. In *Proc. Eurographics*, volume 18, pages 39–50, 1999.

[89] D. Liebowitz and A. Zisserman. Metric rectification for perspective images of planes. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 482–488, 1998.

[90] Y. Liu and T. S. Huang. A linear algorithm for determining motion and structure from line correspondences. *Computer Vision, Graphics, and Image Processing*, 44(1):35–57, 1988.

[91] H. C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135, September 1981.

[92] S. Magda, D. J. Kriegman, T. Zickler, and P. N. Belhumeur. Beyond Lambert: Reconstructing surfaces with arbitrary BRDFs. In *Proc. IEEE International Conference on Computer Vision*, volume 2, pages 391–398, 2001.

[93] W. N. Martin and J. K. Aggarwal. Volumetric descriptions of objects from multiple views. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 5(2):150–158, March 1983.

[94] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In *Proc. British Machine Vision Conference*, pages 384–393, 2002.

[95] C. Matsunaga and K. Kanatani. Calibration of a moving camera using a planer pattern: Optimal computation, reliability evaluation and stabilization by model selection. In *Proc. European Conference on Computer Vision*, volume II, pages 595–609, 2000.

[96] X. Meng and Z. Hu. A new easy camera calibration technique based on circular points. *Pattern Recognition*, 36(5):1155–1164, May 2003.

[97] T. Moons, L. J. Van Gool, M. V. Diest, and E. J. Pauwels. Affine reconstruction from perspective image pairs obtained by a translating camera. In J.L. Mundy, A. Zisserman, and D.A. Forsyth, editors, *Applications of Invariance in Computer Vision*, volume 825, pages 297–316. Springer–Verlag, 1994.

[98] M. Mühlich and R. Mester. The role of total least squares in motion analysis. In *Proc. European Conference on Computer Vision*, volume II, pages 305–321, 1998.

[99] M. Mühlich and R. Mester. A considerable improvement in non-iterative homography estimation using TLS and equilibration. *Pattern Recognition Letters*, 22(11):1181–1189, 2001.

[100] M. Mühlich and R. Mester. Improving motion and orientation estimation using an equilibrated total least squares approach. In *Proc. International Conference on Image Processing*, volume 2, pages 929–932, 2001.

[101] P. J. Narayanan, P. Rander, and T. Kanade. Constructing virtual worlds using dense stereo. In *Proc. IEEE International Conference on Computer Vision*, pages 3 – 10, January 1998.

[102] S. Nayar, K. Ikeuchi, and T. Kanade. Determining shape and reflectance of hybrid surfaces by photometric sampling. *IEEE Transactions on Robotics and Automation*, 6(4):418–431, August 1990.

[103] S. K. Nayar, K. Ikeuchi, and T. Kanade. Shape from interreflections. *International Journal of Computer Vision*, 6(3):173–195, 1991.

[104] F. E. Nicodemus, J. C. Richmond, J. J. Hsia, I. W. Ginsberg, and T. Limperis. Geometrical consideration and nomenclature for reflectance, 1977. NBS Monograph 160.

[105] M. Okutomi and T. Kanade. A multiple-baseline stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(4):353–363, 1993.

[106] J. Oliensis. A critique of structure-from-motion algorithms. *Computer Vision and Image Understanding*, 80(2):172–214, 2000.

[107] M. A. Penna. Camera calibration: A quick and easy way to determine the scale factor. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(12):1240–1245, December 1991.

[108] B. T. Phong. Illumination for computer generated pictures. *Communications of the ACM*, 18(6):311–317, 1975.

[109] M. Pollefeys and L. Van Gool. Stratified self-calibration with the modulus constraint. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(8):707–724, 1999.

[110] M. Pollefeys, L. Van Gool, M. Vergauwen, F. Verbiest, K. Cornelis, J. Tops, and R. Koch. Visual modeling with a hand-held camera. *International Journal of Computer Vision*, 59(3):207–232, 2004.

[111] M. Pollefeys, R. Koch, and L. J. Van Gool. Self-calibration and metric reconstruction in spite of varying and unknown internal camera parameters. In *Proc. IEEE International Conference on Computer Vision*, pages 90–95, 1998.

[112] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical recipes in C*. Cambridge University Press, second edition, 1992.

[113] P. Pritchett and A. Zisserman. Wide baseline stereo matching. In *Proc. IEEE International Conference on Computer Vision*, pages 754–760, 1998.

[114] A. C. Prock and C. R. Dyer. Towards real-time voxel coloring. In *Proc. DARPA Image Understanding Workshop*, pages 315–321, 1998.

[115] J. H. Reiger and D. T. Lawton. Processing differential image motion. *Journal of Optical Society of America A*, 2:354–359, 1985.

[116] H. Saito and T. Kanade. Shape reconstruction in projective grid space from large number of images. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, June 1999.

[117] J. Salvi, X. Armangué, and J. Batlle. A comparative review of camera calibrating methods with accuracy evaluation. *Pattern Recognition*, 35(7):1617–1635, 2002.

[118] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47(1-3):7–42, 2002.

[119] S. M. Seitz and C. R. Dyer. Photorealistic scene reconstruction by voxel coloring. *International Journal of Computer Vision*, 35(2):151–173, 1999.

[120] J. G. Semple and G. T. Kneebone. *Algebraic Projective Geometry*. Oxford University Press, first edition, 1952.

[121] P. Sen, B. Chen, G. Garg, S. R. Marschner, M. Horowitz, M. Levoy, and H. P. A. Lensch. Dual photography. *Proc. International SIGGRAPH Conference*, 24(3):745–755, 2005.

[122] Y. Seo and K. S. Hong. About the self-calibration of a rotating and zooming camera: Theory and practice. In *Proc. IEEE International Conference on Computer Vision*, volume 1, pages 183–189, 1999.

[123] J. A. Shufelt. Performance evaluation and analysis of vanishing point detection techniques. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(3):282–288, 1999.

[124] G. Slabaugh, B. Culbertson, T. Malzbender, and R. Schafer. A survey of methods for volumetric scene reconstruction from photographs. In *International Workshop on Volume Graphics*, 2001.

[125] G. Slabaugh, T. Malzbender, and W. B. Culbertson. Volumetric warping for voxel coloring on an infinite domain. In *European Workshop on 3D Structure from Multiple Images of Large-Scale Environments*, pages 109–123, 2000.

[126] G. Slabaugh, R. Schafer, and M. Hans. Image-based photo hulls. In *Proc. International Symposium on 3D Data Processing, Visualization and Transmission*, pages 704–708, 2002.

[127] C. Slama. *Manual of Photogrammetry*. American Society of Photogrammetry, fourth edition, 1980.

[128] W. C. Snyder. Reciprocity of the bidirectional reflectance distribution function (BRDF) in measurements and models of structured surfaces. *IEEE Transactions on Geoscience and Remote Sensing*, 36(2):685–691, 1998.

[129] W. C. Snyder. Definition and invariance properties of structured surface BRDF. *IEEE Transactions on Geoscience and Remote Sensing*, 40(5):1032–1037, 2002.

[130] W. C. Snyder. Structured surface bidirectional reflectance distribution function reciprocity: theory and counterexamples. *Applied Optics*, 41(21):4307–4313, 2002.

[131] G. P. Stein. Internal camera calibration using rotation and geometric shapes. Master's thesis, Massachusetts Institute of Technology, 1993.

[132] G. P. Stein. Accurate internal camera calibration using rotation, with analysis of sources of error. In *Proc. IEEE International Conference on Computer Vision*, pages 230–236, 1995.

[133] G. P. Stein and A. Shashua. On degeneracy of linear reconstruction from three views: Linear Line Complex and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(3):244–251, 1999.

[134] D. E. Stevenson and M. M. Fleck. Robot aerobics: four easy steps to a more flexible calibration. In *Proc. IEEE International Conference on Computer Vision*, pages 34–39, 1995.

[135] P. F. Sturm. Critical motion sequences for monocular self-calibration and uncalibrated Euclidean reconstruction. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 1100–1105, 1997.

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

[137] P. F. Sturm. Critical motion sequences for the self-calibration of cameras and stereo systems with variable focal length. *Image and Vision Computing*, 20(5-6):415–426, March 2002.

[138] P. F. Sturm and S. J. Maybank. On plane-based camera calibration: A general algorithm, singularities, applications. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 432–437, 1999.

[139] P. F. Sturm and B. Triggs. A factorization based algorithm for multi-image projective structure and motion. In *Proc. European Conference on Computer Vision*, volume II, pages 709–720, 1996.

[140] R. Szeliski. Rapid octree construction from image sequences. *Computer Vision, Graphics, and Image Processing*, 58(1):23–32, July 1993.

[141] H. D. Tagare and R. J. P. deFigueiredo. A theory of photometric stereo for a class of diffuse non-Lambertian surfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(2):133–152, 1991.

[142] K. Tarabanis, R. Y. Tsai, and D. S. Goodman. Calibration of a computer-controlled robotic vision sensor with a zoom lens. *Computer Vision, Graphics, and Image Processing*, 59(2):226–241, March 1994.

[143] D. Tell and S. Carlsson. Wide baseline point matching using affine invariants computed from intensity profiles. In *Proc. European Conference on Computer Vision*, pages 814–828, 2000.

[144] H. Teramoto and G. Xu. Camera calibration by a single image of balls: From conics to the absolute conic. In *Proc. Asian Conference on Computer Vision*, pages 499–506, 2002.

[145] D. Terzopoulos. Multi-level reconstruction of visual surfaces: Variational principles and finite element representations. A.I. Memo 671, MIT, 1982.

[146] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: a factorization method. *International Journal of Computer Vision*, 9(2):137–154, 1992.

[147] P. H. S. Torr and D. W. Murray. The development and comparison of robust methods for estimating the fundamental matrix. *International Journal of Computer Vision*, 24(3):271–300, 1997.

[148] P. H. S. Torr and A. Zisserman. Robust parameterization and computation of the trifocal tensor. *Image and Vision Computing*, 15:591–605, 1997.

[149] B. Triggs. Matching constraints and the joint image. In *Proc. IEEE International Conference on Computer Vision*, pages 338–343, 1995.

[150] B. Triggs. Autocalibration from planar scenes. In *Proc. European Conference on Computer Vision*, pages 89–105, June 1998.

[151] W. Triggs. Auto-calibration and the absolute quadric. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 609–614, 1997.

[152] W. Triggs, P. F. McLauchlan, R. I. Hartley, and A. Fitzgibbon. Bundle ajustment—a modern synthesis. In *Proc. International Workshop on Vision Algorithms: Theory and Practice*, pages 298–372, 1999.

[153] E. Trucco, R. B. Fisher, and A. W. Fitzgibbon. Direct calibration and data consistency in 3-D laser scanning. In *Proc. British Machine Vision Conference*, pages 489–498, 1994.

[154] E. Trucco and A. Verri. *Introductory Techniques for 3-D Computer Vision*. Prentice Hall, 1998.

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

[156] P. Tu and P. R. S. Mendonça. Surface reconstruction via Helmholtz reciprocity with a single image pair. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 541–547, 2003.

[157] P. Tu, P. R. S. Mendonça, J. Ross, and J. Miller. Surface registration with a Helmholtz reciprocity image pair. In *IEEE Workshop on Color and Photometric Methods in Computer Vision*, 2003.

[158] T. Tuytelaars and L. Van Gool. Matching widely separated views based on affine invariant regions. *International Journal of Computer Vision*, 59(1):61–85, 2004.

[159] L. L. Wang and W. H. Tsai. Computing camera parameters using vanishing-line information from a rectangular parallelepiped. *Machine Vision and Applications*, 3(3):129–141, 1990.

[160] L. L. Wang and W. H. Tsai. Camera calibration by vanishing lines for 3-d computer vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(4):370–376, 1991.

[161] J. Weng, P. Cohen, and M. Herniou. Camera calibration with distortion models and accuracy evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(10):965–980, October 1992.

[162] J. Weng and T. Huang. Motion and structure from line correspondences: Closed-form solution, uniqueness, and optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(3):318–335, 1992.

[163] R. G. Willson. Modeling and calibration of automated zoom lenses. In *Proc. SPIE #2350: Videometrics III*, pages 170–186, October 1994.

[164] R. G. Willson. *Modeling and Calibration of Automated Zoom Lenses*. PhD thesis, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, January 1994.

[165] R. G. Willson and S. Shafer. A perspective projection camera model for zoom lenses. In *Proc. Conference on Optical 3-D Measurement Techniques*, October 1993.

[166] R. G. Willson and S. Shafer. What is the center of the image? *Journal of Optical Society of America A*, 11(11):2946–2955, November 1994.

[167] R. J. Woodham. Photometric method for determining surface orientation from multiple images. *Optical Engineering*, 19(1):139–144, 1980.

[168] R. J. Woodham. Gradient and curvature from the photometric stereo method, including local confidence estimation. *Journal of Optical Society of America A*, 11(11):3050–3068, 1994.

[169] C. Yang, F. Sun, and Z. Hu. Planar conic based camera calibration. In *Proc. International Conference on Patter Recognition*, volume 1, pages 555–558, September 2000.

[170] R. Yang, M. Pollefeys, and G. Welch. Dealing with textureless regions and specular highlights-a progressive space carving scheme using a novel photo-consistency measure. In *Proc. IEEE International Conference on Computer Vision*, pages 576–584, 2003.

[171] A. Yao and A. Calway. Dense 3-D structure from image sequences using probabilistic depth carving. In *Proc. British Machine Vision Conference*, pages 211–220, September 2003.

[172] R. Zhang, P.-S. Tsai, J. E. Cryer, and M. Shah. Shape from shading: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(8):690–706, 1999.

[173] Z. Zhang. Parameter estimation techniques: A tutorial with application to conic fitting. Technical Report RR-2676, INRIA, 1995.

[174] Z. Zhang. Determining the epipolar geometry and its uncertainty: A review. *International Journal of Computer Vision*, 27(2):161–195, 1998.

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

[176] Z. Zhang. Camera calibration with one-dimensional objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(7):892–899, 2004.

[177] T. Zickler, P. N. Belhumeur, and D. J. Kriegman. Helmholtz stereopsis: Exploiting reciprocity for surface reconstruction. In *Proc. European Conference on Computer Vision*, volume III, pages 869–884, 2002.

[178] T. E. Zickler, P. N. Belhumeur, and D. J. Kriegman. Helmholtz stereopsis: Exploiting reciprocity for surface reconstruction. *International Journal of Computer Vision*, 49(2/3):215–227, 2002.

[179] T. E. Zickler, P. N. Belhumeur, and D. J. Kriegman. Toward a stratification of Helmholtz stereopsis. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 548–555, 2003.

[180] T. E. Zickler, J. Ho, D. J. Kriegman, J. Ponce, and P. N. Belhumeur. Binocular Helmholtz stereopsis. In *Proc. IEEE International Conference on Computer Vision*, pages 1411–1417, 2003.