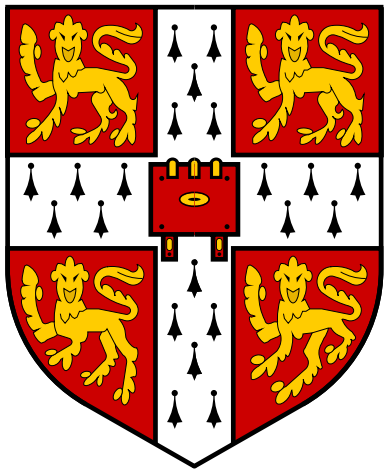
Structure and Motion from Silhouettes

by
Kwan-Yee Kenneth Wong
Wolfson College



Department of Engineering
University of Cambridge

A dissertation submitted to
the University of Cambridge
for the degree of
Doctor of Philosophy
Michaelmas Term 2001

Declaration

I hereby declare that no part of this thesis has already been or is being submitted for any other degree or qualification. This dissertation is the result of my own original work carried out in the Department of Engineering at the University of Cambridge, except where explicit reference has been made to the work of others. This dissertation contains 36,194 words and 91 figures.

我思，故我在

"*Cogito, ergo sum.*" (I think, therefore I am.)
- René Descartes, *Le Discours de la Méthode*.

Kwan-Yee Kenneth Wong
Wolfson College

Doctor of Philosophy
Michaelmas Term, 2001

Structure and Motion from Silhouettes

abstract

Silhouettes (or outlines) are often a dominant image feature, and can be extracted relatively easily and reliably. They provide rich information about both the shape and motion of an object, and are indeed the only information available in the case of smooth textureless surfaces. Nonetheless, due to the viewpoint dependence of silhouettes, they do not readily provide point correspondences, and hence structure and motion from silhouettes has been a challenging problem.

This dissertation first studies the static properties of silhouettes. By relating the idea of camera calibration from vanishing points to the symmetry property exhibited in the silhouettes of surfaces of revolution (SOR), a novel technique for estimating the intrinsic parameters of a camera from 2 or more silhouettes of SOR has been developed. Besides, a simple algorithm for recovering the 3D shape of a SOR using its silhouette from a single view is presented, followed by an analysis of the ambiguity in the reconstruction.

This dissertation then studies the dynamic properties of silhouettes, and introduces a complete and practical system for generating high quality 3D models from a sequence of 2D silhouettes. The input to the system is an image sequence of an object under both unknown circular motion and unknown general motion. By exploiting a simple parameterization of the fundamental matrix, circular motion can be estimated easily and accurately from the silhouettes. The registration of arbitrary general views, using silhouettes from the estimated circular motion, reveals information which is concealed under circular motion, and greatly improves both the shape and textures of the 3D models. In contrast to previous techniques, only the 2 outer epipolar tangents to the silhouettes are required in estimating both the circular and general motion, making the system practical in virtually all situations.

Acknowledgements

This thesis describes the results of my research carried out in the Speech, Vision and Robotics (SVR) Group of the Department of Engineering at the University of Cambridge. I would like to express my greatest gratitude towards my supervisor Prof. Roberto Cipolla for his constant support and encouragement. It was under his guidance that my research began to take root. I am also very grateful to my friend and colleague Dr. Paulo Mendonça for his invaluable advice and enthusiastic help. In particular, the theory for circular motion estimation, presented in Chapter 5, is the result of a close collaboration with him. I also want to thank him and his wife Cláudia Mendonça for their irreplaceable friendship.

I really enjoy working in the SVR Group and appreciate the company of all the group members. In particular, I would like to thank Dr. Tom Drummond, Paul Smith, Adrain Broadhurst, Anthony Dick, Duncan Robertson, Björn Stenger and Martin Weber for the discussions and all those enjoyable social events. During the course of my research, I have also benefited from the critical comments of many members of the vision research community outside the SVR Group. These include Prof. Olivier Faugeras, Prof. Richard Szeliski, Prof. Andrew Zisserman, Dr. Andrew Fitzgibbon, Dr. Geoff Cross, Dr. Yong-Duek Seo and Dr. Joan Lasenby. I would like to express my thankfulness towards them all. I also wish to thank Prof. H. S. Hung, Dr. Y. S. Moon, Dr. K. H. Wong and Dr. Tong Lee from The Chinese University of Hong Kong. Without their enthusiastic advice and support, I would not have started my PhD studies at the University of Cambridge.

This work was supported by the Cambridge Overseas Trust, and I would like to thank Schlumberger Cambridge Research for contributing towards my award from the Trust. I am also indebted to the Department of Engineering and Wolfson College for their financial support, which enabled me to take part in various conferences, workshops and seminars.

Last, but not least, I wish to thank my parents, my sisters Winnie and Jessica, my brother-in-law Jimmy Cheng, and my beloved one Wings Wong, for their everlasting love and support. This thesis is dedicated to them for all the untold sacrifices they have made.

Kwan-Yee Kenneth Wong

Contents

List of Figures

Notations

Points, Lines and Planes

$\tilde{\mathbf{x}}$	2D point in Cartesian coordinates $[x\ y]^{\mathrm{T}}$
$\tilde{\mathbf{X}}$	3D point in Cartesian coordinates $[X\ Y\ Z]^{\mathrm{T}}$
$\mathbf{x}$	2D point in homogeneous coordinates $[\alpha x\ \alpha y\ \alpha]^{\mathrm{T}}$
$\mathbf{X}$	3D point in homogeneous coordinates $[\alpha X\ \alpha Y\ \alpha Z\ \alpha]^{\mathrm{T}}$
$\mathbf{l}$	2D line $\mathbf{l}^{\mathrm{T}}\mathbf{x} = 0$
Π	3D plane $\Pi^{\mathrm{T}}\mathbf{X} = 0$

Camera Parameters

f	Focal length
a	Aspect ratio
ς	Skew
(u_0, v_0)	Principal point

Operators

$\dot{g}(s)$	Differentiation of function $g(s)$ with respect to s
$\mathbf{a} \cdot \mathbf{b}$	Scalar product of vectors $\mathbf{a}$ and $\mathbf{b}$
$\mathbf{a} \times \mathbf{b}$	Cross product of vectors $\mathbf{a}$ and $\mathbf{b}$
$[\mathbf{a}]_{\times}$	Anti-symmetric matrix of vector $\mathbf{a}$
$\det(\mathbf{M})$	Determinant of matrix $\mathbf{M}$
$\mathbf{M}^{-1}$	Inverse of matrix $\mathbf{M}$
$\mathbf{M}^{\mathrm{T}}$	Transpose of matrix $\mathbf{M}$
$\mathbf{M}^{+}$	Pseudo-inverse of matrix $\mathbf{M}$
$\mathbf{m}^{\perp}$	Null vector of matrix $\mathbf{M}$

Named Variables

$\mathbf{c}$	Optical center of a camera
$\mathbf{x}_0$	Principal point of a camera
$\mathbf{P}$	Projection matrix
$\mathbf{K}$	Calibration matrix
$\mathbf{R}$	Rotation matrix
$\mathbf{t}$	Translation vector
$\mathbf{E}$	Essential matrix
$\mathbf{F}$	Fundamental matrix
$\mathbf{e}$	Epipole
Γ	Contour generator of an object
ρ	Silhouette of an object
$\mathbf{p}$	Viewing vector
$\mathbf{n}$	Surface normal
$\mathbf{v}$	Vanishing point
$\mathbf{W}$	Harmonic homology
$\mathbf{T}$	Bilateral symmetry (about y-axis) transformation matrix
$\mathbf{S}$	Skew symmetry transformation matrix
$\mathbf{B}$	Bilateral symmetry transformation matrix
$\mathbf{\Omega}$	Absolute quadric
$\boldsymbol{\omega}$	Dual image of absolute conic

Matrices and Vectors

$\mathbb{I}_n$	Identity matrix of size $n \times n$
$\mathbb{O}_n$	Null vector of length n

Chapter 1

Introduction

"*The beginning is the most important part of the work.*"

- Plato, *The Republic*, Book II, 377B.

1.1 Motivation

The generation of realistic 3D models of real world objects is of great interest in many fields and has many practical applications. For instance, such 3D models can be used in model-based tracking system, video games, virtual reality, movie making and internet showroom. Traditionally, in computer graphics, such 3D models are constructed using specialized design softwares in a polygon by polygon fashion. Such approach is very time consuming and the quality of the output model depends very much on the skill of the user. The introduction of laser scan systems allows 3D objects to be "scanned" into the computer directly. In spite of that, such systems are very expensive and require careful calibration before use. Besides, they cannot cope with specular surfaces or surfaces with low reflectance, and can only handle objects of limited size. By allowing 3D models to be reconstructed automatically from image sequences, the *structure from motion*

techniques [132, 41, 81, 146, 36] in computer vision provide a cost-efficient solution to the above problem. In addition, vision-based systems can also handle objects with various size and reflectance.

1.1.1 Structure from Motion

In structure from motion (also known as *structure and motion*), image features are first extracted from the sequence by corner or edge detection techniques [16, 51] using the intensity gradient information. Such image features originate from scene structures like corners and edges, as well as from surface markings. Image features that correspond to the projections of the same scene structure are then matched, and this is referred to as the *correspondence problem* [89]. Initially, unguided matching is usually done by normalized cross-correlation of image intensities. By assuming the rigidity of the scene, the image motion is interpreted as completely arising from a rigid (relative) motion between the viewer and the scene. This motion can be computed from the matched image features (correspondences) by estimating the *epipolar geometry* [5, 40] which describes the geometry of stereo cameras (see Section 2.3.1, and also [5]). A guided matching, using the epipolar constraint (see Section 2.3.1), can then be performed to obtain more correspondences. The matching can also be further aided by using other geometric constraints like uniqueness, ordering, figural continuity and disparity gradient (see [40] for details). With a calibrated camera, Euclidean structure can then be obtained by triangulation [55] of the correspondences.

1.1.2 Smooth Textureless Surfaces

For smooth textureless surfaces, the dominant image feature is the *silhouette* (alternatively referred to as *apparent contour*, *occluding contour*, *profile* or *outline*). The silhouette is the projection of the locus of points on the surface at which the line of sight is orthogonal to the surface normal. In contrast to the features arising from corners, edges and surface markings, which are *viewpoint independent*, silhouettes are inherently *viewpoint dependent*. In general, 2 silhouettes of an arbitrary smooth object observed from 2 distinct viewpoints are the projections of 2 distinct curves in space, and hence they do not readily provide correspondences. As a result, the assumption of rigidity does not hold for silhouettes, and this calls for the development of a completely different set of techniques [111, 107, 45, 22, 4, 30, 64].

The major theme of this thesis is to develop a practical system for generating realistic 3D models of smooth objects. The static and dynamic properties of silhouettes are analyzed, and exploited to develop novel algorithms for solving the structure and motion problem. Such a model building system is particularly suitable for creating a digital archive of sculptures, which are often composed of smooth textureless surfaces (see figure 1.1).

1.2 Approach

This thesis aims at tackling the problem of structure and motion for smooth objects using silhouettes alone. It shows that silhouettes provide rich information which can be exploited for camera calibration, motion estimation and shape recovery. By refraining from the use of other image features like corners and textures, the

Figure 1.1: A miniature model of Michelangelo's David statue. Sculpture is often composed of smooth textureless surfaces for which the dominant image feature is the silhouette.

algorithms developed here are more general and can be applied to virtually all kinds of objects. The details of the approach employed here are listed below.

1.2.1 Imaging Model

Before information can be extracted from an image and be interpreted, the imaging model has to be defined. Due to its simplicity and expressiveness, the perspective or pin-hole camera model is commonly used as the imaging model in computer vision, and it is also the camera model adopted in this dissertation. The process of image formation by a pin-hole camera can be conveniently represented by a 3×4 projection matrix [112], which is composed of a camera calibration matrix and a rigid body transformation (see Section 2.2.1). In order to achieve Euclidean reconstruction, it is necessary to estimate the intrinsic parameters of the camera (i.e. the camera calibration matrix). In this dissertation, the symmetry property exhibited in the silhouettes of surfaces of revolution (SOR) is analyzed

and exploited for estimating these parameters.

1.2.2 Shape Recovery

2D images contain cues to surface shape and orientation, however their interpretations are ambiguous since depth information is lost during the image formation process. Nonetheless, if some strong a priori knowledge of the object is available, like a parametric description, then a single view alone allows shape recovery. In this dissertation, the surface geometry of surfaces of revolution is studied. Through the use of differential geometry and projective invariant [147, 77, 101, 149, 33], it is shown that the 3D shape of a surface of revolution can be recovered from its silhouette in a single view, up to an 1-parameter ambiguity.

An alternative approach for depth recovery is to introduce viewer motion. In this dissertation, the problem of motion estimation from silhouettes of an arbitrary object is tackled by first limiting the motion to be circular (e.g. turntable sequences) [44, 96]. By exploiting a simple parameterization of the fundamental matrix [81], expressed in terms of the fixed image features in the sequence, the circular motion can be estimated easily and accurately. The drawbacks of using circular motion alone for model reconstruction are then overcome by the registration of arbitrary general motion with the estimated circular motion. This divide-and-conquer approach avoids the common problems that exist in almost every algorithm for motion estimation from silhouettes, namely the need for a good but nontrivial initialization, the unrealistic demand for a large number of epipolar tangent points [111, 107, 22], and the presence of local minima.

1.2.3 Shape Representation

Depending on the nature of the surface and the image sequence, either a surface model or a volumetric model can be constructed from the set of silhouettes with known viewer motion. If a dense, continuous sequence is available, a surface model can be obtained by reconstructing the contour generators of a simple surface using differential techniques [24, 134, 12, 125]. On the other hand, if only sparse, discrete views are available and the object has relatively complex topologies, volume intersection techniques [108, 124] can be employed to produce a volumetric model which represents the visual hull [73, 74] of the object.

Due to its ability to describe object with more complex topologies, the volume intersection approach is chosen in this dissertation for model reconstruction from silhouettes. A simple technique for constructing an octree [63, 92] from the silhouettes is implemented. The octree representation allows the model to be constructed at different levels of resolution according to needs. Despite its modeling power, an octree is not very suitable for high speed rendering. For this reason, a triangulated mesh is extracted from the octree, and the resulting surface model can then be displayed efficiently with conventional graphics rendering algorithms (implemented either in hardware or software).

1.2.4 Theory and Practice

The ultimate goal of this thesis is to provide *practical* solutions for the problem of structure and motion from silhouettes. All the theories developed in this thesis have been implemented and tested against both synthetic and real data to demonstrate the feasibility of the algorithms. In particular, programs with user-friendly

interfaces, written in Microsoft Visual C++, have been developed to provide an easy-to-use system for producing high quality 3D models of objects from their silhouettes.

1.3 Contributions

Through the studies of the static and dynamic properties of silhouettes, computational theories have been developed in this thesis to provide practical solutions for the problem of structure and motion from silhouettes. The main contributions of this thesis include:

- a novel technique for camera calibration from silhouettes of surfaces of revolutions (Chapter 3). The method presented here allows the intrinsic parameters of a camera to be estimated from 2 or more silhouettes of surfaces of revolution (like bowls and vases etc.), which are commonly found in daily life. The use of such objects has the advantages of easy accessibility and low cost, in contrast to the traditional calibration patterns;

- a simple algorithm for reconstructing a surface of revolution from a single view (Chapter 4). The algorithm developed here allows a surface of revolution to be recovered from its silhouette in a single view, and produces an 1-parameter family of solutions. Analysis of the reconstruction ambiguity is also presented.

- the introduction of the use of *outer* epipolar tangents for motion estimation from silhouettes (Chapter 5). The outer epipolar tangents correspond to the 2 epipolar tangent planes that touch the object, and are always available

except when the baseline passes through the object. The use of the outer epipolar tangents, which are guaranteed to be in correspondence, avoids false matches due to self-occlusions and greatly simplifies the matching problem;

- a complete and practical system for generating high quality 3D models from 2D silhouettes (Chapter 5 and Chapter 6). The system introduced here produces a 3D model of an object from an image sequence of the object under both unknown circular motion and unknown general motion. In contrast to previous silhouette-based techniques, only the 2 outer epipolar tangents to the silhouettes are required for the motion estimation, making the system practical in virtually all situations.

1.4 Outline of the Thesis

Chapter 2. This chapter reviews some fundamental concepts in computer vision, which form the theoretical background for the analysis of silhouettes in the rest of this dissertation. It first reviews the pin-hole (perspective) camera model and presents the 3×4 projection matrix [112] that models the image formation process. It then gives a brief review of the epipolar geometry. Simple derivations for the essential matrix [78] and the fundamental matrix [81] are presented, followed by an analysis of the reconstruction ambiguity. Finally, it studies the differential geometry of a smooth object under perspective projection, and analyzes the epipolar geometry associated with its silhouettes.

Chapter 3. In this chapter a novel technique [141] for camera calibration from silhouettes of surfaces of revolution is introduced. It begins by giving a survey of the literature on camera calibration techniques. It then briefly reviews the theory of camera calibration from vanishing points. The symmetry property exhibited in the silhouettes of surfaces of revolution is then related to the idea of calibration from vanishing points, and a simple technique is developed for calibrating a camera from 2 or more silhouettes of surfaces of revolution. Experimental results on both synthetic and real data are presented, which demonstrate the accuracy and robustness of the algorithm.

Chapter 4. This chapter addresses the problem of reconstructing a surface of revolution from a single view. It first briefly reviews existing techniques for shape from contour using a single view. It then studies the surface geometry of surfaces of revolution, and shows that the surface normal at any point on a surface of revolution is coplanar with the axis of revolution. This coplanarity constraint is used to derive a simple depth equation for the silhouette under a special viewing condition. A simple algorithm is then introduced for rectifying the silhouette under general viewing condition so that it resembles the special viewing condition up to an 1-parameter ambiguity. The resulting ambiguity in the reconstruction is analyzed and experimental results on real data are presented.

Chapter 5. In this chapter, the problem of motion estimation from silhouettes is tackled. It starts by giving a literature review on motion estimation from silhouettes. It then introduces and justifies the use of outer epipolar tangents for motion estimation. A novel technique [96] for recovering the motion of an object

undergoing circular motion is presented, followed by a simple technique [138] for registering an arbitrary general view of the object with the circular motion. Convincing 3D models produced from experiments on various objects are presented, which demonstrate the accuracy and practicality of the system.

Chapter 6. This chapter studies the problem of model reconstruction from silhouettes. A survey of the literature on model reconstruction from silhouettes is first presented. It then briefly reviews the octree representation, and introduces an efficient algorithm for constructing an octree using silhouettes from multiple views. The implementation details for the silhouette extraction and intersection test are presented, followed by a description of an algorithm for extracting a triangulated mesh from the octree. Finally, experimental results on real data are presented, showing the quality of the reconstruction.

Chapter 7. This chapter presents a summary of the theories and algorithms developed in this dissertation, followed by a brief discussion of possible future work.

Chapter 2

Epipolar Geometry and Silhouettes: A Review

"Everything should be made as simple as possible, but no simpler."

- Albert Einstein.

2.1 Introduction

This chapter reviews some fundamental concepts in computer vision, which form the theoretical background for the analysis of silhouettes in the rest of this dissertation. In particular, the *epipolar geometry* [5, 40] plays an important role in both motion estimation and scene reconstruction. Due to the viewpoint dependency of the silhouettes, the epipolar geometry for viewing smooth objects demands special attentions.

Section 2.2 first reviews the pin-hole camera model, which is used in the derivation of the epipolar geometry in *stereo vision* [70, 6]. Section 2.3 gives a brief review of the epipolar geometry, which is summarized by the essential matrix [78] and the fundamental matrix [81]. A complete review on epipolar geometry can be found in [40, 146]. Section 2.4 studies the differential geometry of

a smooth object under perspective projection, and the epipolar geometry associated with the silhouettes. Further details on the differential geometry of smooth surfaces and silhouettes can be found in [24, 27].

2.2 Imaging Model

2.2.1 Pin-Hole Camera

In computer vision, a camera is commonly modeled as a pin-hole (perspective) camera and the imaging process can be expressed as

$$\alpha \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbf{P} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}, \tag{2.1}$$

where (X, Y, Z) is the coordinates of a 3D point $\mathbf{X}$, (u, v) is the image coordinates of the projection of $\mathbf{X}$, and α is an arbitrary scale factor. $\mathbf{P}$ is a 3×4 matrix known as the *projection matrix* [112] which models the pin-hole camera. The projection matrix $\mathbf{P}$ is not any general 3×4 matrix, but has a special structure given by [40]

$$\mathbf{P} = \mathbf{K}[\mathbf{R}\ \mathbf{t}], \tag{2.2}$$

where $\mathbf{K}$ is a 3×3 upper triangular matrix known as the *camera calibration matrix*, $\mathbf{R}$ is a 3×3 rotation matrix and $\mathbf{t}$ is a 3×1 translation vector. $\mathbf{R}$ and $\mathbf{t}$ are called the *extrinsic parameters* [40] of the camera, and they represent the rigid body transformation between the camera and the scene (see figure 2.1). The camera calibration matrix $\mathbf{K}$ has the form [40]

$$\mathbf{K} = \begin{bmatrix} af & \varsigma & u_0 \\ 0 & f & v_0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{2.3}$$

where f is the *focal length*, a is the *aspect ratio*, and ς is the *skew* which depends on the angle between the image axes. (u_0, v_0) is called the *principal point*, and it is the point at which the optical axis (z_c-axis) intersects the image plane (see figure 2.1). The focal length, aspect ratio, skew and principal point are referred to as the *intrinsic parameters* [40] of the camera, and *camera calibration* is the process of estimating these parameters. If the image axes are orthogonal to each other, which is often the case, ς will be equal to 0. In practice, the aspect ratio and skew of a camera are often assumed to be 1 and zero, respectively, to give more stable results in camera calibration. A camera is said to be calibrated if its intrinsic parameters are known. If both the intrinsic and extrinsic parameters of a camera are known, then the camera is said to be fully calibrated.

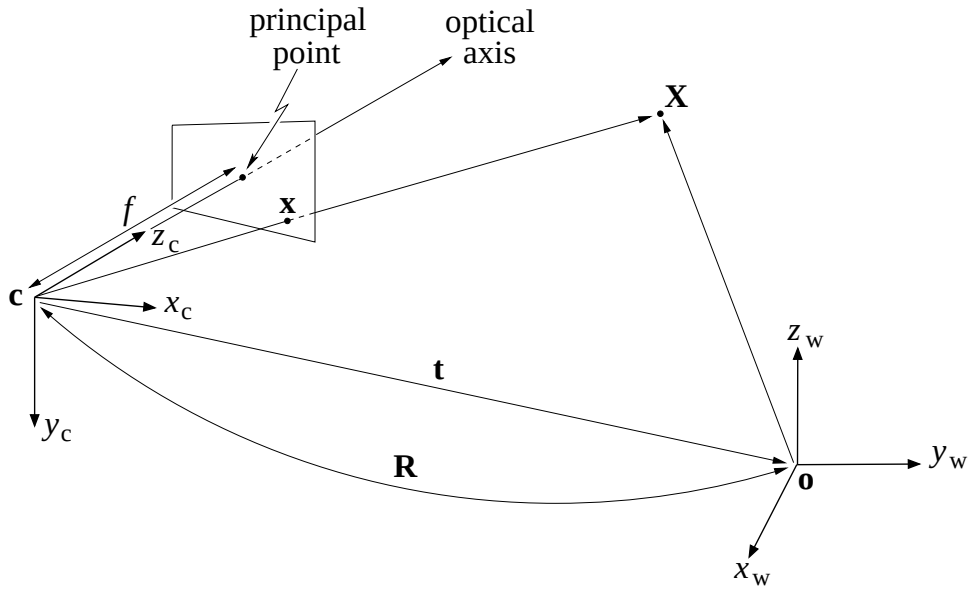


Figure 2.1: The extrinsic parameters of a camera represent the rigid body transformation between the world coordinate system (centered at **o**) and the camera coordinate system (centered at **c**), and the intrinsic parameters represent the camera internal parameters like focal length, aspect ratio, skew and principal point.

Given a point $\mathbf{x}$ in the image, the viewing vector from the camera center to the focal plane at unit distance for the point $\mathbf{x}$ is given by [40]

$$\hat{\mathbf{p}} = \mathbf{K}^{-1}\frac{\mathbf{x}}{x_3}, \tag{2.4}$$

in the camera coordinate system, and

$$\mathbf{p} = \mathbf{R}^{-1}\mathbf{K}^{-1}\frac{\mathbf{x}}{x_3}, \tag{2.5}$$

in the world coordinate system, respectively, where x_3 is the 3^{rd} coefficient of $\mathbf{x}$.

2.2.2 Vanishing Points and Horizon Lines

Under perspective projection, parallel lines in the world appear to meet at a single point in the image. This point is known as the *vanishing point* [40] corresponding to the direction of those parallel lines, and it is the image of a point at infinity at which those parallel lines "intersect". Vanishing points have been used to add realism to art since the 15^{th} century in Florence and during the period of Renaissance (see figure 2.2).

Similarly, parallel planes in the world appear to meet in a single line in the image. This line is known as the *horizon line* [40], and it is the image of a line at infinity along which those planes "intersect". Any set of parallel lines lying on those planes will have a vanishing point on the horizon line (see figure 2.3).

2.3 Stereo Vision

2.3.1 Epipolar Geometry

Figure 2.4 shows a pair of pin-hole cameras $\mathbf{P}_1$ and $\mathbf{P}_2$, with distinct centers $\mathbf{c}_1$ and $\mathbf{c}_2$ respectively. The line joining $\mathbf{c}_1$ and $\mathbf{c}_2$ is called the *baseline* [40] and

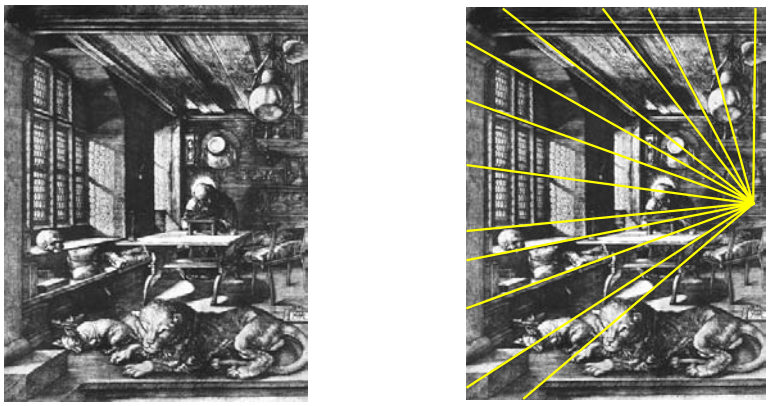


Figure 2.2: In his engraving “St. Jerome dans sa Cellule” produced in 1514, Albrecht Durer used perspective construction to give a sense of depth by making parallel lines in the ceiling and on the wall converge to a vanishing point.

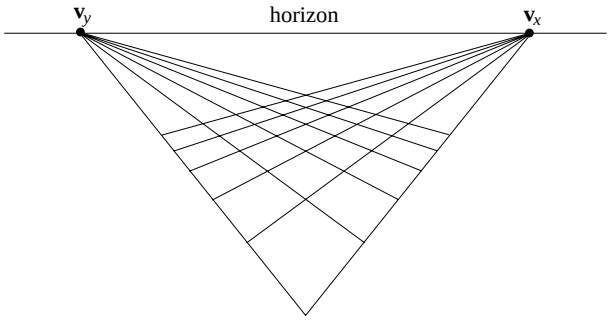


Figure 2.3: A horizon line is the image of a line at infinity along which parallel planes “intersect”. Any set of parallel lines lying on those planes will have a vanishing point on the horizon line.

the intersections of the baseline with the 2 image planes are known as the *epipoles* [40]. The epipole $\mathbf{e}_1$ is the image of $\mathbf{c}_2$ in $\mathbf{P}_1$. Similarly, the epipole $\mathbf{e}_2$ is the image of $\mathbf{c}_1$ in $\mathbf{P}_2$. The plane defined by $\mathbf{c}_1$, $\mathbf{c}_2$ and any arbitrary non-collinear 3D point $\mathbf{X}$ is known as the *epipolar plane* [40]. The intersections of the epipolar plane with the 2 image planes give 2 corresponding *epipolar lines* [40]. The epipolar line $\mathbf{l}_1$ in $\mathbf{P}_1$ is the image of the line through $\mathbf{c}_2$, $\mathbf{x}_2$ and $\mathbf{X}$. Similarly, the epipolar line $\mathbf{l}_2$ in $\mathbf{P}_2$ is the image of the line through $\mathbf{c}_1$, $\mathbf{x}_1$ and $\mathbf{X}$. It follows that the correspondence of a point on one image must lie on the corresponding epipolar line on the other image and vice versa (see figure 2.5), and this is known as the *epipolar constraint* [40]. Note that the set of epipolar planes forms a pencil of planes containing the baseline, and hence the epipolar lines on each image form a pencil of lines containing the corresponding epipole.

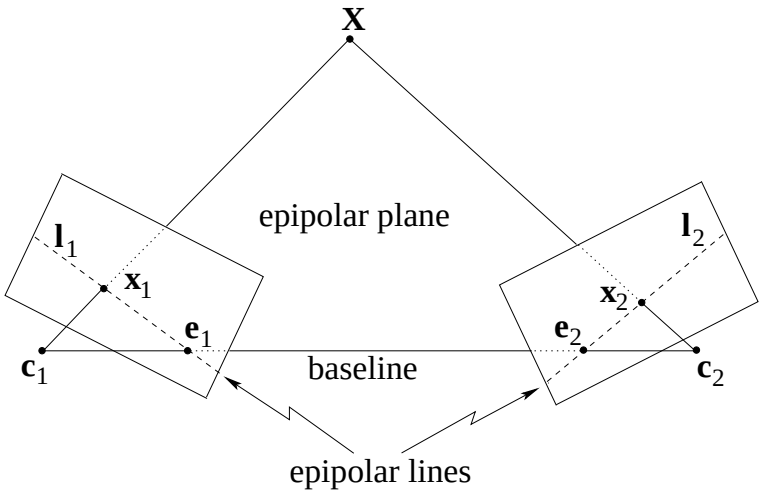


Figure 2.4: Epipolar geometry between 2 cameras.

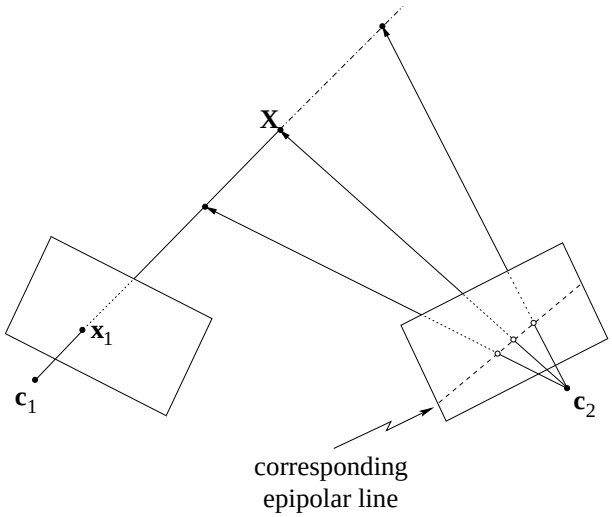


Figure 2.5: The epipolar constraint: given an image point $\mathbf{x}_1$ on one image, its correspondence on the other image must lie on the corresponding epipolar line which is the image of the line through $\mathbf{c}_1$, $\mathbf{x}_1$ and $\mathbf{X}$.

2.3.2 The Essential Matrix E

Consider 2 pin-hole cameras $\mathbf{P}_1$ and $\mathbf{P}_2$, with relative rotation $\mathbf{R}$ and translation $\mathbf{t} \neq \mathbb{O}_3$. Given a point $\tilde{\mathbf{X}}_1$ in the camera coordinate system of $\mathbf{P}_1$, its position $\tilde{\mathbf{X}}_2$ in the camera coordinate system of $\mathbf{P}_2$ is given by

$$\tilde{\mathbf{X}}_2 = \mathbf{R}\tilde{\mathbf{X}}_1 + \mathbf{t}. \tag{2.6}$$

Pre-multiplying both sides of (2.6) by $\tilde{\mathbf{X}}_2^{\mathrm{T}}[\mathbf{t}]_\times$ gives [78]

$$\begin{aligned} \tilde{\mathbf{X}}_2^{\mathrm{T}}[\mathbf{t}]_\times \mathbf{R}\tilde{\mathbf{X}}_1 &= 0 \\ \tilde{\mathbf{X}}_2^{\mathrm{T}}\mathbf{E}\tilde{\mathbf{X}}_1 &= 0 \end{aligned} \tag{2.7}$$

where $\mathbf{E}$ is a 3×3 matrix known as the *essential matrix* [78], given by

$$\mathbf{E} = [\mathbf{t}]_\times \mathbf{R}. \tag{2.8}$$

Equation (2.7) also holds for the viewing vectors $\hat{\mathbf{p}}_1$ and $\hat{\mathbf{p}}_2$ of the points $\mathbf{x}_1$ and $\mathbf{x}_2$, which are the projections of $\mathbf{X}_1$ and $\mathbf{X}_2$ in $\mathbf{P}_1$ and $\mathbf{P}_2$ respectively. This yields the epipolar constraint [78]

$$\hat{\mathbf{p}}_2^{\mathrm{T}}\mathbf{E}\hat{\mathbf{p}}_1 = 0. \tag{2.9}$$

The epipoles $\hat{\mathbf{e}}_1$ and $\hat{\mathbf{e}}_2$, in the camera coordinate systems of $\mathbf{P}_1$ and $\mathbf{P}_2$ respectively, are given by the right and left null spaces of $\mathbf{E}$. It follows from equation (2.8) that $\det(\mathbf{E}) = 0$ and $\mathbf{E}$ is therefore of maximum rank 2 [130, 41]. Note that $\mathbf{E}$ only depends on the relative rotation and translation between the 2 cameras and is defined only up to a scale factor, hence it has only 5 degrees of freedom.

2.3.3 The Fundamental Matrix F

Consider 2 pin-hole cameras $\mathbf{P}_1$ and $\mathbf{P}_2$ with distinct centers. Let $\mathbf{x}_1$ and $\mathbf{x}_2$ be the images of an arbitrary 3D point $\mathbf{X}$ in $\mathbf{P}_1$ and $\mathbf{P}_2$ respectively, i.e.

$$\mathbf{x}_1 = \mathbf{P}_1\mathbf{X}, \text{ and} \tag{2.10}$$

$$\mathbf{x}_2 = \mathbf{P}_2\mathbf{X}. \tag{2.11}$$

The image point $\mathbf{x}_1$ defines an optical ray on which $\mathbf{X}$ must lie. The equation of this optical ray is given by [146]

$$\mathbf{X}_{\text{ray}}(s) = \mathbf{p}_1^{\perp} + s\mathbf{P}_1^{+}\mathbf{x}_1, \tag{2.12}$$

where $\mathbf{P}_1^{+}$ is the pseudo-inverse of $\mathbf{P}_1$ and $\mathbf{p}_1^{\perp}$ is a null vector of $\mathbf{P}_1$. Note that $\mathbf{p}_1^{\perp}$ indicates the camera center of $\mathbf{P}_1$ and $\mathbf{P}_1^{+}\mathbf{x}_1$ gives the viewing direction. There exists s_0 such that $\mathbf{X} = \mathbf{X}_{\text{ray}}(s_0)$, and substituting $\mathbf{X}_{\text{ray}}(s_0)$ into (2.11) gives [146]

$$\begin{aligned} \mathbf{x}_2 &= \mathbf{P}_2(\mathbf{p}_1^{\perp} + s_0\mathbf{P}_1^{+}\mathbf{x}_1) \\ &= \mathbf{P}_2\mathbf{p}_1^{\perp} + s_0\mathbf{P}_2\mathbf{P}_1^{+}\mathbf{x}_1 \end{aligned} \tag{2.13}$$

Pre-multiplying both sides of (2.13) by $\mathbf{x}_2^{\mathrm{T}}[\mathbf{P}_2\mathbf{p}_1^{\perp}]_{\times}$ gives [146]

$$\begin{aligned} \mathbf{x}_2^{\mathrm{T}}[\mathbf{P}_2\mathbf{p}_1^{\perp}]_{\times}\mathbf{P}_2\mathbf{P}_1^{+}\mathbf{x}_1 &= 0 \\ \mathbf{x}_2^{\mathrm{T}}\mathbf{F}\mathbf{x}_1 &= 0, \end{aligned} \tag{2.14}$$

where $\mathbf{F}$ is a 3×3 matrix known as the *fundamental matrix* [81], given by

$$\mathbf{F} = [\mathbf{P}_2\mathbf{p}_1^{\perp}]_{\times}\mathbf{P}_2\mathbf{P}_1^{+}. \tag{2.15}$$

Equation (2.14) gives an expression of the epipolar constraint in homogeneous image coordinates, which does not require the knowledge of the intrinsic parameters of the 2 cameras. The epipoles $\mathbf{e}_1$ and $\mathbf{e}_2$, in homogeneous image coordinates, can be obtained from the right and left null spaces of $\mathbf{F}$ respectively, and are given by

$$\mathbf{e}_1 = (\mathbf{P}_2\mathbf{P}_1^{+})^{-1}(\mathbf{P}_2\mathbf{p}_1^{\perp}), \text{ and} \tag{2.16}$$

$$\mathbf{e}_2 = \mathbf{P}_2\mathbf{p}_1^{\perp}. \tag{2.17}$$

Since $\mathbf{F}$ is defined only up to a scale factor and $\det(\mathbf{F}) = 0$, it has only 7 degrees of freedom. By substituting (2.17) into (2.15), $\mathbf{F}$ can be rewritten in a *plane plus parallax representation* [81], given by

$$\mathbf{F} = [\mathbf{e}_2]_{\times}\mathbf{M}, \tag{2.18}$$

where $\mathbf{M} = \mathbf{P}_2\mathbf{P}_1^{+}$ is a *plane induced homography* [81]. Note that replacing $\mathbf{M}$ in (2.18) by any matrix

$$\mathbf{M}' = \mathbf{M} + \mathbf{e}_2\mathbf{a}^{\mathrm{T}}, \tag{2.19}$$

where $\mathbf{a}$ is any arbitrary 3-vector, will yield the same fundamental matrix [54]. This corresponds to choosing a different plane that induces the homography. The

homography $\mathbf{M}'$ will map epipolar lines to corresponding epipolar lines [52, 81] by

$$\mathbf{l}_2 = \mathbf{M}'^{-\mathrm{T}}\mathbf{l}_1, \text{ and} \tag{2.20}$$

$$\mathbf{l}_1 = \mathbf{M}'^{\mathrm{T}}\mathbf{l}_2, \tag{2.21}$$

where $\mathbf{l}_1$ and $\mathbf{l}_2$ are a pair of corresponding epipolar lines in $\mathbf{P}_1$ and $\mathbf{P}_2$ respectively.

2.3.4 Reconstruction Ambiguity

Both the essential matrix and the fundamental matrix encode information about the geometry of stereo cameras which is necessary for motion estimation. It is well-known that from image correspondences (or equivalently the fundamental matrix) alone, the projection matrices and the reconstruction of the scene points can only be determined up to an arbitrary *projective transformation* [39, 53]. Consider again equations (2.10) and (2.11):

$$\mathbf{x}_1 = \mathbf{P}_1\mathbf{X} = \mathbf{P}_1\mathbf{H}\mathbf{H}^{-1}\mathbf{X}, \text{ and} \tag{2.22}$$

$$\mathbf{x}_2 = \mathbf{P}_2\mathbf{X} = \mathbf{P}_2\mathbf{H}\mathbf{H}^{-1}\mathbf{X}, \tag{2.23}$$

where $\mathbf{H}$ is any arbitrary nonsingular 4×4 matrix representing a projective transformation. Equations (2.22) and (2.23) suggest that ($\mathbf{P}_1\mathbf{H}$, $\mathbf{P}_2\mathbf{H}$, $\mathbf{H}^{-1}\mathbf{X}$) is also a valid reconstruction from the image points resulted from ($\mathbf{P}_1$, $\mathbf{P}_2$, $\mathbf{X}$). This can be verified by substituting $\mathbf{P}_1$ and $\mathbf{P}_2$ in equation (2.15) by $\mathbf{P}_1\mathbf{H}$ and $\mathbf{P}_2\mathbf{H}$ respectively, and it will yield the same fundamental matrix.

The reconstruction ambiguity can be reduced by upgrading the fundamental matrix to an essential matrix. Let $\mathbf{P}_1 = \mathbf{K}_1[\mathbf{R}_1 \ \mathbf{t}_1]$ and $\mathbf{P}_2 = \mathbf{K}_2[\mathbf{R}_2 \ \mathbf{t}_2]$. Substi-

tuting $\mathbf{P}_1$ and $\mathbf{P}_2$ into (2.15) gives

$$\begin{aligned}\mathbf{F} &= \mathbf{K}_2^{-\mathrm{T}}[\mathbf{t}_2 - \mathbf{R}_2\mathbf{R}_1^{\mathrm{T}}\mathbf{t}_1]_\times \mathbf{R}_2\mathbf{R}_1^{\mathrm{T}}\mathbf{K}_1^{-1} \\ &= \mathbf{K}_2^{-\mathrm{T}}[\mathbf{t}]_\times \mathbf{R}\mathbf{K}_1^{-1}, \end{aligned} \tag{2.24}$$

where $\mathbf{R} = \mathbf{R}_2\mathbf{R}_1^{\mathrm{T}}$ and $\mathbf{t} = (\mathbf{t}_2 - \mathbf{R}_2\mathbf{R}_1^{\mathrm{T}}\mathbf{t}_1)$ are the relative rotation and translation between $\mathbf{P}_1$ and $\mathbf{P}_2$. Hence if the camera calibration matrices $\mathbf{K}_1$ and $\mathbf{K}_2$ are known, the associated fundamental matrix $\mathbf{F}$ can be upgraded to an essential matrix

$$\mathbf{E} = \mathbf{K}_2^{\mathrm{T}}\mathbf{F}\mathbf{K}_1, \tag{2.25}$$

which can then be decomposed to recover the relative rotation and translation between the cameras. Since the essential matrix is defined only up to a scale factor, only the direction of the relative translation can be recovered and this results in a reconstruction up to a *similarity transformation.*

2.4 Smooth Object and Its Projection

2.4.1 Contour Generators

Consider a smooth object and a static pin-hole camera. A set of rays which are tangent to the surface of the object can be cast from the camera center. These rays touch the object along a smooth curve known as the *contour generator* [87, 24] (see figure 2.6). In the literature, the contour generator is also known as the *extremal boundary* [7] or the *rim* [68]. The contour generator separates the visible part from the occluded part of the object, and can be parameterized by s as [24]

$$\tilde{\Gamma}(s) = \tilde{\mathbf{c}} + \lambda(s)\mathbf{p}(s), \text{ where} \tag{2.26}$$

$$\mathbf{p}(s) \cdot \mathbf{n}(s) = 0. \tag{2.27}$$

In equation (2.26), $\tilde{\mathbf{c}}$ indicates the camera center, $\mathbf{p}(s)$ is the viewing vector from $\tilde{\mathbf{c}}$ to the focal plane at unit distance for the point $\tilde{\Gamma}(s)$, and $\lambda(s)$ is the depth of the point $\tilde{\Gamma}(s)$ along the optical axis from $\tilde{\mathbf{c}}$. The tangency constraint is expressed in equation (2.27), where $\mathbf{n}(s)$ indicates the unit surface normal at $\tilde{\Gamma}(s)$. It follows from equations (2.26) and (2.27) that the contour generator depends on both local surface geometry and the viewpoint.

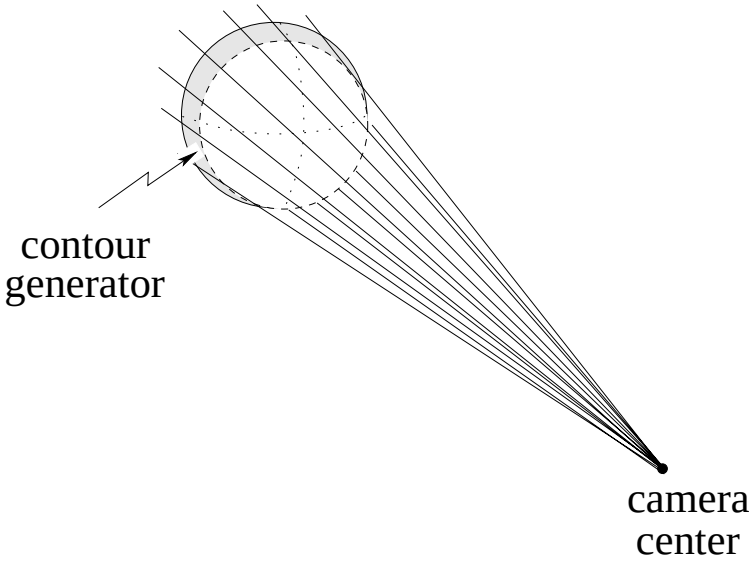


Figure 2.6: Optical rays which are tangent to the surface from the camera center touch the surface along a smooth curve known as the contour generator. The contour generator separates the visible part from the occluded part of the object.

In general, the viewing direction and the contour generator will not be orthogonal to each other, but are in conjugate directions with respect to the second fundamental form II [71, 68]. This means that the change in surface normal for an infinitesimal movement in the direction of the contour generator is orthogonal to the viewing direction. Consider the tangent to the contour generator at $\tilde{\Gamma}(s)$,

given by [24]

$$\frac{d\,\tilde{\Gamma}(s)}{d\,s} = \frac{d\,\lambda(s)}{d\,s}\mathbf{p}(s) + \lambda(s)\frac{d\,\mathbf{p}(s)}{d\,s}. \tag{2.28}$$

This tangent must lie on the tangent plane of the surface at $\tilde{\Gamma}(s)$, and hence it satisfies [24]

$$\frac{d\,\tilde{\Gamma}(s)}{d\,s} \cdot \mathbf{n}(s) \;\; = \;\; 0. \tag{2.29}$$

Taking the scalar product with $\mathbf{n}(s)$ from the right on both sides of (2.28), and substituting (2.27) and (2.29), gives [24]

$$\frac{d\,\mathbf{p}(s)}{d\,s} \cdot \mathbf{n}(s) = 0. \tag{2.30}$$

Differentiating (2.27) with respect to s and substituting (2.30) yields [24]

$$\mathbf{p}(s) \cdot \frac{d\,\mathbf{n}(s)}{d\,s} = 0, \tag{2.31}$$

which proves the conjugate direction relationship between the viewing ray and the contour generator.

2.4.2 Silhouettes

A contour generator is projected onto the image plane as an *apparent contour* (also known as a *profile*). A *silhouette* is a subset of the apparent contour where the viewing rays of the contour generator touch the object (i.e. not passing through the object). If the camera is (fully) calibrated, the viewing rays $\mathbf{p}(s)$ of the contour generator can be recovered from the silhouette (see Section 2.2.1). These rays define a *viewing cone* on which the contour generator lies, and within which the object is confined (see figure 2.7). However, the depth parameter $\lambda(s)$ in equation (2.26), and hence the contour generator itself, cannot be determined from a

single view alone. It follows from equation (2.30) that, like the viewing ray, the tangent to the silhouette also lies on the tangent plane of the surface at $\tilde{\Gamma}(s)$. This allows the unit surface normal at $\tilde{\Gamma}(s)$ to be determined up to a sign by [24]

$$\mathbf{n}(s) = \frac{\mathbf{p}(s) \times \frac{d\ \mathbf{p}(s)}{d\ s}}{\left|\mathbf{p}(s) \times \frac{d\ \mathbf{p}(s)}{d\ s}\right|}. \tag{2.32}$$

The sign of $\mathbf{n}(s)$ can be fixed if the side of the silhouette on which the surface lies is known (see figure 2.8).

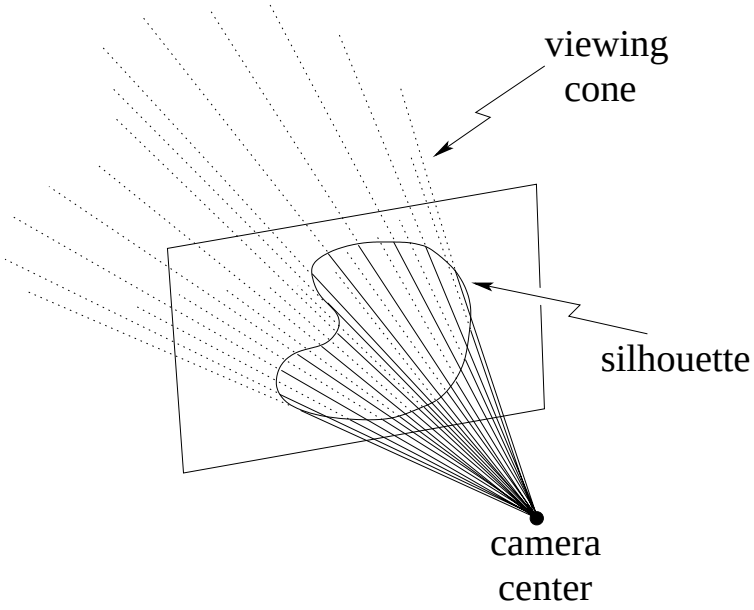


Figure 2.7: The viewing rays of the contour generator can be recovered from the silhouette and the camera center. These rays define a viewing cone on which the contour generator lies, and within which the object is confined

Due to the viewpoint dependency of the contour generators, silhouettes from 2 distinct viewpoints will be, in general, the projections of 2 different space curves (contour generators). As a result, the rigidity constraint no longer holds and there will be no correspondence between points in the 2 silhouettes. The only exception

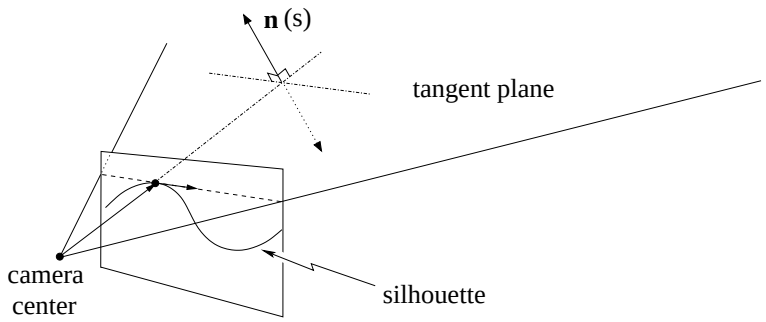


Figure 2.8: The unit surface normal can be determined from a single silhouette. The sign of the normal can be fixed if the side of the silhouette on which the surface lies is known.

is the *frontier point* [45, 22, 27] which is the intersection of the 2 contour generators in space and is visible in both views (see figure 2.9). Since the viewing rays of the frontier point from the 2 camera centers are both tangent to the surface, the frontier point lies on an epipolar plane which is tangent to the surface. It follows that a frontier point will be projected onto a point in the silhouette which is also an *epipolar tangent point* [111, 107, 22].

2.4.3 Epipolar Parameterization

Consider a smooth object and a moving pin-hole camera. As the camera moves, the contour generator slips over the visible surface of the object. As a result, the surface of the object can be parameterized by the spatial-temporal surface swept out by the contour generator due to camera motion. By introducing the time parameter t to equations (2.26) and (2.27), the parameterization of the surface is given by [24]

$$\tilde{\Gamma}(s,t) = \tilde{\mathbf{c}}(t) + \lambda(s,t)\mathbf{p}(s,t), \text{ where} \tag{2.33}$$

$$\mathbf{p}(s,t) \cdot \mathbf{n}(s,t) = 0. \tag{2.34}$$

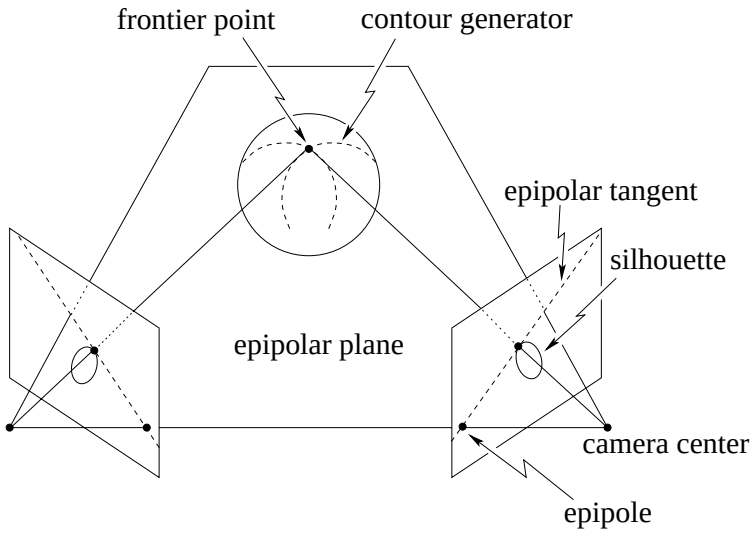


Figure 2.9: A frontier point is the intersection of 2 contour generators and lies on an epipolar plane which is tangent to the surface. It follows that a frontier point will be projected onto a point in the silhouette which is also an epipolar tangent point.

Such a parameterization is, however, under-constrained: the s-parameter curve $\tilde{\Gamma}(s, t_0)$ with constant t is the contour generator from the camera center $\tilde{\mathbf{c}}(t_0)$, whereas the t-parameter curve $\tilde{\Gamma}(s_0, t)$ with constant s has no physical interpretation. The most widely used parameterization is the epipolar parameterization [23] (see figure 2.10) which is derived from the epipolar geometry in stereo vision. The epipolar parameterization is defined by

$$\frac{\partial \tilde{\Gamma}(s,t)}{\partial t} \times \mathbf{p}(s,t) = \mathbb{O}_3 \,. \tag{2.35}$$

Equation (2.35) implies that the tangent to the t-parameter curve is chosen to be in the direction of the viewing ray. The physical interpretation is that points on the contour generator are chosen to move along the viewing rays, in an infinitesimal sense, as the camera moves. Since the viewing ray and the contour generator are in conjugate directions (see Section 2.4.1), so are the tangent plane basis vectors

$\frac{\partial\tilde{\Gamma}(s,t)}{\partial t}$ and $\frac{\partial\tilde{\Gamma}(s,t)}{\partial s}$ of the parameterized surface. Note that the epipolar parameterization is degenerate at frontier pointers where $\frac{\partial\tilde{\Gamma}(s,t)}{\partial t} = \mathbb{O}_3$ [24, 47, 26].

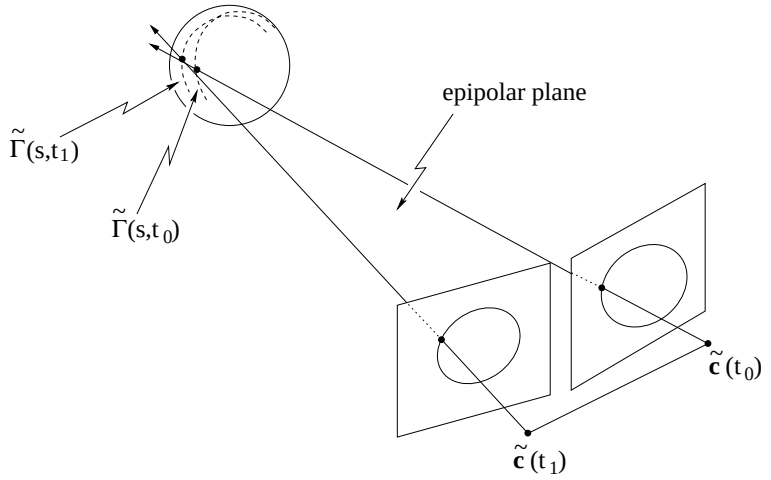


Figure 2.10: Epipolar parameterization for the spatial-temporal surface swept by the contour generator.

2.5 Summary

In this chapter, the pin-hole camera model, the epipolar geometry and the projection of smooth objects have been reviewed. These are essential to the development of the theories and algorithms presented in the rest of this dissertation.

The process of image formation by a pin-hole camera can be represented by a 3×4 projection matrix. The projection matrix can be decomposed into the intrinsic and extrinsic parameters, which represent the camera internal parameters and the rigid body transformation between the camera and the scene respectively. The estimation of the intrinsic parameters (i.e. camera calibration) from surfaces of revolution will be studied in Chapter 3, whereas the estimation of the extrinsic parameters (i.e. motion estimation) from silhouettes will be addressed in Chap-

ter 5.

The epipolar constraint in stereo vision is encoded by the essential matrix in the case of calibrated cameras, or by the fundamental matrix in the case of uncalibrated cameras. The estimation of the fundamental matrix from point correspondences forms the basis of virtually every motion estimation algorithm. The reconstruction ambiguity, arise from using point correspondences alone, can be reduced from a projective transformation to a similarity transformation by upgrading the fundamental matrix to an essential matrix using the calibration matrices. In Chapter 4 and Chapter 5, the calibration matrices of the cameras are assumed to be known from off-line calibration, and hence scaled Euclidean reconstruction can be achieved.

The contour generator of a smooth object depends on both local surface geometry and viewpoint, and so is its projection (silhouette) on the image plane. In the case of a (fully) calibrated camera, the surface normal along the contour generator can be determined from the silhouette using the tangency constraint. This surface normal information is utilized in Chapter 4 for reconstructing a surface of revolution from a single view. In general, due to the viewpoint dependency of the contour generators, the epipolar constraint cannot be applied to the points in the 2 silhouettes observed from 2 distinct viewpoints. The intersections between 2 contour generators result in frontier points, which are visible in both views and satisfy the epipolar constraint. The point correspondences induced by the frontier points are exploited in Chapter 5 to develop a practical algorithm for motion estimation from silhouettes.

Chapter 3

Camera Calibration from Symmetry

"...it is therefore useful, because it is symmetrical and fair."

- Ralph Waldo Emerson, *Art*, First Series.

3.1 Introduction

An essential step for motion estimation and 3D Euclidean reconstruction, 2 important tasks in computer vision, is the determination of the intrinsic parameters of cameras. This process, known as *camera calibration*, usually involves taking images of some special patterns with known geometry, extracting the features in the images, and minimizing their reprojection errors. Details of such calibration algorithms can be found in [42, 129, 75] and [40, Chapter 3]. These methods do not require direct mechanical measurements on the cameras, and often produce very good results. Nevertheless, they involve the design and use of highly accurate tailor-made calibration patterns, which are often both difficult and expensive to be manufactured.

In this chapter a novel technique for camera calibration is introduced. It relates the idea of calibration from vanishing points [17, 25, 76] to the symmetry prop-

erty exhibited in the silhouettes of surfaces of revolution [147, 77, 101, 149, 33]. The method presented here allows the camera to be calibrated from 2 or more silhouettes of surfaces of revolution (like bowls and vases etc.), which are commonly found in daily life. The use of such objects has the advantages of easy accessibility and low cost, in contrast to the traditional calibration patterns.

A survey of the literature on camera calibration is given in Section 3.2, followed by a brief review of camera calibration from vanishing points in Section 3.3. The symmetry property associated with the silhouettes of surfaces of revolution is reviewed in Section 3.4, and Section 3.5 shows how such a symmetry property can be related to the vanishing points associated with a set of 3 mutually orthogonal directions. By extending the techniques for calibration from vanishing points, the symmetry property can be used in the development of a practical algorithm for camera calibration [141]. Such an algorithm, detailed in Section 3.6, is capable of dealing with both known and unknown aspect ratio. The degenerate cases in which the algorithm fails are discussed in Section 3.7. Section 3.8 first presents results of experiments conducted on synthetic data, which are used to perform an evaluation on the robustness of the algorithm in the presence of noise. Experiments on real data then show the usefulness of the proposed method. Finally, discussions are presented in Section 3.9.

3.2 Previous Works

Classical calibration techniques [15, 119, 38] in photogrammetry involve full-scale nonlinear optimizations with large number of parameters. Despite being able to adopt accurate complex camera models, these techniques require a good

initialization and are computationally expensive. In [1], Abdel-Aziz and Karara presented the *direct linear transformation* (DLT), which is one of the most commonly used calibration techniques in computer vision. By ignoring lens distortion and treating the coefficients of the 3×4 projection matrix as unknowns, DLT only involves solving a system of linear equations which can be done by a linear least-squares method. In practice, the linear solution obtained from DLT is usually refined iteratively by minimizing the reprojection errors of the 3D reference points [42, 40]. In [129, 75], Tsai and Lenz introduced the *radial alignment constraint* (RAS) and developed a technique which also accounts for lens distortion.

All the calibration techniques mentioned so far require the knowledge of the 3D coordinates of a certain number of reference points and their corresponding image coordinates. In [17], Caprile and Torre showed that, under the assumption of zero skew and aspect ratio 1, it is possible to calibrate a camera from the vanishing points associated with 3 mutually orthogonal directions. This idea was further elaborated in [25, 76] to develop practical systems for reconstructing architectural scenes. In contrast to traditional calibration techniques, these methods depend only on the presence of some special structures, but not on the exact geometry of those structures.

The theory of *self-calibration* was first introduced by Maybank and Faugeras [91], who established the relationship between camera calibration and the epipolar transformation via the *absolute conic* [40]. Implementation of the theory in [91], together with real data experiments, were given by Luong and Faugeras [82] for fixed intrinsic parameters. In [128], Triggs introduced the *absolute quadric* and gave a simpler formulation which can incorporate any constraint on the intrinsic parameters easily. Based on [128], a practical technique for self-calibration of

multiple cameras with varying intrinsic parameters was developed by Pollefeys et al. in [104]. Other approaches to self-calibration also include restricting the camera motion to pure rotation [35] or planar motion [3].

The calibration technique introduced in this chapter, namely *calibration from surfaces of revolution*, falls into the same category as calibration from vanishing points (see figure 3.1). Like calibration from vanishing points, which only requires the presence of 3 mutually orthogonal directions, the technique presented here only requires the calibration target to be a surface of revolution, but the exact geometry of the surface is not important. A linear solution can be obtained in the case of zero skew and known aspect ratio, which can be further refined by a nonlinear optimization that is also capable of recovering unknown aspect ratio.

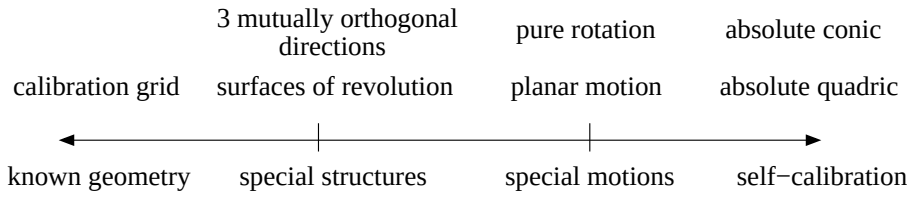


Figure 3.1: Different categories of camera calibration techniques.

3.3 Calibration from Vanishing Points

In [17], Caprile and Torre showed that under the assumption of zero skew and aspect ratio 1, the principal point of a camera will coincide with the orthocenter of a triangle with vertices given at 3 vanishing points from 3 mutually orthogonal directions. This property of the vanishing points, together with the symmetry property associated with the silhouettes of surfaces of revolution, will be used later in Section 3.5 to derive a simple technique for camera calibration. A simple

derivation of Caprile and Torre's result is given below.

Consider a pin-hole camera with focal length f, aspect ratio 1, zero skew and principal point $\tilde{\mathbf{x}}_0$. The vector from the camera center to any point $\tilde{\mathbf{x}}$ on the image plane, in camera coordinate system, is given by $[(\tilde{\mathbf{x}} - \tilde{\mathbf{x}}_0)^{\mathrm{T}} \ f]^{\mathrm{T}}$. Let $\tilde{\mathbf{v}}_q$, $\tilde{\mathbf{v}}_r$ and $\tilde{\mathbf{v}}_s$ be 3 vanishing points associated with 3 mutually orthogonal directions $\mathbf{N}_q$, $\mathbf{N}_r$ and $\mathbf{N}_s$ respectively. The 3 vectors from the camera center to $\tilde{\mathbf{v}}_q$, $\tilde{\mathbf{v}}_r$ and $\tilde{\mathbf{v}}_s$ will be mutually orthogonal to each other, and hence

$$(\tilde{\mathbf{v}}_q - \tilde{\mathbf{x}}_0) \cdot (\tilde{\mathbf{v}}_r - \tilde{\mathbf{x}}_0) + f^2 = 0, \tag{3.1}$$

$$(\tilde{\mathbf{v}}_r - \tilde{\mathbf{x}}_0) \cdot (\tilde{\mathbf{v}}_s - \tilde{\mathbf{x}}_0) + f^2 = 0, \tag{3.2}$$

$$(\tilde{\mathbf{v}}_s - \tilde{\mathbf{x}}_0) \cdot (\tilde{\mathbf{v}}_q - \tilde{\mathbf{x}}_0) + f^2 = 0. \tag{3.3}$$

Subtracting (3.3) from (3.1) gives

$$(\tilde{\mathbf{v}}_q - \tilde{\mathbf{x}}_0) \cdot (\tilde{\mathbf{v}}_r - \tilde{\mathbf{v}}_s) = 0. \tag{3.4}$$

Equation (3.4) shows that $\tilde{\mathbf{x}}_0$ lies on a line passing through $\tilde{\mathbf{v}}_q$ and orthogonal to the line joining $\tilde{\mathbf{v}}_r$ and $\tilde{\mathbf{v}}_s$. Similarly, subtracting (3.1) from (3.2) and (3.2) from (3.3) gives

$$(\tilde{\mathbf{v}}_r - \tilde{\mathbf{x}}_0) \cdot (\tilde{\mathbf{v}}_s - \tilde{\mathbf{v}}_q) = 0, \tag{3.5}$$

$$(\tilde{\mathbf{v}}_s - \tilde{\mathbf{x}}_0) \cdot (\tilde{\mathbf{v}}_q - \tilde{\mathbf{v}}_r) = 0. \tag{3.6}$$

Equations (3.4)–(3.6) imply that the principal point $\tilde{\mathbf{x}}_0$ coincides with the orthocenter of the triangle with vertices $\tilde{\mathbf{v}}_q$, $\tilde{\mathbf{v}}_r$ and $\tilde{\mathbf{v}}_s$. Besides, equations (3.1)–(3.3) show that the focal length f is equal to the square root of the product of the distances from the orthocenter to any vertex and to the opposite side (see figure 3.2). As a result, under the assumption of zero skew and aspect ratio 1, it is possible

to estimate the principal point and the focal length of a camera using vanishing points from 3 mutually orthogonal directions. A similar derivation was also presented by Cipolla et al. in [25].

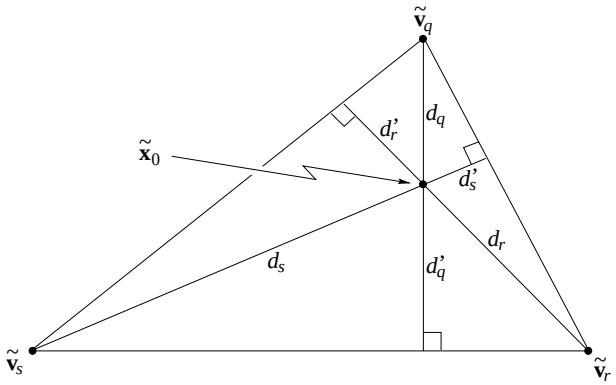


Figure 3.2: The principal point $\tilde{\mathbf{x}}_0$ of the camera coincides with the orthocenter of the triangle with vertices given at the vanishing points $\tilde{\mathbf{v}}_q$, $\tilde{\mathbf{v}}_r$ and $\tilde{\mathbf{v}}_s$ associated with 3 mutually orthogonal directions, and the focal length of the camera is given by $f = \sqrt{d_q d_q'} = \sqrt{d_r d_r'} = \sqrt{d_s d_s'}$.

3.4 Symmetry in Surfaces of Revolution

The silhouette of a surface of revolution, viewed under a pin-hole camera, will be invariant to a harmonic homology [149]. This property of the silhouette can be exploited to calibrate the intrinsic parameters of a camera, as will be shown in Section 3.5. A simple proof of this symmetry property is given below, which also shows that the axis of the associated harmonic homology is given by the image of the revolution axis, and that the center of the homology is given by the vanishing point corresponding to the normal direction of the plane containing the axis of revolution and the camera center.

Consider a surface of revolution $\mathbf{S}_\mathrm{r}$ whose axis of revolution coincides with

the y-axis, being viewed by a pin-hole camera $\hat{\mathbf{P}} = [\mathbb{I}_3 \ \mathbf{t}]$ where $\mathbf{t} = [0 \ 0 \ d_z]^{\mathrm{T}}$ with $d_z > 0$. By symmetry considerations, it is easy to see that the silhouette $\hat{\rho}$ of $\mathbf{S}_{\mathrm{r}}$ formed on the image plane will be bilaterally symmetric about the image of the revolution axis $\hat{\mathbf{l}}_{\mathrm{s}} = [1 \ 0 \ 0]^{\mathrm{T}}$. A simple proof of this is given in Appendix B. The lines of symmetry (i.e. lines joining symmetric points on $\hat{\rho}$) will be parallel to the normal $\mathbf{N}_x = [1 \ 0 \ 0 \ 0]^{\mathrm{T}}$ of the plane Π_{s} that contains the axis of revolution and the camera center, and the vanishing point associated with $\mathbf{N}_x$ is given by $\hat{\mathbf{v}}_x = [1 \ 0 \ 0]^{\mathrm{T}}$. The bilateral symmetry exhibited in $\hat{\rho}$ can be described by the transformation [94, 96]

$$\begin{aligned} \mathbf{T} &= \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ &= \mathbb{I}_3 - 2\frac{\hat{\mathbf{v}}_x \hat{\mathbf{l}}_{\mathrm{s}}^{\mathrm{T}}}{\hat{\mathbf{v}}_x^{\mathrm{T}} \hat{\mathbf{l}}_{\mathrm{s}}}. \end{aligned} \tag{3.7}$$

Note that the transformation $\mathbf{T}$ is a *harmonic homology* (see Appendix A, and also [117, 29] for details) with axis $\hat{\mathbf{l}}_{\mathrm{s}}$ and center $\hat{\mathbf{v}}_x$, which maps every point in $\hat{\rho}$ to its symmetric counterpart in $\hat{\rho}$. The silhouette $\hat{\rho}$ is thus said to be invariant to the harmonic homology $\mathbf{T}$ (i.e. $\hat{\rho} = \mathbf{T}\hat{\rho}$).

Now consider an arbitrary pin-hole camera $\mathbf{P}$ by introducing the intrinsic parameters represented by the camera calibration matrix $\mathbf{K}$ to $\hat{\mathbf{P}}$, and by applying the rotation $\mathbf{R}$ to $\hat{\mathbf{P}}$ about its optical center. Hence $\mathbf{P} = \mathbf{K}\mathbf{R}[\mathbb{I}_3 \ \mathbf{t}]$ or $\mathbf{P} = \mathbf{H}\hat{\mathbf{P}}$, where $\mathbf{H} = \mathbf{K}\mathbf{R}$. Let $\mathbf{x}$ be the projection of a 3D point $\mathbf{X}$ in $\mathbf{P}$, hence

$$\begin{aligned} \mathbf{x} &= \mathbf{P}\mathbf{X} \\ &= \mathbf{H}\hat{\mathbf{P}}\mathbf{X} \\ &= \mathbf{H}\hat{\mathbf{x}}, \end{aligned} \tag{3.8}$$

where $\hat{\mathbf{x}} = \hat{\mathbf{P}}\mathbf{X}$. Equation (3.8) implies that the 3×3 matrix $\mathbf{H}$ represents a planar homography which transforms the image formed by $\hat{\mathbf{P}}$ into the image formed by $\mathbf{P}$. Similarly, $\mathbf{H}^{-1}$ transforms the image formed by $\mathbf{P}$ into the image formed by $\hat{\mathbf{P}}$. The silhouette ρ of $\mathbf{S}_\mathrm{r}$, formed on the image plane of $\mathbf{P}$, can thus be obtained by applying the planar homography $\mathbf{H}$ to $\hat{\rho}$ (i.e. $\rho = \mathbf{H}\hat{\rho}$). Let $\hat{\mathbf{x}}$ and $\hat{\mathbf{x}}'$ be a pair of symmetric points in $\hat{\rho}$, and $\mathbf{x} = \mathbf{H}\hat{\mathbf{x}}$ and $\mathbf{x}' = \mathbf{H}\hat{\mathbf{x}}'$ be their correspondences in ρ. The symmetry between $\hat{\mathbf{x}}$ and $\hat{\mathbf{x}}'$ is given by

$$\hat{\mathbf{x}}' \;=\; \mathbf{T}\hat{\mathbf{x}}. \tag{3.9}$$

Substituting $\hat{\mathbf{x}}$ and $\hat{\mathbf{x}}'$ in (3.9) by $\mathbf{H}^{-1}\mathbf{x}$ and $\mathbf{H}^{-1}\mathbf{x}'$, respectively, gives [94, 96]

$$\begin{aligned}
(\mathbf{H}^{-1}\mathbf{x}') &= \mathbf{T}(\mathbf{H}^{-1}\mathbf{x}) \\
\mathbf{x}' &= \mathbf{H}\mathbf{T}\mathbf{H}^{-1}\mathbf{x} \\
&= \mathbf{H}(\mathbb{I}_3 - 2\frac{\hat{\mathbf{v}}_x\hat{\mathbf{l}}_\mathrm{s}^\mathrm{T}}{\hat{\mathbf{v}}_x^\mathrm{T}\hat{\mathbf{l}}_\mathrm{s}})\mathbf{H}^{-1}\mathbf{x} \\
&= (\mathbb{I}_3 - 2\frac{\mathbf{v}_x\mathbf{l}_\mathrm{s}^\mathrm{T}}{\mathbf{v}_x^\mathrm{T}\mathbf{l}_\mathrm{s}})\mathbf{x},
\end{aligned} \tag{3.10}$$

where $\mathbf{v}_x = \mathbf{H}\hat{\mathbf{v}}_x$, and $\mathbf{l}_\mathrm{s} = \mathbf{H}^{-\mathrm{T}}\hat{\mathbf{l}}_\mathrm{s}$. Note that $\mathbf{v}_x$ is the vanishing point corresponding to the normal direction $\mathbf{N}_x$ in $\mathbf{P}$, and $\mathbf{l}_\mathrm{s}$ is the image of the revolution axis of $\mathbf{S}_\mathrm{r}$ in $\mathbf{P}$. Let $\mathbf{W} = \mathbf{H}\mathbf{T}\mathbf{H}^{-1}$ be the harmonic homology with axis $\mathbf{l}_\mathrm{s}$ and center $\mathbf{v}_x$. Equation (3.10) shows that $\mathbf{W}$ will map each point in ρ to its symmetric counterpart in ρ, and hence ρ is invariant to the harmonic homology $\mathbf{W}$ (i.e. $\rho = \mathbf{W}\rho$).

In general, the harmonic homology $\mathbf{W}$ has 4 degrees of freedom. When the camera is pointing directly towards the axis of revolution, the harmonic homology will reduce to a *skew symmetry* [66, 99, 19, 115], where the vanishing point $\mathbf{v}_x$ is

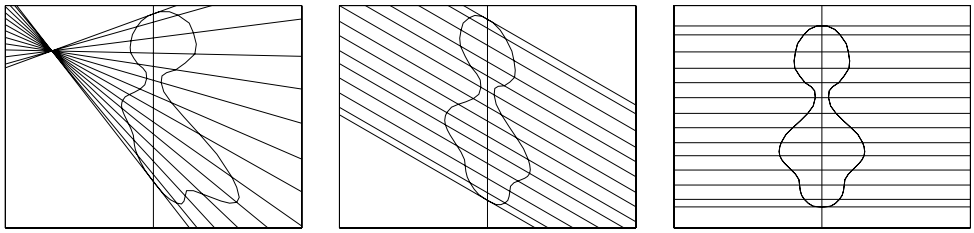
at infinity. The skew symmetry can be described by the transformation

$$\mathbf{S} = \frac{1}{\cos(\phi-\theta)} \begin{bmatrix} -\cos(\phi+\theta) & -2\cos\phi\sin\theta & 2d_{\mathrm{l}}\cos\phi \\ -2\sin\phi\cos\theta & \cos(\phi+\theta) & 2d_{\mathrm{l}}\sin\phi \\ 0 & 0 & \cos(\phi-\theta) \end{bmatrix}, \qquad (3.11)$$

where $d_{\mathrm{l}} = u_0\cos\theta + v_0\sin\theta$. The image of the revolution axis and the vanishing point are given by $\mathbf{l}_{\mathrm{s}} = [\cos\theta \ \ \sin\theta \ \ -d_{\mathrm{l}}]^{\mathrm{T}}$ and $\mathbf{v}_x = [\cos\phi \ \ \sin\phi \ \ 0]^{\mathrm{T}}$ respectively, and $\mathbf{S}$ has only 3 degrees of freedom. If the camera also has zero skew and aspect ratio 1, the transformation will then become a *bilateral symmetry*, given by

$$\mathbf{B} = \begin{bmatrix} -\cos 2\theta & -\sin 2\theta & 2d_{\mathrm{l}}\cos\theta \\ -\sin 2\theta & \cos 2\theta & 2d_{\mathrm{l}}\sin\theta \\ 0 & 0 & 1 \end{bmatrix}. \qquad (3.12)$$

While $\mathbf{l}_{\mathrm{s}}$ will have the same form as in the case of skew symmetry, the vanishing point will now be both at infinity and has a direction orthogonal to $\mathbf{l}_{\mathrm{s}}$. As a result, $\mathbf{B}$ has only 2 degrees of freedom. These 3 different cases of symmetry are illustrated in figure 3.3.



(a) (b) (c)

Figure 3.3: (a) Silhouette of a surface of revolution under general viewing conditions. The symmetry of the silhouette is described by a harmonic homology defined by the image of the revolution axis and a vanishing point. (b) When the camera is pointing directly towards the axis of revolution, the transformation reduces to a skew symmetry, which is a particular case of the harmonic homology where the vanishing point is at infinity. (c) If the camera also has zero skew and aspect ratio 1, the transformation becomes a bilateral symmetry, in which the vanishing point is at infinity and has a direction orthogonal to the image of the revolution axis.

3.5 Camera Calibration

Consider a surface of revolution $\mathbf{S}_\mathrm{r}$ viewed by a pin-hole camera $\mathbf{P} = \mathbf{K}[\mathbf{R}\ \mathbf{t}]$. Let ρ be the silhouette of $\mathbf{S}_\mathrm{r}$, $\mathbf{l}_\mathrm{s}$ be the image of the revolution axis of $\mathbf{S}_\mathrm{r}$, and $\mathbf{v}_x$ be the vanishing point corresponding to the normal direction $\mathbf{N}_x$ of the plane Π_s that contains the revolution axis of $\mathbf{S}_\mathrm{r}$ and the camera center of $\mathbf{P}$. The silhouette ρ is then invariant to the harmonic homology $\mathbf{W}$ with axis $\mathbf{l}_\mathrm{s}$ and center $\mathbf{v}_\mathrm{x}$ (see Section 3.4).

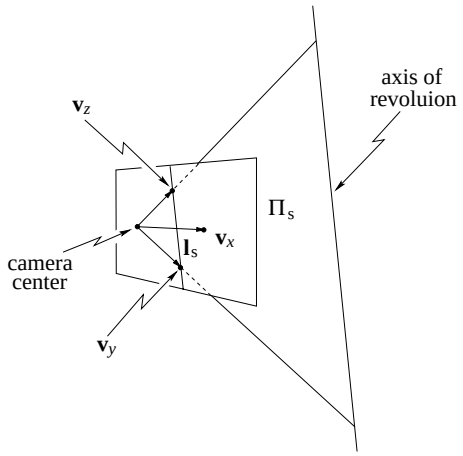


Figure 3.4: Three mutually orthogonal directions associated with a surface of revolution.

Consider now any 2 vectors $\mathbf{N}_y$ and $\mathbf{N}_z$ parallel to the plane Π_s and orthogonal to each other, which together with $\mathbf{N}_x$ form a set of 3 mutually orthogonal directions (see figure 3.4). Under the assumption of zero skew and aspect ratio 1, the vanishing points associated with these 3 directions can be used to determine the principal point and the focal length of $\mathbf{P}$, as shown in Section 3.3. By construction, the vanishing points $\mathbf{v}_y$ and $\mathbf{v}_z$, corresponding to the directions $\mathbf{N}_y$ and $\mathbf{N}_z$ respectively, will lie on the image of the revolution axis $\mathbf{l}_\mathrm{s}$. Given the

harmonic homology $\mathbf{W}$ associated with ρ, with an axis given by the image of the revolution axis $\mathbf{l}_s$ and a center given by the vanishing point $\mathbf{v}_x$, the principal point $\mathbf{x}_0$ of $\mathbf{P}$ will therefore lie on a line $\mathbf{l}_x$ passing through $\mathbf{v}_x$ and orthogonal to $\mathbf{l}_s$, and the focal length f will be equal to the square root of the product of the distances from the principal point $\mathbf{x}_0$ to $\mathbf{v}_x$ and to $\mathbf{l}_s$ respectively (see figure 3.5). As a result, the principal point can be estimated from 2 or more silhouettes of surfaces of revolution, and the focal length follows.

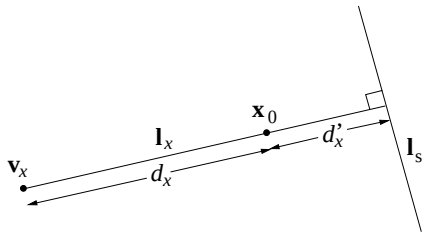


Figure 3.5: The vanishing point $\mathbf{v}_x$ and the image of the revolution axis $\mathbf{l}_s$ define a line $\mathbf{l}_x$ on which the principal point $\mathbf{x}_0$ must lie, and the focal length f is equal to $\sqrt{d_x d'_x}$.

Alternatively, consider the equation of the plane Π_s, which can be deduced from $\mathbf{P}$ and the image of the revolution axis $\mathbf{l}_s$, and is given by

$$\Pi_s = \mathbf{P}^T \mathbf{l}_s. \tag{3.13}$$

By definition, $\mathbf{v}_x$ is the vanishing point corresponding to the normal direction $\mathbf{N}_x$ of the plane Π_s, and hence

$$\mathbf{v}_x = \mathbf{P}\mathbf{N}_x. \tag{3.14}$$

By introducing the *absolute quadric* $\mathbf{\Omega} = \begin{bmatrix} \mathbb{I}_3 & \mathbb{O}_3 \\ \mathbb{O}_3^T & 0 \end{bmatrix}$ [128], equation (3.14) can

be rewritten as

$$\begin{aligned} \mathbf{v}_x &= \mathbf{P}\boldsymbol{\Omega}\Pi_{\mathrm{s}} \\ &= \mathbf{P}\boldsymbol{\Omega}\mathbf{P}^{\mathrm{T}}\mathbf{l}_{\mathrm{s}} \\ &= \boldsymbol{\omega}\ \mathbf{l}_{\mathrm{s}}, \end{aligned} \tag{3.15}$$

where $\boldsymbol{\omega} = \mathbf{K}\mathbf{K}^{\mathrm{T}}$ is the projection of the absolute quadric in $\mathbf{P}$, known as the *dual image of the absolute conic* [128]. Equation (3.15) gives the pole-polar relationship, with respect to the image of the absolute conic, between the vanishing point $\mathbf{v}_x$ of the normal direction of the plane Π_{s} and the vanishing line $\mathbf{l}_{\mathrm{s}}$ of Π_{s} [148]. By assuming the skew of $\mathbf{P}$ to be zero (i.e. $\varsigma = 0$), substituting (2.3) into (3.15) gives

$$\mathbf{v}_x = \begin{bmatrix} a^2 f^2 + u_0^2 & u_0 v_0 & u_0 \\ u_0 v_0 & f^2 + v_0^2 & v_0 \\ u_0 & v_0 & 1 \end{bmatrix} \mathbf{l}_{\mathrm{s}}, \tag{3.16}$$

where f, a and (u_0,v_0) are the intrinsic parameters of $\mathbf{P}$, as defined in Section 2.2.1. It follows that the harmonic homology associated with the silhouette of a surface of revolution will provide 2 constraints on the 4 intrinsic parameters of a camera. As a result, under the assumption of fixed intrinsic parameters and zero skew, it is possible to calibrate a camera from 2 or more silhouettes of surfaces of revolution. Further, if the aspect ratio is assumed to be 1 (i.e. $a = 1$), it can be derived from equation (3.16) that the focal length f is equal to the square root of the product of the distances from the principal point (u_0, v_0) to the vanishing point $\mathbf{v}_x$ and to the image of the revolution axis $\mathbf{l}_{\mathrm{s}}$. These results agree with the analysis of the vanishing points.

3.6 Algorithm and Implementation

3.6.1 Estimation of the Harmonic Homology W

The silhouette ρ of a surface of revolution is extracted from the image by applying a Canny edge detector [16] (see figure 3.6). The harmonic homology $\mathbf{W}$ that maps each side of the silhouette ρ to its symmetric counterpart is then estimated by minimizing the geometric distances between the original silhouette ρ and its transformed version $\rho' = \mathbf{W}\rho$. This can be done by sampling N evenly spaced points $\mathbf{x}_i$ along the silhouette ρ and optimizing the cost function

$$\mathrm{Cost}_{\mathbf{W}}(\mathbf{v}_x, \mathbf{l}_s) = \sqrt{\frac{1}{N}\sum_{i=1}^{N} \mathrm{dist}(\mathbf{W}(\mathbf{v}_x, \mathbf{l}_s)\mathbf{x}_i, \rho)^2}, \tag{3.17}$$

where $\mathrm{dist}(\mathbf{W}(\mathbf{v}_x, \mathbf{l}_s)\mathbf{x}_i, \rho)$ is the orthogonal distance from the transformed sample point $\mathbf{x}'_i = \mathbf{W}(\mathbf{v}_x, \mathbf{l}_s)\mathbf{x}_i$ to the original silhouette ρ.

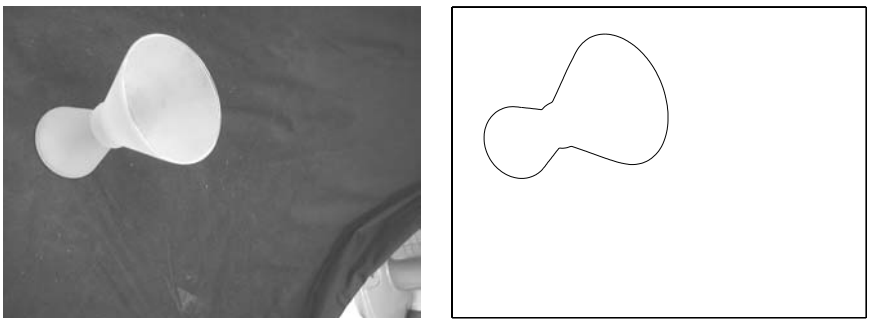


Figure 3.6: The silhouette of a surface of revolution (candle holder) is extracted by applying a Canny edge detector.

The successes of most nonlinear optimization problems require a good initialization so as to avoid convergence to local minima. This is achieved here by using bitangents of the silhouette [147]. Two points in the silhouette ρ near a bitangent are selected and a polynomial is fitted to the silhouette in the neighbor-

hood of each point. The bitangent and the bitangent points can then be obtained analytically from the 2 polynomials. Consider 2 corresponding bitangents $\mathbf{l}_b$ and $\mathbf{l}'_b$ on the 2 sides of ρ, with bitangent points $\mathbf{x}_1$, $\mathbf{x}_2$ and $\mathbf{x}'_1$, $\mathbf{x}'_2$ respectively (see figure 3.7). Let $\mathbf{l}_d$ be the line joining $\mathbf{x}_1$ and $\mathbf{x}'_2$, and $\mathbf{l}'_d$ be the line joining $\mathbf{x}'_1$ and $\mathbf{x}_2$. The intersection of $\mathbf{l}_b$ with $\mathbf{l}'_b$ and the intersection of $\mathbf{l}_d$ with $\mathbf{l}'_d$ define a line which will provide an estimate for the image of the revolution axis $\mathbf{l}_s$. Let $\mathbf{l}_c$ be the line joining $\mathbf{x}_1$ and $\mathbf{x}'_1$, and $\mathbf{l}'_c$ be the line joining $\mathbf{x}_2$ and $\mathbf{x}'_2$. The intersection of $\mathbf{l}_c$ with $\mathbf{l}'_c$ will provide an estimate for the vanishing point $\mathbf{v}_x$. The initialization of $\mathbf{l}_s$ and $\mathbf{v}_x$ from bitangents often provides an excellent initial guess for the optimization problem. This is generally good enough to avoid any local minimum and allows convergence to the global minimum in a small number of iterations. The estimation of the harmonic homology $\mathbf{W}$ is summarized in algorithm 3.1.

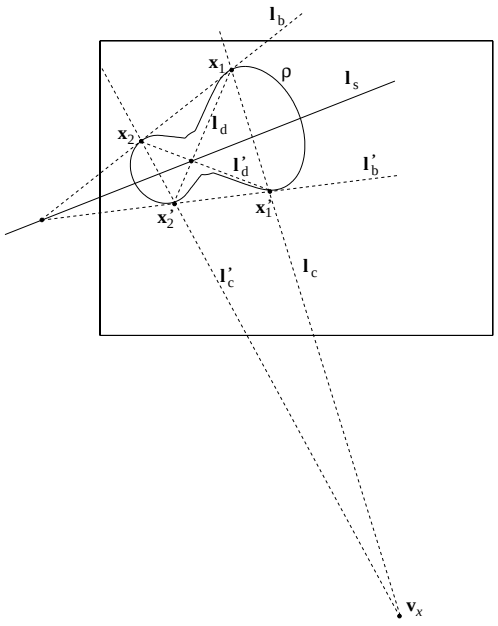


Figure 3.7: Initialization of the optimization parameters $\mathbf{l}_s$ and $\mathbf{v}_x$ from the bitangents and lines formed from the bitangent points.

Algorithm 3.1 Estimation of the harmonic homology $\mathbf{W}$.

extract the silhouette ρ of a surface of revolution
by applying a Canny edge detector;
sample N evenly spaced points $\mathbf{x}_i$ along ρ;
initialize the image of the revolution axis $\mathbf{l}_\mathrm{s}$ and the vanishing point $\mathbf{v}_x$
by identifying bitangents in ρ;
while not converged **do**
transform each point $\mathbf{x}_i$ by $\mathbf{W}$;
compute the distances from the transformed points $\mathbf{x}'_i = \mathbf{W}\mathbf{x}_i$ to ρ;
update $\mathbf{l}_\mathrm{s}$ and $\mathbf{v}_x$ to minimize the cost function in (3.17);
end while

3.6.2 Estimation of the Intrinsic Parameters

When the aspect ratio of the camera is 1, the line $\mathbf{l}_x$ passing through the principal point (u_0, v_0) and the vanishing point $\mathbf{v}_x$ will be orthogonal to the image of the revolution axis $\mathbf{l}_\mathrm{s}$ (see Section 3.5). Consider $\mathbf{v}_x = [v_1\ v_2\ v_3]^\mathrm{T}$ and $\mathbf{l}_\mathrm{s} = [l_1\ l_2\ l_3]^\mathrm{T}$. $\mathbf{l}_x$ can be expressed in terms of $\mathbf{v}_x$ and $\mathbf{l}_\mathrm{s}$, and is given by

$$\mathbf{l}_x = \begin{bmatrix} l_2 v_3 \\ -l_1 v_3 \\ l_1 v_2 - l_2 v_1 \end{bmatrix}. \tag{3.18}$$

Given 2 such lines $\mathbf{l}_{x1}$ and $\mathbf{l}_{x2}$, the principal point (u_0, v_0) will then be given by the intersection of $\mathbf{l}_{x1}$ with $\mathbf{l}_{x2}$. When more than 2 lines are available, the principal point (u_0, v_0) can be estimated by a linear least-squares method from

$$\begin{bmatrix} \mathbf{l}_{x1}^\mathrm{T} \\ \mathbf{l}_{x2}^\mathrm{T} \\ \vdots \\ \mathbf{l}_{xM}^\mathrm{T} \end{bmatrix} \begin{bmatrix} \alpha u_0 \\ \alpha v_0 \\ \alpha \end{bmatrix} = \mathbf{0}, \tag{3.19}$$

where $M \geq 2$ is the total number of lines (i.e. number of silhouettes) and α is a scale factor. The estimated principal point (u_0, v_0) is then projected onto each line

$\mathbf{l}_{xi}$ orthogonally as $\mathbf{x}_{0i}$, and the focal length f will be given by

$$f = \frac{1}{M}\sum_{i=1}^{M}\sqrt{\mathrm{dist}(\mathbf{x}_{0i}, \mathbf{v}_{xi}) \times \mathrm{dist}(\mathbf{x}_{0i}, \mathbf{l}_{si})}, \tag{3.20}$$

where $\mathrm{dist}(\mathbf{x}_{0i}, \mathbf{v}_{xi})$ is the distance between $\mathbf{x}_{0i}$ and $\mathbf{v}_{xi}$, and $\mathrm{dist}(\mathbf{x}_{0i}, \mathbf{l}_{si})$ is the orthogonal distance from $\mathbf{x}_{0i}$ to the image of the revolution axis $\mathbf{l}_{si}$. Note that the terms for summation are the focal lengths estimated from each pair of $\mathbf{v}_{xi}$ and $\mathbf{l}_{si}$ with the estimated principal point projected onto the corresponding $\mathbf{l}_{xi}$ (see Section 3.5), and the focal length f is then taken to be the mean of these estimated values.

The principal point (u_0, v_0) and the focal length f, obtained linearly from equations (3.19) and (3.20), can be further refined by optimizing the cost function

$$\mathrm{Cost}_{a=1}(f, u_0, v_0) = \sum_{i=1}^{N}\mathrm{dist}(\mathbf{K}\mathbf{K}^{\mathrm{T}}\mathbf{l}_{si}, \mathbf{v}_{xi})^2, \tag{3.21}$$

where $\mathbf{K}$ is the camera calibration matrix formed from f and (u_0, v_0), with zero skew and aspect ratio 1, as defined in equation (2.3), and $\mathrm{dist}(\mathbf{K}\mathbf{K}^{\mathrm{T}}\mathbf{l}_{si}, \mathbf{v}_{xi})$ is the distance between the point $\mathbf{v}'_{xi} = \mathbf{K}\mathbf{K}^{\mathrm{T}}\mathbf{l}_{\mathrm{s}}$ and $\mathbf{v}_{xi}$.

When the aspect ratio a of the camera is known but not equal to 1, there exists a planar homography $\mathbf{A}(a)$ that transforms the image into one that would have been obtained from a camera with the same focal length f, aspect ratio 1 and principal point (u'_0, v'_0). The homography $\mathbf{A}(a)$ is given by

$$\mathbf{A}(a) = \begin{bmatrix} \frac{1}{a} & 0 & -\frac{u_0}{a} + u'_0 \\ 0 & 1 & -v_0 + v'_0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{3.22}$$

where a is the aspect ratio of the original camera, and (u_0, v_0) and (u'_0, v'_0) are the principal points of the original and transformed cameras respectively. By setting

the principal point (u_0', v_0') of the transformed camera to $(u_0/a, v_0)$, the homography $\mathbf{A}(a)$ is reduced to

$$\mathbf{A}'(a) = \begin{bmatrix} \frac{1}{a} & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{3.23}$$

The vanishing points $\mathbf{v}_{xi}$ and the images of the revolution axis $\mathbf{l}_{si}$ are transformed by $\mathbf{A}'(a)$ and $\mathbf{A}'^{-\mathrm{T}}(a)$ respectively, and equations (3.18)–(3.21) can then be applied to obtain the principal point (u_0', v_0') and the focal length f. Note that the principal point (u_0', v_0') obtained in this way is the principal point of the transformed camera, and the principal point (u_0, v_0) of the original camera is simply given by

$$\begin{bmatrix} u_0 \\ v_0 \end{bmatrix} = \begin{bmatrix} a u_0' \\ v_0' \end{bmatrix}. \tag{3.24}$$

When the aspect ratio a of the camera is unknown, the camera can be calibrated by first assuming aspect ratio 1 to obtain (u_0, v_0) and f linearly, which are then used to initialize a full optimization with the cost function

$$\mathrm{Cost}_{\mathbf{K}}(f, a, u_0, v_0) = \sum_{i=1}^{N} \mathrm{dist}(\mathbf{K}\mathbf{K}^{\mathrm{T}}\mathbf{l}_{si}, \mathbf{v}_{xi})^2, \tag{3.25}$$

where $\mathbf{K}$ is the camera calibration matrix formed from f, a and (u_0, v_0), with zero skew, and $\mathrm{dist}(\mathbf{K}\mathbf{K}^{\mathrm{T}}\mathbf{l}_{si}, \mathbf{v}_{xi})$ is the distance between the point $\mathbf{v}_{xi}' = \mathbf{K}\mathbf{K}^{\mathrm{T}}\mathbf{l}_{\mathrm{s}}$ and $\mathbf{v}_{xi}$.

3.7 Degenerate Cases

3.7.1 Conic Silhouette

If the silhouette ρ of a surface of revolution is a conic, there will be an infinite number of harmonic homologies to which the silhouette ρ will be invariant. Such

a situation results in a degenerate case for camera calibration from surfaces of revolution.

Consider a conic represented by a 3×3 symmetric matrix $\mathbf{C}$, such that every point $\mathbf{x}$ on the conic satisfies

$$\mathbf{x}^{\mathrm{T}}\mathbf{C}\mathbf{x} = 0. \tag{3.26}$$

Given a point $\mathbf{x}_{\mathrm{e}}$ outside the conic $\mathbf{C}$, 2 tangents can be drawn from $\mathbf{x}_{\mathrm{e}}$ to $\mathbf{C}$ (see figure 3.8), and the line $\mathbf{l}_{\mathrm{e}}$ passing through the 2 tangent points is given by

$$\mathbf{l}_{\mathrm{e}} = \mathbf{C}\mathbf{x}_{\mathrm{e}}. \tag{3.27}$$

Let $\mathbf{W}_{\mathrm{e}}$ be a harmonic homology with axis $\mathbf{l}_{\mathrm{e}}$ and center $\mathbf{x}_{\mathrm{e}}$, given by

$$\mathbf{W}_{\mathrm{e}} \;=\; \mathbb{I}_3 - 2\frac{\mathbf{x}_{\mathrm{e}}\mathbf{l}_{\mathrm{e}}^{\mathrm{T}}}{\mathbf{x}_{\mathrm{e}}^{\mathrm{T}}\mathbf{l}_{\mathrm{e}}}. \tag{3.28}$$

Substituting (3.27) into (3.28) gives

$$\mathbf{W}_{\mathrm{e}} \;=\; \mathbb{I}_3 - 2\frac{\mathbf{x}_{\mathrm{e}}\mathbf{x}_{\mathrm{e}}^{\mathrm{T}}\mathbf{C}^{\mathrm{T}}}{\mathbf{x}_{\mathrm{e}}^{\mathrm{T}}\mathbf{C}\mathbf{x}_{\mathrm{e}}}. \tag{3.29}$$

Let $\mathbf{x}$ be a point on $\mathbf{C}$ and $\mathbf{x}' = \mathbf{W}_{\mathrm{e}}\mathbf{x}$, and consider the equation

$$\begin{aligned} \mathbf{x}'^{\mathrm{T}}\mathbf{C}\mathbf{x}' &= (\mathbf{W}_{\mathrm{e}}\mathbf{x})^{\mathrm{T}}\mathbf{C}(\mathbf{W}_{\mathrm{e}}\mathbf{x}) \\ &= \mathbf{x}^{\mathrm{T}}(\mathbf{W}_{\mathrm{e}}^{\mathrm{T}}\mathbf{C}\mathbf{W}_{\mathrm{e}})\mathbf{x}. \end{aligned} \tag{3.30}$$

Substituting (3.29) into (3.30) gives

$$\begin{aligned} \mathbf{x}'^{\mathrm{T}}\mathbf{C}\mathbf{x}' &= \mathbf{x}^{\mathrm{T}}[(\mathbb{I}_3 - 2\frac{\mathbf{x}_{\mathrm{e}}\mathbf{x}_{\mathrm{e}}^{\mathrm{T}}\mathbf{C}^{\mathrm{T}}}{\mathbf{x}_{\mathrm{e}}^{\mathrm{T}}\mathbf{C}\mathbf{x}_{\mathrm{e}}})^{\mathrm{T}}\mathbf{C}(\mathbb{I}_3 - 2\frac{\mathbf{x}_{\mathrm{e}}\mathbf{x}_{\mathrm{e}}^{\mathrm{T}}\mathbf{C}^{\mathrm{T}}}{\mathbf{x}_{\mathrm{e}}^{\mathrm{T}}\mathbf{C}\mathbf{x}_{\mathrm{e}}})]\mathbf{x} \\ &= \mathbf{x}^{\mathrm{T}}[(\mathbb{I}_3 - 2\frac{\mathbf{C}\mathbf{x}_{\mathrm{e}}\mathbf{x}_{\mathrm{e}}^{\mathrm{T}}}{\mathbf{x}_{\mathrm{e}}^{\mathrm{T}}\mathbf{C}\mathbf{x}_{\mathrm{e}}})(\mathbf{C} - 2\frac{\mathbf{C}\mathbf{x}_{\mathrm{e}}\mathbf{x}_{\mathrm{e}}^{\mathrm{T}}\mathbf{C}^{\mathrm{T}}}{\mathbf{x}_{\mathrm{e}}^{\mathrm{T}}\mathbf{C}\mathbf{x}_{\mathrm{e}}})]\mathbf{x} \\ &= \mathbf{x}^{\mathrm{T}}\mathbf{C}\mathbf{x} \\ &= 0. \end{aligned} \tag{3.31}$$

Equation (3.31) implies that any point $\mathbf{x}_e$ outside the conic $\mathbf{C}$ and the corresponding line $\mathbf{l}_e = \mathbf{C}\mathbf{x}_e$ will define a harmonic homology $\mathbf{W}_e$ to which the conic $\mathbf{C}$ will be invariant. As a result, if the silhouette of the surface of revolution is a conic, there will not be a unique solution for the optimization problem of the harmonic homology $\mathbf{W}$ associated with the silhouette, and hence it provides no information on the intrinsic parameters of the camera.

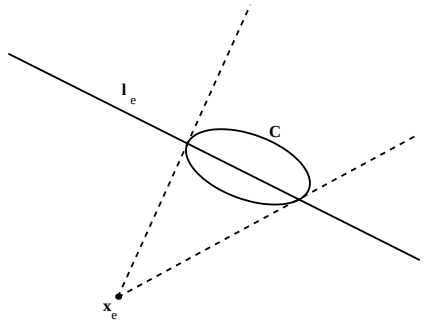


Figure 3.8: A conic $\mathbf{C}$ will be invariant to any harmonic homology with a center given by any point $\mathbf{x}_e$ outside the conic, and an axis given by $\mathbf{l}_e = \mathbf{C}\mathbf{x}_e$.

3.7.2 Vanishing Point at Infinity

When the camera is pointing towards the axis of revolution of the surface, the silhouette will exhibit bilateral or skew symmetry, and the vanishing point $\mathbf{v}_x$ will be at infinity (see Section 3.4). In this situation, the line $\mathbf{l}_x$ passing through the vanishing point $\mathbf{v}_x$ and being orthogonal to the image of the revolution axis $\mathbf{l}_s$ cannot be determined, nor is the distance $\mathrm{dist}(\mathbf{K}\mathbf{K}^{\mathrm{T}}\mathbf{l}_{si}, \mathbf{v}_{xi})$ in equations (3.21) and (3.25) defined. This causes the algorithm presented in Section 3.6 to fail. Nonetheless, it is obvious that the principal point is now constrained to lie on the image of the revolution axis. If the camera is pointing towards the axis in all images, then only the principal point can be estimated. In spite of that, such a

situation can easily be avoided during image acquisition and does not restrict the usefulness of the technique.

3.8 Experiments and Results

Experiments on both synthetic and real data were carried out, and the results are presented in the following subsections. In both cases, the cameras were assumed to have zero skew.

3.8.1 Synthetic Data

Generation of Data

The experimental setup consisted of a surface of revolution viewed by 3 identical synthetic cameras, as show in figure 3.9. The synthetic images had a dimension of 640×480 pixels, and the intrinsic parameters of the synthetic cameras were given by the calibration matrix

$$\mathbf{K} = \begin{bmatrix} 700 & 0 & 320 \\ 0 & 700 & 240 \\ 0 & 0 & 1 \end{bmatrix}. \tag{3.32}$$

The surface of revolution was composed of 2 spheres intersecting each other. Each sphere was represented by a 4×4 symmetric matrix $\mathbf{Q}_i$ whose projection was given by [31]

$$\mathbf{C}_{ij} = (\mathbf{P}_j \mathbf{Q}_i^{-1} \mathbf{P}_j^{\mathrm{T}})^{-1}, \tag{3.33}$$

where $\mathbf{P}_j$ was a 3×4 projection matrix and $\mathbf{C}_{ij}$ was a 3×3 symmetric matrix representing the conic, which was the projection of $\mathbf{Q}_i$ in $\mathbf{P}_j$. The silhouette of the surface of revolution in each image was found by projecting each sphere $\mathbf{Q}_i$

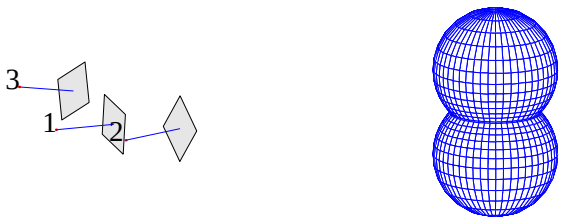


Figure 3.9: The experimental setup consisted of a surface of revolution, which was composed of 2 intersecting spheres, viewed by 3 identical synthetic cameras.

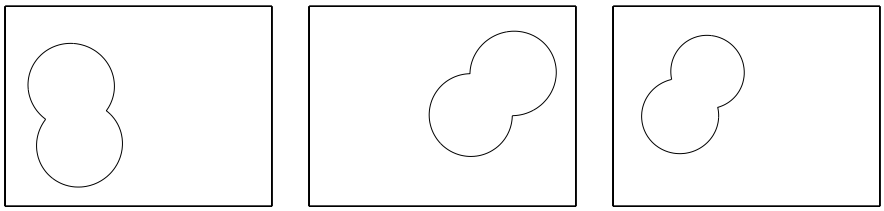
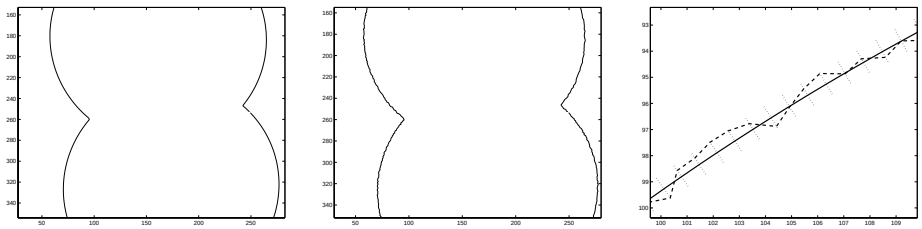


Figure 3.10: Silhouettes of the surface of revolution in the images taken by the 3 synthetic cameras.

onto the image j as the conic $\mathbf{C}_{ij}$ and finding points on each conic that lie outside the other conic. The silhouettes formed by the 3 cameras are shown in figure 3.10.

In order to evaluate the robustness of the algorithm described in Section 3.6, uniform random noise was added to each silhouette. Each point in the silhouette was perturbed in a direction normal to the local tangent, and the magnitudes of the noise were smoothed by a Gaussian filter so as to avoid unrealistic jaggedness along the silhouette (see figure 3.11).



(a) (b) (c)

Figure 3.11: (a) The original silhouette. (b) The resultant silhouette after uniform random noise of maximum 0.5 pixels being added. (c) The noise-free and noisy silhouettes are represented by solid and dash lines respectively, and the dotted lines indicate the bounds for noise along the normal direction of each point.

Results on Synthetic Data

Experiments on noise-free data (see figure 3.10) and data with 5 different noise levels were carried out. The 5 noise levels were 0.5, 0.7, 1.0, 1.2 and 1.5 pixels respectively. The noise level for typical real images is between 0.5 to 1.0 pixels, and the distortion of the silhouette will be too great to be realistic when the noise level is above 1.5 pixels (see figure 3.12).

For each noise level, 10 experiments were conducted using the algorithm described in Section 3.6. In the estimation of the harmonic homology $\mathbf{W}$, the number of sample points used was 100. Table 3.1 shows the mean values of the es-

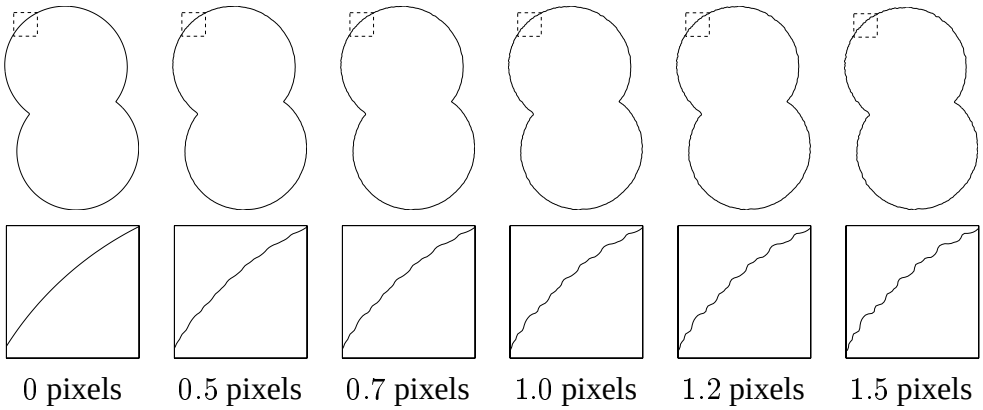


Figure 3.12: Silhouettes with noise levels of 0, 0.5, 0.7, 1.0, 1.2 and 1.5 pixels respectively. For noise level above 1.5 pixels, the distortion of the silhouette will be too great to be realistic.

timated intrinsic parameters over the 10 experiments for each noise level. The rms errors of the estimated intrinsic parameters for each noise level are listed in table 3.2, where the values in brackets are the percentage errors relative to the ground truth values. Table 3.2 shows that results obtained using the unknown aspect ratio method were slightly better than those obtained under the assumption of aspect ratio 1. As the noise level increased, the relative errors of the estimated intrinsic parameters increased. From figure 3.13, it can be seen that the normalized rms error of the focal length increased almost linearly with noise. For a noise level of 1.5 pixels, the error of the focal length was less than 4.5% and the error of the principal point was less than 10% in both x and y directions.

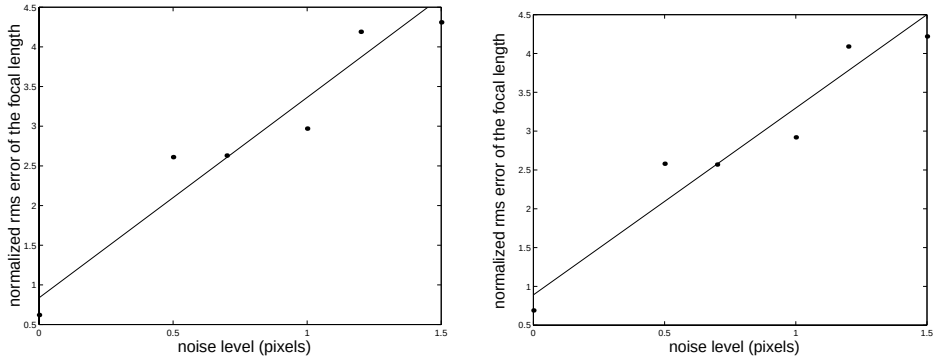
3.8.2 Real Data

The Ground Truth

The camera used in the real data experiments was a digital camera with a resolution of 640×480 pixels. The ground truth for the intrinsic parameters of the

Assumptions: zero skew and aspect ratio 1				
noise	f	-	u_0	v_0
0	695.66	-	319.34	262.18
0.5	689.68	-	317.43	252.17
0.7	696.02	-	321.13	251.21
1.0	696.63	-	323.07	248.71
1.2	695.18	-	322.71	248.42
1.5	701.85	-	325.34	244.21
Assumptions: zero skew				
noise	f	a	u_0	v_0
0	695.15	1.0016	319.60	261.92
0.5	689.51	1.0005	317.53	252.07
0.7	695.65	1.0012	321.35	250.98
1.0	696.18	1.0014	323.33	248.44
1.2	694.82	1.0011	322.95	248.16
1.5	701.51	1.0011	325.60	243.94

Table 3.1: Results of calibration from silhouettes under different noise levels. The intrinsic parameters listed are the mean values over the 10 experiments for each noise level.



aspect ratio 1 unknown aspect ratio

Figure 3.13: The normalized rms errors of the estimated focal length for different noise levels.

Assumptions: zero skew and aspect ratio 1				
noise	f	-	u_0	v_0
0	4.34 (0.62%)	-	0.66 (0.21%)	22.18 (9.24%)
0.5	18.25 (2.61%)	-	6.88 (2.15%)	13.57 (5.66%)
0.7	18.41 (2.63%)	-	8.32 (2.60%)	13.26 (5.53%)
1.0	20.78 (2.97%)	-	11.02 (3.44%)	16.12 (6.72%)
1.2	29.31 (4.19%)	-	12.94 (4.04%)	16.59 (6.91%)
1.5	30.14 (4.31%)	-	15.79 (4.94%)	18.78 (7.83%)
Assumptions: zero skew				
noise	f	a	u_0	v_0
0	4.85 (0.69%)	0.0016 (0.16%)	0.40 (0.12%)	21.92 (9.13%)
0.5	18.04 (2.58%)	0.0015 (0.15%)	7.02 (2.20%)	13.53 (5.64%)
0.7	18.01 (2.57%)	0.0021 (0.21%)	8.63 (2.70%)	13.20 (5.50%)
1.0	20.44 (2.92%)	0.0020 (0.20%)	11.36 (3.55%)	16.22 (6.76%)
1.2	28.62 (4.09%)	0.0028 (0.28%)	13.39 (4.18%)	16.73 (6.97%)
1.5	29.54 (4.22%)	0.0028 (0.28%)	16.32 (5.10%)	19.05 (7.94%)

Table 3.2: The rms errors of the estimated intrinsic parameters for each noise level. The values in brackets are the percentage errors relative to the ground truth values.

camera was obtained using a calibration grid. Eleven images of a calibration grid were taken with the camera at different orientations (see figure 3.14). Corner features were extracted from each image using a Canny edge detector [16] and line fitting techniques. For each image, the camera was calibrated using the DLT technique [1] followed by an optimization which minimized the reprojection errors of the corner features [42, 40]. The results of calibration from the calibration grid are shown in table 3.3.

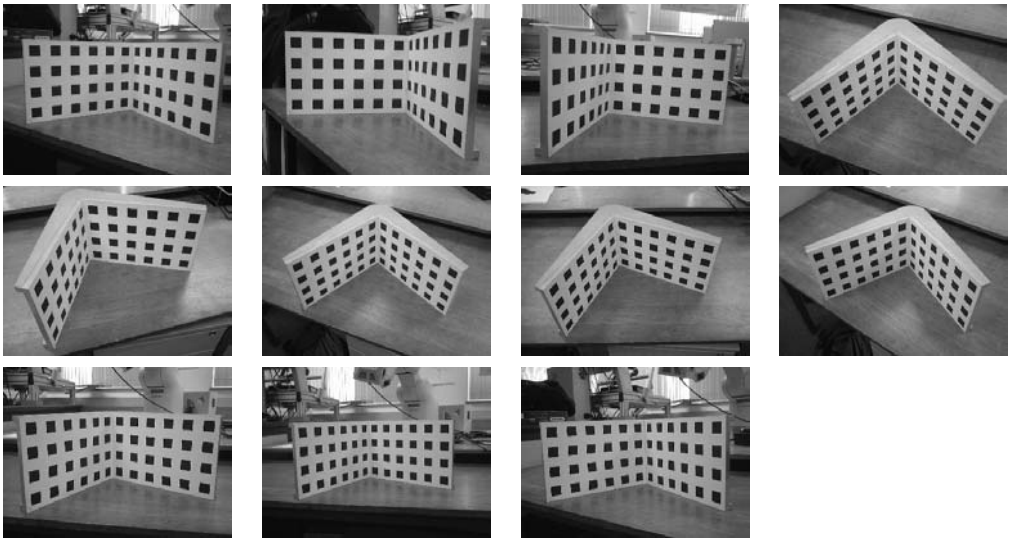


Figure 3.14: Eleven images of a calibration grid taken by the digital camera for calibration.

Results on Real Data

Two sequences of real images of surfaces of revolution were used for the calibration of the digital camera. The first sequence consisted of 3 images of 2 bowls, which provided 4 silhouettes of surfaces of revolution (see figure 3.15). The second sequence consisted of 8 images of a candle holder, which provided 8 silhouettes of surface of revolution (see figure 3.16). The results of calibration

Assumptions: zero skew and aspect ratio 1				
-	f	-	u_0	v_0
mean	684.98	-	322.60	232.15
std	3.49	-	3.47	3.93
Assumptions: zero skew				
-	f	a	u_0	v_0
mean	685.52	0.9992	322.60	232.15
std	3.38	0.0020	3.46	3.94

Table 3.3: Results of calibration from 11 images of the calibration grid.

from the 2 image sequences are shown in table 3.4. Table 3.5 shows the percentage errors of the estimated intrinsic parameters relative to the ground truth values. Figure 3.17 shows the lines $\mathbf{l}_{xi}$ passing through the corresponding vanishing point $\mathbf{v}_{xi}$ and orthogonal to the corresponding image of the revolution axis $\mathbf{l}_{si}$.

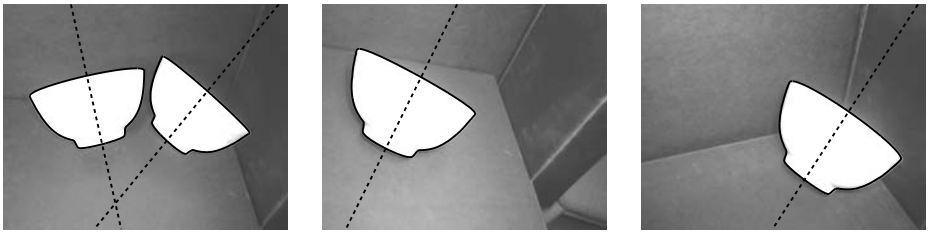


Figure 3.15: Three images of 2 bowls with the extracted silhouettes and estimated images of the revolution axis plotted in solid and dash lines respectively.

From table 3.4 and table 3.5, it can be seen that the intrinsic parameters estimated from the candle holder sequence were better than those from the bowls sequence. This can be explained as the silhouettes in the candle holder sequence showed much greater perspective effect than those in the bowls sequence (see figure 3.15 and figure 3.16). Besides, the candle holder sequence also provided more silhouettes, and hence more constraints, than the bowls sequence for the estimation of the intrinsic parameters. The focal length estimated from the bowls

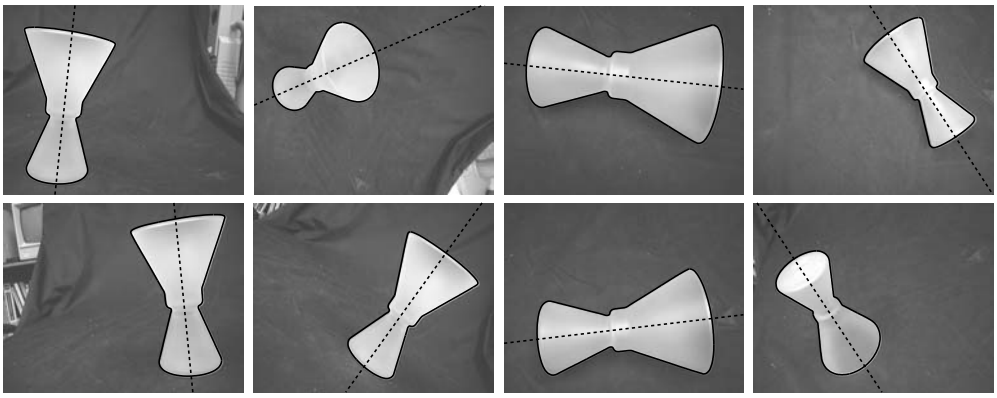


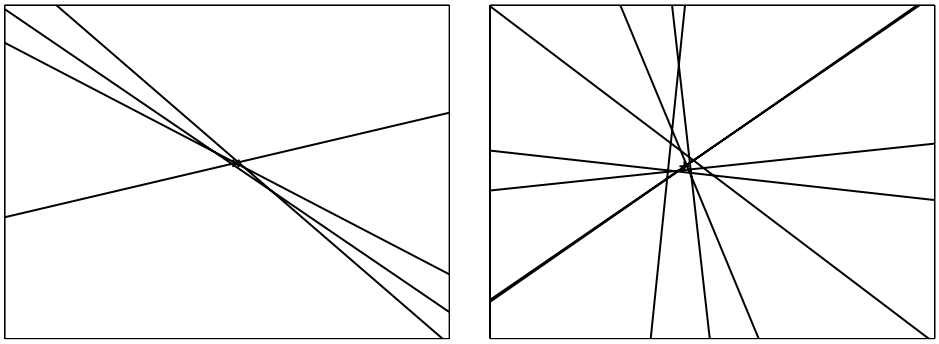
Figure 3.16: Eight images of a candle holder with the extracted silhouettes and estimated images of the revolution axis plotted in solid and dash lines respectively.

Assumptions: zero skew and aspect ratio 1				
image set	f	-	u_0	v_0
Bowls	708.34	-	320.65	245.58
Candle holder	703.21	-	329.90	232.96
Assumptions: zero skew				
image set	f	a	u_0	v_0
Bowls	708.95	0.9987	320.59	245.63
Candle holder	694.75	1.0360	328.99	232.06

Table 3.4: Results of calibration from the bowls and candle holder sequences.

Assumptions: zero skew and aspect ratio 1				
image set	f	-	u_0	v_0
Bowls	3.41%	-	-0.60%	5.79%
Candle holder	2.66%	-	2.26%	0.35%
Assumptions: zero skew				
image set	f	a	u_0	v_0
Bowls	3.42%	-0.05%	-0.62%	5.81%
Candle holder	1.35%	3.68%	1.98%	-0.04%

Table 3.5: Percentage errors in the results of calibration from the bowls and candle holder sequences.



bowls sequence candle holder sequence

Figure 3.17: The solid lines represent the lines $\mathbf{l}_{xi}$ passing through the corresponding vanishing point $\mathbf{v}_{xi}$ and orthogonal to the corresponding axis of revolution $\mathbf{l}_{si}$. Since the principal point $\mathbf{x}_0$ must lie on these lines, it can be estimated as the intersection of 2 or more lines $\mathbf{l}_{xi}$.

sequence had an error less than 3.5% relative to the ground truth focal length. For the candle holder sequence, the error of the estimated focal length decreased from 2.66% to 1.35% when the assumption of aspect ratio 1 was dropped. Note that in both synthetic and real data experiments, the estimated focal length tended to be closer to the ground truth value when the aspect ratio was allowed to change to an incorrect value. This may be due to the fact that the cost functions given by equations (3.21) and (3.25) are only some algebraic errors. It suggests that a proper cost function should consist of the geometric errors between the original and transformed silhouettes instead, like the one given in equation (3.17) for the estimation of the harmonic homology.

3.9 Discussions

By exploiting the symmetry property exhibited in the silhouettes of surfaces of revolution and the property of vanishing points, a practical technique for camera

calibration has been developed. The use of surfaces of revolution makes the calibration process easier, in not requiring the use of any precisely machined device with known geometry such as a calibration grid. Besides, a surface of revolution can always be generated by rotating an object of any arbitrary shape around a fixed axis. Despite the fact that strong perspective effect is required, the proposed method is promising, as demonstrated by the experimental results on both synthetic and real data. The focal lengths were estimated with high accuracy, having an error of only around 4% with respect to the ground truth.

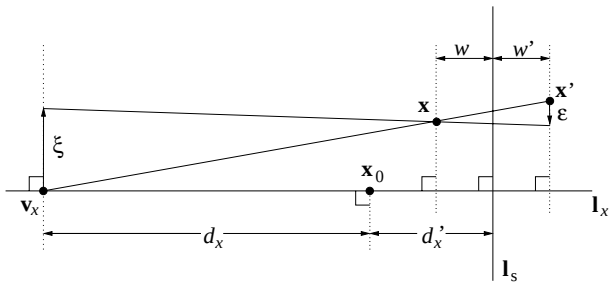


Figure 3.18: Error analysis in the estimation of the principal point as the focal length varies.

Experiments show that in estimating the harmonic homology $\mathbf{W}$ associated with the silhouette ρ of a surface of revolution, the uncertainty is essentially in the vanishing point $\mathbf{v}_x$. Since $\mathbf{v}_x$ is, in general, tens of thousands of pixels away from the axis $\mathbf{l}_s$, its error in a direction orthogonal to $\mathbf{l}_s$ is negligible in the computation of the principal point and focal length. On the other hand, the error of $\mathbf{v}_x$ in a direction parallel to $\mathbf{l}_s$ will lead to the same error in the estimated principal point $\mathbf{x}_0$. This is due to the fact that, under the assumptions of zero skew and aspect ratio 1, $\mathbf{x}_0$ must lie on the line $\mathbf{l}_x$ passing through $\mathbf{v}_x$ and orthogonal to $\mathbf{l}_s$ (see Section 3.5). Figure 3.18 shows a point $\mathbf{x}$ in ρ which is transformed by $\mathbf{W}$ to its

symmetric counterpart $\mathbf{x}'$ in ρ. If $\mathbf{v}_x$ has an error ξ in a direction parallel to $\mathbf{l}_s$, then the transformed point will have an error ε (see figure 3.18). It is easy to see that ξ and ε are related to each other by

$$\frac{\xi}{\varepsilon} = \frac{d_x + d'_x - w}{w + w'}. \tag{3.34}$$

Since $d_x = f^2/d'_x$ is much greater than d'_x, w and w', and that w and w' are roughly equal with respect to d_x, equation (3.34) can be rewritten as

$$\frac{\xi}{\varepsilon} \simeq \frac{f^2}{2d'_x w}. \tag{3.35}$$

Equation (3.35) implies that if d'_x, w and the cost given by equation (3.17) after the optimization are assumed to be constant, then the error ξ of $\mathbf{v}_x$, and hence the error of the principal point $\mathbf{x}_0$, in a direction parallel to $\mathbf{l}_s$ will be proportional to f^2. This limits the usefulness of the technique to wide angle cameras.

Chapter 4

Reconstruction of Surfaces of Revolution from Single View

"Beauty depends on size as well as symmetry."

- Aristotle, *Poetics*, ch. 7, sec. 4.

4.1 Introduction

2D images contain cues to surface shape and orientation, however their interpretations are inherently ambiguous because depth information is lost during the image formation process when 3D structures in the world are projected onto 2D images. Multiple images from different viewpoints can be used to resolve these ambiguities, and this results in techniques like *stereo vision* [70, 6] and *structure from motion* [132, 79]. Nonetheless, under certain appropriate assumptions, it is possible to infer scene structure, like surface orientation and curvature, from a single image. Examples of such techniques include *shape from shading* [58, 61, 142, 59, 145] under the assumptions of point light source and Lambertian surface, *shape from line drawings* [112, 50, 60, 28, 136, 83, 131, 122, 84, 102] under the assumption of trihedral-vertex polyhedral scene structure or smooth-

ness, *shape from texture* [48, 137, 34, 65, 11, 86, 85] under the assumption of homogeneous or isotropic texture, and *shape from contour* [8, 68, 69, 24] under the assumption of viewing a smooth object.

In this chapter, a simple technique for recovering the 3D shape of a surface of revolution from a single view is introduced. The image of the surface of revolution is first rectified by a planar homography so that the resulting silhouette exhibits bilateral symmetry. Surface normals along the contour generator are then determined from the rectified silhouette, and depth information can then be recovered using a coplanarity constraint between the surface normal and the axis of revolution.

Section 4.2 briefly reviews existing techniques of shape from contour from single view in the literature. Section 4.3 presents a parameterization for surfaces of revolution and studies the surface geometry of surfaces of revolution. In particular, the surface normal and the axis of revolution are shown to be coplanar. This coplanarity constraint is exploited in Section 4.4 to derive a simple technique for reconstructing a surface of revolution from a single view using its silhouette. The ambiguity in the reconstruction under a general camera configuration is studied and analyzed in Section 4.5. It is shown that such an ambiguity cannot be described by a projective transformation. The algorithm and implementation are described in Section 4.6 and results of real data experiments are presented in Section 4.7. Finally discussions are given in Section 4.8.

4.2 Previous Works

The earliest study of silhouettes in the literature dates back to 1978, when Barrow and Tenenbaum [7, 8] showed that surface orientation along the silhouette can be computed directly from image data. In his book [88], Marr pointed out that it is possible to infer the sign of the Gaussian curvature of an object from its silhouette. His observations were made more precise by Koenderink [68, 69] who showed that the sign of the Gaussian curvature is equal to the sign of the curvature of the silhouette, and convexities, concavities and inflections of the silhouette indicate convex, hyperbolic and parabolic surface points respectively. In [24], Cipolla and Blake showed that the curvature of the silhouette has the same sign as the normal curvature along the contour generator under perspective projection. A similar result was derived for orthographic projection by Brady et al. in [13].

In all the above studies mentioned, the authors only made use of a single monocular image to infer geometric information from the silhouette. In fact, if some strong a priori knowledge of the object is available, like a parametric description, then a single view alone allows shape recovery. Due to its expressiveness, generalized cylinders (GCs), introduced by Binford [10] in 1971, are commonly used as a parametric description for visual representation. The invariant properties of straight homogeneous generalized cylinders (SHGCs) and their silhouettes had been studied by various researchers [57, 106, 77], and exploited for object recognition and object pose estimation. In [114, 49, 144], algorithms for segmentation and 3D recovery of SHGCs under orthographic projection were presented. In [133], Ulupinar and Nevatia addressed the recovery of curved-axis planar right constant generalized cylinders (PRCGCs) under orthographic projection.

Their idea was further developed by Zerroug and Nevatia [143] who implemented a technique for segmentation and 3D recovery of both PRCGCs and circular planar right generalized cylinders (circular PRGCs) from a single real image, under orthographic projection.

This chapter addresses the problem of recovering the 3D shape of a surface of revolution from a single view. Surfaces of revolution belong to a subclass of SHGCs, in which the planar cross-section is a circle centered at and orthogonal to the axis. The work presented here is different from previous works [133, 143] in that rather than orthographic projection, which is a quite restricted case, perspective projection is assumed. Like other methods for shape recovery of GCs from a single view, the algorithm introduced here makes use of the invariant property of the surface of revolution and its silhouette to locate the image of the revolution axis. The algorithm also uses the information of the image of the revolution axis to rectify the image so that the resulting silhouette exhibits bilateral symmetry. Such a rectification leads to a simpler differential analysis of the silhouette and yields a simple equation for depth recovery.

4.3 Surface of Revolution

Let $\tilde{\mathbf{C}}_{\mathrm{r}}(s) = [X(s)\ Y(s)\ 0]^{\mathrm{T}}$ be a regular and differentiable planar curve on the x-y plane where $X(s) > 0$ for all s. A surface of revolution can be generated by rotating $\tilde{\mathbf{C}}_{\mathrm{r}}$ about the y-axis, and is given by

$$\tilde{\mathbf{S}}_{\mathrm{r}}(s,\theta) = \begin{bmatrix} X(s)\cos\theta \\ Y(s) \\ X(s)\sin\theta \end{bmatrix}, \tag{4.1}$$

where θ is the angle parameter for a complete circle. The tangent plane basis vectors

$$\frac{\partial \tilde{\mathbf{S}}_{\mathrm{r}}}{\partial s} = \begin{bmatrix} \dot{X}(s)\cos\theta \\ \dot{Y}(s) \\ \dot{X}(s)\sin\theta \end{bmatrix} \text{ and } \frac{\partial \tilde{\mathbf{S}}_{\mathrm{r}}}{\partial \theta} = \begin{bmatrix} -X(s)\sin\theta \\ 0 \\ X(s)\cos\theta \end{bmatrix} \tag{4.2}$$

are independent since $\dot{X}(s)$ and $\dot{Y}(s)$ are never both zeros at the same time, and $X(s) > 0$ for all s. Hence $\tilde{\mathbf{S}}_{\mathrm{r}}$ is immersed and has a well-defined tangent plane at each point, with the normal given by

$$\begin{aligned} \mathbf{n}(s,\theta) &= \frac{\partial \tilde{\mathbf{S}}_{\mathrm{r}}}{\partial s} \times \frac{\partial \tilde{\mathbf{S}}_{\mathrm{r}}}{\partial \theta} \\ &= \begin{bmatrix} X(s)\dot{Y}(s)\cos\theta \\ -X(s)\dot{X}(s) \\ X(s)\dot{Y}(s)\sin\theta \end{bmatrix}. \end{aligned} \tag{4.3}$$

Through any point $\tilde{\mathbf{S}}_{\mathrm{r}}(s_0,\theta_0)$ on the surface, there is a *meridian curve* which is the curve obtained by rotating $\tilde{\mathbf{C}}_{\mathrm{r}}$ about the y-axis by an angle $-\theta_0$, and a *latitude circle* which is a circle on the plane $y = Y(s_0)$ and with its center on the y-axis. Note that the meridian curve and the latitude circle are orthogonal to each other, and they form the principal curves of the surface (see figure 4.1). It follows from equation (4.3) that the surface normal at $\tilde{\mathbf{S}}_{\mathrm{r}}(s_0,\theta_0)$ lies on the plane containing the y-axis and the point $\tilde{\mathbf{S}}_{\mathrm{r}}(s_0,\theta_0)$, and is normal to the meridian curve through $\tilde{\mathbf{S}}_{\mathrm{r}}(s_0,\theta_0)$. By circular symmetry, the surface normals along a latitude circle will all meet at one point on the y-axis.

4.4 Reconstruction from a Single View

Consider a surface of revolution $\tilde{\mathbf{S}}_{\mathrm{r}}$ whose axis of revolution coincides with the y-axis, and a pin-hole camera $\hat{\mathbf{P}} = [\mathbb{I}_3\ \mathbf{t}]$ where $\mathbf{t} = [0\ 0\ d_z]^{\mathrm{T}}$ and $d_z > 0$. Let the

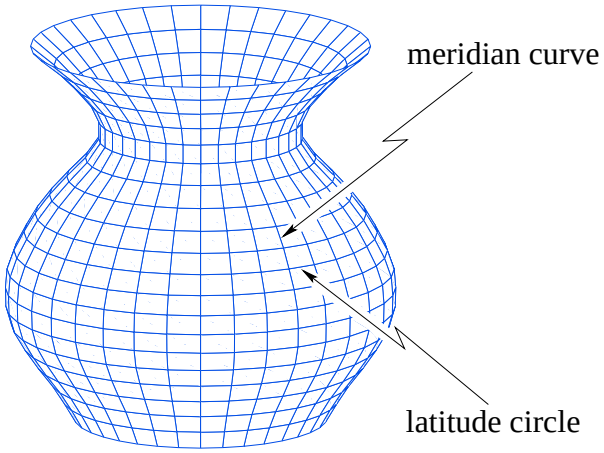


Figure 4.1: The meridian curves and latitude circles form the principal curves of the surface of revolution.

contour generator be parameterized by s as

$$\tilde{\Gamma}(s) = \tilde{\mathbf{c}} + \lambda(s)\mathbf{p}(s), \text{ where} \tag{4.4}$$

$$\mathbf{p}(s) \cdot \mathbf{n}(s) = 0. \tag{4.5}$$

In equation (4.4), $\tilde{\mathbf{c}}$ indicates the camera center at $[0\ 0\ -d_z]^{\mathrm{T}}$, $\mathbf{p}(s)$ is the viewing vector from $\tilde{\mathbf{c}}$ to the focal plane at unit distance for the point $\tilde{\Gamma}(s)$, and $\lambda(s)$ is the depth of the point $\tilde{\Gamma}(s)$ from $\tilde{\mathbf{c}}$ along the z direction. Note that $\mathbf{p}(s)$ has the form $[x(s)\ y(s)\ 1]^{\mathrm{T}}$, where $(x(s), y(s))$ is a point in the silhouette. Equation (4.5) expresses the tangency constraint, where $\mathbf{n}(s)$ is the unit surface normal at $\tilde{\Gamma}(s)$. It has been shown in Section 2.4.2 that $\mathbf{n}(s)$ can be determined up to a sign by

$$\begin{aligned} \mathbf{n}(s) &= \frac{\mathbf{p}(s) \times \frac{d\,\mathbf{p}(s)}{d\,s}}{\left|\mathbf{p}(s) \times \frac{d\,\mathbf{p}(s)}{d\,s}\right|}, \\ &= \frac{1}{\alpha_{\mathrm{n}}(s)} \begin{bmatrix} -\dot{y}(s) \\ \dot{x}(s) \\ x(s)\dot{y}(s) - \dot{x}(s)y(s) \end{bmatrix}, \end{aligned} \tag{4.6}$$

where $\alpha_{\mathrm{n}}(s) = \left|\mathbf{p}(s) \times \frac{d\,\mathbf{p}(s)}{d\,s}\right|$. In Section 4.3, it has been shown that the surface normal $\mathbf{n}(s)$ will lie on the plane containing the y-axis and the point $\tilde{\Gamma}(s)$. This coplanarity constraint can be expressed as

$$\mathbf{n}(s)^{\mathrm{T}}[\mathbf{n}_y]_{\times}\tilde{\Gamma}(s) = 0, \tag{4.7}$$

where $\mathbf{n}_y = [0\ 1\ 0]^{\mathrm{T}}$. Let $\mathbf{n}(s) = [n_1(s)\ n_2(s)\ n_3(s)]^{\mathrm{T}}$ and expanding (4.7) gives

$$\begin{aligned}
\begin{bmatrix} n_1(s) & n_2(s) & n_3(s) \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \lambda(s)x(s) \\ \lambda(s)y(s) \\ \lambda(s) - d_z \end{bmatrix} &= 0 \\
\begin{bmatrix} n_1(s) & n_2(s) & n_3(s) \end{bmatrix} \begin{bmatrix} \lambda(s) - d_z \\ 0 \\ -\lambda(s)x(s) \end{bmatrix} &= 0 \\
n_1(s)(\lambda(s) - d_z) - n_3(s)\lambda(s)x(s) &= 0.
\end{aligned} \tag{4.8}$$

By rearranging (4.8), the depth of the point $\tilde{\Gamma}(s)$ is given by

$$\lambda(s) = \frac{d_z n_1(s)}{n_1(s) - n_3(s)x(s)}. \tag{4.9}$$

Hence, the contour generator can be recovered from the silhouette and is given by

$$\begin{aligned}
\Gamma(s) &= \begin{bmatrix} \tilde{\mathbf{c}} + \lambda(s)\mathbf{p}(s) \\ 1 \end{bmatrix} \\
&= \begin{bmatrix} \tilde{\mathbf{c}} + \frac{d_z n_1(s)}{n_1(s) - n_3(s)x(s)}\mathbf{p}(s) \\ 1 \end{bmatrix} \\
&= \begin{bmatrix} d_z\dot{y}(s)x(s) \\ d_z\dot{y}(s)y(s) \\ d_z\alpha_{\Gamma}(s) \\ \dot{y}(s) - \alpha_{\Gamma}(s) \end{bmatrix},
\end{aligned} \tag{4.10}$$

where $\alpha_{\Gamma}(s) = (\dot{x}(s)y(s) - x(s)\dot{y}(s))x(s)$. Since the distance d_z cannot be recovered from the image, the reconstruction is determined only up to a similarity transformation. The surface of revolution can then be obtained by rotating the

contour generator about the y-axis, and is given by

$$\tilde{\mathbf{S}}_{\mathrm{r}}(s,\theta) = \begin{bmatrix} X(s)\cos\theta \\ Y(s) \\ X(s)\sin\theta \end{bmatrix}, \tag{4.11}$$

where $X(s) = \sqrt{(\lambda(s)x(s))^2 + (\lambda(s) - d_z)^2}$ and $Y(s) = \lambda(s)y(s)$.

Now consider an arbitrary pin-hole camera $\mathbf{P}$ by introducing the intrinsic parameters represented by the camera calibration matrix $\mathbf{K}$ to $\hat{\mathbf{P}}$, and by applying the rotation $\mathbf{R}$ to $\hat{\mathbf{P}}$ about its optical center. Hence $\mathbf{P} = \mathbf{K}\mathbf{R}[\mathbb{I}_3 \ \mathbf{t}]$ or $\mathbf{P} = \mathbf{H}\hat{\mathbf{P}}$, where $\mathbf{H} = \mathbf{K}\mathbf{R}$. From the discussions presented in Section 3.4, the resulting silhouette of $\tilde{\mathbf{S}}_{\mathrm{r}}$ will be invariant to a harmonic homology $\mathbf{W}$. Given $\mathbf{K}$ and $\mathbf{W}$, it is possible to rectify the image by a planar homography $\mathbf{H}_{\text{rectify}}$ so that the silhouette becomes bilaterally symmetric about the line $\hat{\mathbf{l}}_{\mathrm{s}} = [1\ 0\ 0]^{\mathrm{T}}$, and hence be invariant to $\mathbf{T}$ (see Section 3.4). This corresponds to normalizing the camera by $\mathbf{K}^{-1}$ and rotating the normalized camera until the revolution axis of $\tilde{\mathbf{S}}_{\mathrm{r}}$ lies on the y-z plane of the camera coordinate system. Note that $\mathbf{H}_{\text{rectify}}$ is not unique, as any homography $\mathbf{H}'_{\text{rectify}}$, given by

$$\mathbf{H}'_{\text{rectify}} = \mathbf{R}_x(\psi)\mathbf{H}_{\text{rectify}} \tag{4.12}$$

where

$$\mathbf{R}_x(\psi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\psi & -\sin\psi \\ 0 & \sin\psi & \cos\psi \end{bmatrix} \tag{4.13}$$

is a rotation about the x-axis by an angle ψ, will yield a silhouette which will be invariant to $\mathbf{T}$ (see Appendix C). There exists ψ_0 such that $\mathbf{R}_x(\psi_0)\mathbf{H}_{\text{rectify}}\mathbf{P} = \hat{\mathbf{P}}$ and the surface of revolution can be reconstructed from the rectified image using the algorithm presented above. In general, ψ_0 cannot be recovered from a single

image and hence there will be an 1-parameter family of solutions for the contour generator, given by

$$\Gamma^{\psi}(s) \;=\; \begin{bmatrix} d_z\dot{y}(s)x(s) \\ d_z\dot{y}(s)(y(s)\cos\psi - \sin\psi) \\ d_z\alpha_{\Gamma}^{\psi}(s) \\ \dot{y}(s)(y(s)\sin\psi + \cos\psi) - \alpha_{\Gamma}^{\psi}(s) \end{bmatrix} \tag{4.14}$$

where $\alpha_{\Gamma}^{\psi}(s) = \{(\dot{x}(s)y(s) - x(s)\dot{y}(s))\cos\psi - \dot{x}(s)\sin\psi\}x(s)$. A detail derivation for Γ^{ψ} is given in Appendix C. The 1-parameter family of surfaces of revolution $\tilde{\mathbf{S}}_{\Gamma}^{\psi}$ can be obtained by rotating Γ^{ψ} about the y-axis. Note that the ambiguity in the reconstruction corresponds to the ambiguity of the orientation of the revolution axis on the y-z plane of the camera coordinate system. If the image of a latitude circle in the surface of revolution can be localized, the orientation of the revolution axis relative to the y-axis of the camera coordinate system can be estimated (see Appendix D), which removes the ambiguity in the reconstruction. Alternatively, the ambiguity can also be removed by knowing the ratio of the radius of any latitude circle in the surface of revolution to the height of the surface of revolution.

4.5 Analysis of the Ambiguity in the Reconstruction

A projective transformation that maps a surface of revolution to another surface of revolution, both with the y-axis as their axes of revolution, has the following generic form

$$\mathbf{H}_{\mathrm{SOR}} = \begin{bmatrix} \cos\varrho & 0 & \sin\varrho & 0 \\ 0 & h_1 & 0 & h_2 \\ \mp\sin\varrho & 0 & \pm\cos\varrho & 0 \\ 0 & h_3 & 0 & h_4 \end{bmatrix} \quad \text{where} \quad \begin{vmatrix} h_1 & h_2 \\ h_3 & h_4 \end{vmatrix} \neq 0. \tag{4.15}$$

A detail derivation for $\mathbf{H}_{\mathrm{SOR}}$ can be found in Appendix E. Assuming that the ambiguity in the reconstruction of the surface of revolution can be described by $\mathbf{H}_{\mathrm{SOR}}$, then both $\mathbf{H}_{\mathrm{SOR}}$ and the transformation induced by $\mathbf{R}_x(\psi)$ will map a latitude circle in $\tilde{\mathbf{S}}_{\mathrm{r}}$ to the same latitude circle in $\tilde{\mathbf{S}}_{\mathrm{r}}^{\psi}$, as a latitude circle is by itself a surface of revolution in the limiting case. Hence, if the ambiguity is projective, there exists ϱ for each $y(s)$ such that $\Gamma^{\psi}(s) = \mathbf{H}_{\mathrm{SOR}}\Gamma(s)$. In Cartesian coordinates, the projective transformation $\Gamma^{\psi}(s) = \mathbf{H}_{\mathrm{SOR}}\Gamma(s)$, with $d_z = 1$, is given by the set of equations

$$\frac{\dot{y}(s)x(s)}{\dot{y}(s)(y(s)\sin\psi + \cos\psi) - \alpha_{\Gamma}^{\psi}(s)} = \frac{\dot{y}(s)x(s)\cos\varrho + \alpha_{\Gamma}(s)\sin\varrho}{h_3\dot{y}(s)y(s) + h_4(\dot{y}(s) - \alpha_{\Gamma}(s))}, \quad (4.16)$$

$$\frac{\dot{y}(s)(y(s)\cos\psi - \sin\psi)}{\dot{y}(s)(y(s)\sin\psi + \cos\psi) - \alpha_{\Gamma}^{\psi}(s)} = \frac{h_1\dot{y}(s)y(s) + h_2(\dot{y}(s) - \alpha_{\Gamma}(s))}{h_3\dot{y}(s)y(s) + h_4(\dot{y}(s) - \alpha_{\Gamma}(s))}, \quad (4.17)$$

$$\frac{\alpha_{\Gamma}^{\psi}(s)}{\dot{y}(s)(y(s)\sin\psi + \cos\psi) - \alpha_{\Gamma}^{\psi}(s)} = \frac{\mp\dot{y}(s)x(s)\sin\varrho \pm \alpha_{\Gamma}(s)\cos\varrho}{h_3\dot{y}(s)y(s) + h_4(\dot{y}(s) - \alpha_{\Gamma}(s))}. \quad (4.18)$$

Rearranging (4.17) gives

$$\begin{aligned}
0 \;=\;& h_2\cos\psi x(s)^4\dot{y}(s)^2 + h_2\sin\psi x(s)^3\dot{x}(s)\dot{y}(s) - \\
& 2h_2\cos\psi x(s)^3\dot{x}(s)y(s)\dot{y}(s) + \\
& (h_1\cos\psi + h_2\sin\psi - h_4\cos\psi)x(s)^2y(s)\dot{y}(s)^2 + \\
& (2h_2\cos\psi + h_4\sin\psi)x(s)^2\dot{y}(s)^2 + \\
& h_2\cos\psi x(s)^2\dot{x}(s)^2y(s)^2 - h_2\sin\psi x(s)^2\dot{x}(s)^2y(s) + \\
& (h_4\cos\psi - h_1\cos\psi - h_2\sin\psi)x(s)\dot{x}(s)y(s)^2\dot{y}(s) + \\
& (h_1\sin\psi - 2h_2\cos\psi - h_4\sin\psi)x(s)\dot{x}(s)y(s)\dot{y}(s) + \\
& h_2\sin\psi x(s)\dot{x}(s)\dot{y}(s) + (h_1\sin\psi - h_3\cos\psi)y(s)^2\dot{y}(s)^2 + \\
& (h_1\cos\psi + h_2\sin\psi + h_3\sin\psi - h_4\cos\psi)y(s)\dot{y}(s)^2 + \\
& (h_2\cos\psi + h_4\sin\psi)\dot{y}(s)^2, \qquad (4.19)
\end{aligned}$$

which holds for all values of $x(s)$, $\dot{x}(s)$, $y(s)$, $\dot{y}(s)$ and ψ. Equation (4.19) yields the following 8 constraints

$$h_2 \cos\psi = 0, \tag{4.20}$$

$$h_2 \sin\psi = 0, \tag{4.21}$$

$$h_1 \cos\psi + h_2 \sin\psi - h_4 \cos\psi = 0, \tag{4.22}$$

$$2h_2 \cos\psi + h_4 \sin\psi = 0, \tag{4.23}$$

$$h_1 \sin\psi - 2h_2 \cos\psi - h_4 \sin\psi = 0, \tag{4.24}$$

$$h_1 \sin\psi - h_3 \cos\psi = 0, \tag{4.25}$$

$$h_1 \cos\psi + h_2 \sin\psi + h_3 \sin\psi - h_4 \cos\psi = 0, \text{ and} \tag{4.26}$$

$$h_2 \cos\psi + h_4 \sin\psi = 0. \tag{4.27}$$

Solving equations (4.20)–(4.27) gives

$$h_1 = h_2 = h_3 = h_4 = 0, \tag{4.28}$$

which makes $\mathbf{H}_{\mathrm{SOR}}$ singular. As a result, the ambiguity in the reconstruction cannot be described by a projective transformation.

4.6 Algorithm and Implementation

4.6.1 Estimation of the Harmonic Homology W

The harmonic homology associated with the silhouette of a surface of revolution can be estimated using an algorithm similar to the one described in Section 3.6.1. Given the camera calibration matrix $\mathbf{K}$, the harmonic homology is completely defined by the axis $\mathbf{l}_s$, as the center is then given by $\mathbf{K}\mathbf{K}^{\mathrm{T}}\mathbf{l}_s$ (see Section 3.5). The silhouette ρ of a surface of revolution is first extracted from the image by applying

a Canny edge detector [16], and the harmonic homology $\mathbf{W}$ that maps each side of ρ to its symmetric counterpart is then estimated by minimizing the geometric distances between the original silhouette ρ and its transformed version $\rho' = \mathbf{W}\rho$ (see figure 4.2). This can be done by sampling N evenly spaced points $\mathbf{x}_i$ along ρ and optimizing the cost function

$$\mathrm{Cost}_{\mathbf{W_K}}(\mathbf{l}_\mathrm{s}) = \sqrt{\frac{1}{N}\sum_{i=1}^{N} \mathrm{dist}(\mathbf{W}(\mathbf{K}\mathbf{K}^\mathrm{T}\mathbf{l}_\mathrm{s}, \mathbf{l}_\mathrm{s})\mathbf{x}_i, \rho)^2}, \tag{4.29}$$

where $\mathrm{dist}(\mathbf{W}(\mathbf{K}\mathbf{K}^\mathrm{T}\mathbf{l}_\mathrm{s}, \mathbf{l}_\mathrm{s})\mathbf{x}_i, \rho)$ is the orthogonal distance from the transformed sample point $\mathbf{x}'_i = \mathbf{W}(\mathbf{K}\mathbf{K}^\mathrm{T}\mathbf{l}_\mathrm{s}, \mathbf{l}_\mathrm{s})\mathbf{x}_i$ to the original silhouette ρ, and

$$\mathbf{W}(\mathbf{K}\mathbf{K}^\mathrm{T}\mathbf{l}_\mathrm{s}, \mathbf{l}_\mathrm{s}) = \mathbb{I}_3 - 2\frac{\mathbf{K}\mathbf{K}^\mathrm{T}\mathbf{l}_\mathrm{s}\mathbf{l}_\mathrm{s}^\mathrm{T}}{\mathbf{l}_\mathrm{s}^\mathrm{T}\mathbf{K}\mathbf{K}^\mathrm{T}\mathbf{l}_\mathrm{s}} \tag{4.30}$$

is the harmonic homology defined by the camera calibration matrix $\mathbf{K}$ and the axis $\mathbf{l}_\mathrm{s}$. The axis $\mathbf{l}_\mathrm{s}$ can be initialized manually by observing the symmetry in the silhouette. Alternatively, $\mathbf{l}_\mathrm{s}$ can be initialized by using bitangents to the silhouette, as described in Section 3.6.1.

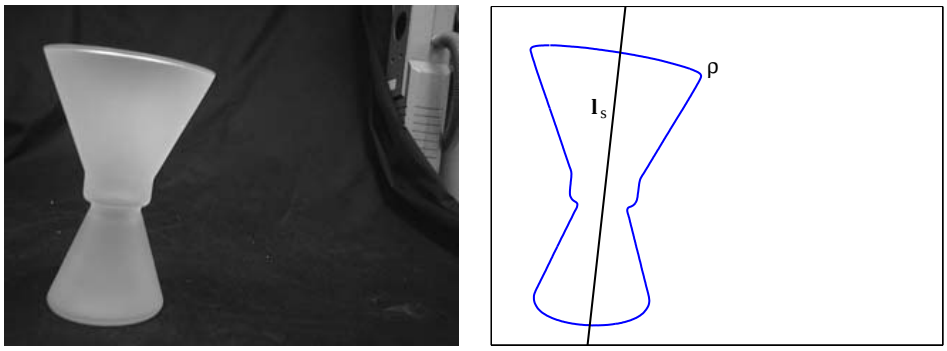


Figure 4.2: The silhouette ρ of a surface of revolution (candle holder) is extracted by applying a Canny edge detector and the axis $\mathbf{l}_\mathrm{s}$ of the harmonic homology associated with the silhouette is estimated.

4.6.2 Image Rectification

After the estimation of the harmonic homology $\mathbf{W}$, the image can be rectified so that the silhouette becomes bilaterally symmetric about the line $\mathbf{l} = [1\ 0\ 0]^{\mathrm{T}}$. Such a rectified image resembles an image that would have been observed by a normalized camera when the axis of the surface of revolution lies on the y-z plane of the camera coordinate system. The image is first normalized by $\mathbf{K}^{-1}$ to remove the effects of the intrinsic parameters of the camera. The axis $\mathbf{l}_{\mathrm{s}}$ of $\mathbf{W}$, and hence the image of the revolution axis, is transformed to

$$\mathbf{l}_{\mathrm{s}}^{\mathrm{n}} = \mathbf{K}^{\mathrm{T}}\mathbf{l}_{\mathrm{s}}. \tag{4.31}$$

The normalized image is then transformed by $\mathbf{R}_{\mathrm{b}}$ which is a rotation matrix that brings $\mathbf{x}_0^{\mathrm{p}}$, the orthogonal projection of the principal point $\mathbf{x}_0 = [0\ 0\ 1]^{\mathrm{T}}$ on the axis $\mathbf{l}_{\mathrm{s}}^{\mathrm{n}}$, to $\mathbf{x}_0$. This corresponds to rotating the normalized camera until it points directly towards the axis of the surface of revolution, and the resulting silhouette will then be bilaterally symmetric about the image of the revolution axis. The axis $\mathbf{n}_{\mathrm{b}}$ and the angle ϕ_{b} of the rotation $\mathbf{R}_{\mathrm{b}}$ are given by

$$\mathbf{n}_{\mathrm{b}} = \frac{\mathbf{x}_0^{\mathrm{p}} \times \mathbf{x}_0}{|\mathbf{x}_0^{\mathrm{p}} \times \mathbf{x}_0|}, \text{ and} \tag{4.32}$$

$$\phi_{\mathrm{b}} = \arccos\left(\frac{\mathbf{x}_0^{\mathrm{p}} \cdot \mathbf{x}_0}{|\mathbf{x}_0^{\mathrm{p}}|\,|\mathbf{x}_0|}\right). \tag{4.33}$$

After transforming the normalized image by the homography $\mathbf{R}_{\mathrm{b}}$, the resulting silhouette $\rho^{\mathrm{b}} = \mathbf{R}_{\mathrm{b}}\mathbf{K}^{-1}\rho$ will be bilaterally symmetric about the transformed image of the revolution axis, given by

$$\begin{aligned} \mathbf{l}_{\mathrm{s}}^{\mathrm{b}} &= \mathbf{R}_{\mathrm{b}}^{-\mathrm{T}}\mathbf{l}_{\mathrm{s}}^{\mathrm{n}} \\ &= \begin{bmatrix} \cos\theta^{\mathrm{b}} \\ \sin\theta^{\mathrm{b}} \\ 0 \end{bmatrix}. \end{aligned} \tag{4.34}$$

The resulting image is then rotated about the point $\mathbf{x}_0$ until the axis of symmetry aligns with the y-axis, and the transformation is given by

$$\mathbf{R}_{\mathrm{a}} = \begin{bmatrix} \cos\theta^{\mathrm{b}} & \sin\theta^{\mathrm{b}} & 0 \\ -\sin\theta^{\mathrm{b}} & \cos\theta^{\mathrm{b}} & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{4.35}$$

This corresponds to rotating the normalized camera, which is now pointing directly towards the axis of the surface of revolution, about its z-axis until the axis of the surface of revolution lies on the y-z plane. The resulting silhouette $\rho^{\mathrm{a}} = \mathbf{R}_{\mathrm{a}}\rho^{\mathrm{b}}$ is now bilaterally symmetric about the line

$$\begin{aligned} \mathbf{l}_{\mathrm{s}}^{\mathrm{a}} &= \mathbf{R}_{\mathrm{a}}^{-\mathrm{T}}\mathbf{l}_{\mathrm{s}}^{\mathrm{b}} \\ &= \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \end{aligned} \tag{4.36}$$

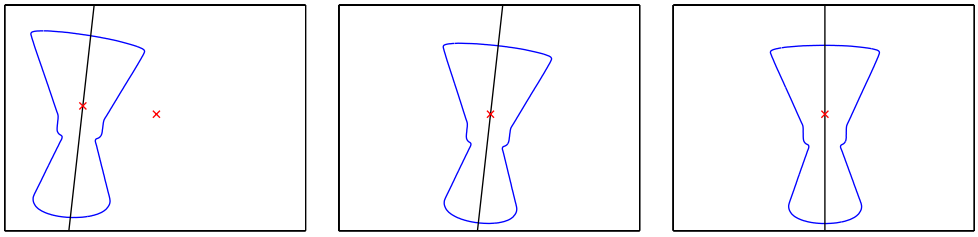
and hence is invariant to the harmonic homology $\mathbf{T}$ (see Section 3.4). The overall transformation for the rectification is given by

$$\mathbf{H}_{\mathrm{rectify}} = \mathbf{R}_{\mathrm{a}}\mathbf{R}_{\mathrm{b}}\mathbf{K}^{-1}, \tag{4.37}$$

and the rectification process is illustrated in figure 4.3.

4.6.3 Depth Recovery

Since the rectified silhouette ρ^{a} is bilaterally symmetric about the y-axis, only one side of ρ^{a} needs to be considered during the reconstruction of the surface of revolution. Points are first sampled from one side of ρ^{a} and the tangent vector (i.e. $\dot{x}(s)$ and $\dot{y}(s)$) at each sample point is estimated by fitting a polynomial to the neighboring points in the rectified silhouette. The surface normal associated with each sample point is then computed from equation (4.6). Finally, the depth of each sample point is recovered from equation (4.9), and the contour generator



(a) (b) (c)

Figure 4.3: (a) The harmonic homology associated with the silhouette of the surface of revolution is estimated, which yields the image of the revolution axis. The image is then normalized by $\mathbf{K}^{-1}$, and the orthogonal projection $\mathbf{x}_0^{\mathrm{p}}$ of the point $\mathbf{x}_0 = [0\ 0\ 1]^{\mathrm{T}}$ on the image of the revolution axis is located. (b) The image is transformed by the homography $\mathbf{R}_{\mathrm{b}}$ so that the point $\mathbf{x}_0$ lies on the image of the revolution axis and the silhouette becomes bilaterally symmetric about the image of the revolution axis. (c) Finally, the image is rotated about the point $\mathbf{x}_0$ until the image of the revolution axis aligns with the y-axis.

and the surface of revolution follow. For $\psi \neq 0$, the viewing vector $\mathbf{p}(s)$ and the associated surface normal $\mathbf{n}(s)$ at each sample point are first transformed by $\mathbf{R}_x(\psi)$. The transformed viewing vector is then normalized so that its 3^{rd} coefficient becomes 1, and equation (4.9) can then be used to recover the depth of the sample point.

4.7 Experiments and Results

Figure 4.4 shows an image of a candle holder and its rectified silhouette. The rectification of the silhouette was done using the algorithm described in Section 4.6. An ellipse was fitted to the bottom of the rectified silhouette for computing the orientation of the revolution axis relative to the y-axis of the camera coordinate system (the ambiguity in solution was resolved manually, see Appendix D), and the angle ψ_0 was estimated to be -5.2924°. In order to illustrate the ambiguity in the reconstruction, 10 surfaces of revolution were reconstructed from the rec-

tified silhouette with $\psi = -20^\circ$, -15°, -10°, -5°, 0°, 5°, 10°, 15°, 20° and ψ_0 respectively. The reconstructed 3D models of the candle holder are shown in figure 4.6, together with their corresponding curves of revolution that generated the surfaces. For the sake of easy comparison, all the estimated surfaces of revolution were scaled to have unit heights. From figure 4.6, it can be seen that as ψ increased, the curve of revolution expanded towards the top and shrank towards the bottom. This can be explained by the fact that as ψ increases, the bottom of the surface is assumed to be tilted more towards the camera before the application of the rotation $\mathbf{R}_x(\psi)$. As a result, the surface needs to expand towards the top and shrink towards the bottom to give the same silhouette in the rectified image under perspective projection. The radius of the topmost circle and the height of the candle holder, measured manually using a ruler with a resolution of 1mm, were 5.7cm and 17.1cm respectively. The ratio of the radius of the topmost circle to the height of the reconstructed candle holder, with $\psi = \psi_0$, was 0.3433. This ratio agreed with the ground truth value (5.7/17.1 = 0.3333) and had a relative error of 3% only.

Another example is given in figure 4.5, which shows an image of a bowl and its rectified silhouette. An ellipse was fitted to the top of the rectified silhouette and the angle ψ_0 was estimated to be 2.7192°. The reconstructed 3D models of the bowl and their corresponding curves of revolution that generated the surfaces are shown in figure 4.7. The radius of the topmost circle and the height of the bowl, measured manually using a ruler with a resolution of 1mm, were 6.4cm and 6.2cm respectively. The ratio of the radius of the topmost circle to the height of the reconstructed bowl, with $\psi = \psi_0$, was 1.0995. This ratio was close to the ground truth value (6.4/6.2 = 1.0323) and had a relative error of 6.5%.

Figure 4.4: Image of a candle holder and its rectified silhouette which exhibits bilateral symmetry.

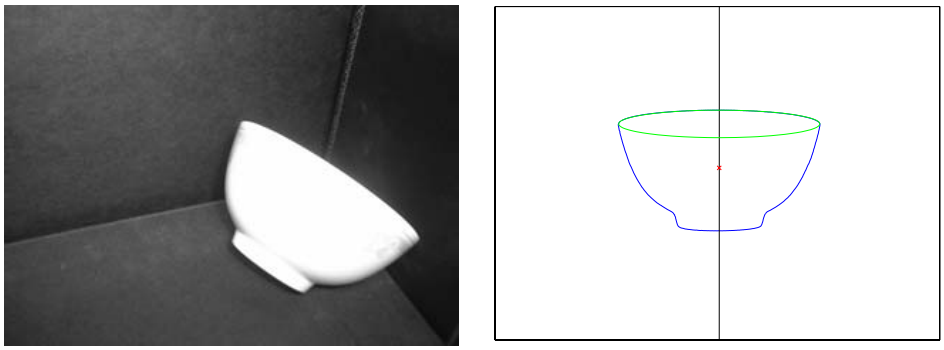


Figure 4.5: Image of a bowl and its rectified silhouette which exhibits bilateral symmetry.

4.8 Discussions

By exploiting the coplanarity constraint between the axis of revolution and the surface normal, a simple technique for recovering the 3D shape of a surface of revolution from a single view has been developed. The technique presented here assumes perspective projection and uses information from the silhouette only. The invariant property of the surface of revolution and its silhouette has been used to rectify the image so that the silhouette becomes bilaterally symmetric about the

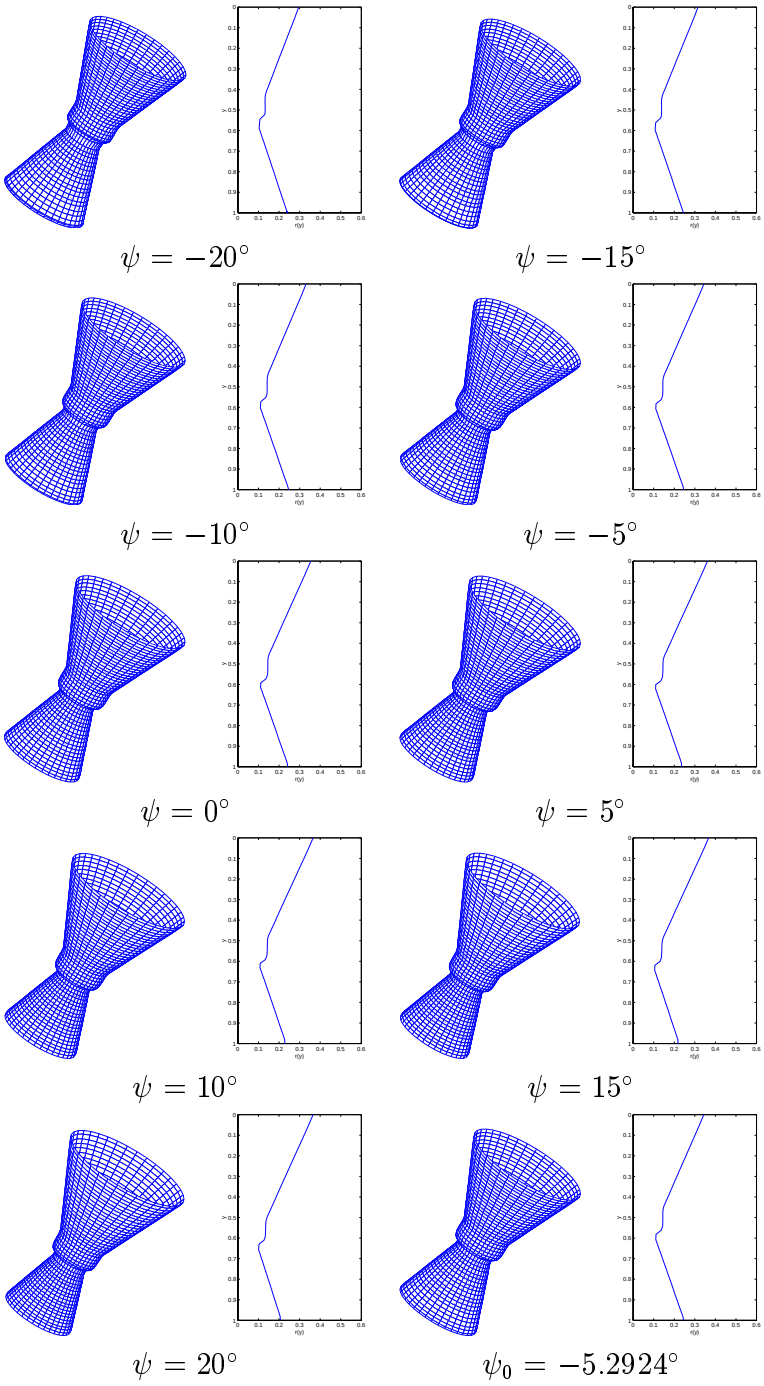


Figure 4.6: 3D models of the candle holder estimated from a single view and the corresponding curves of revolution.

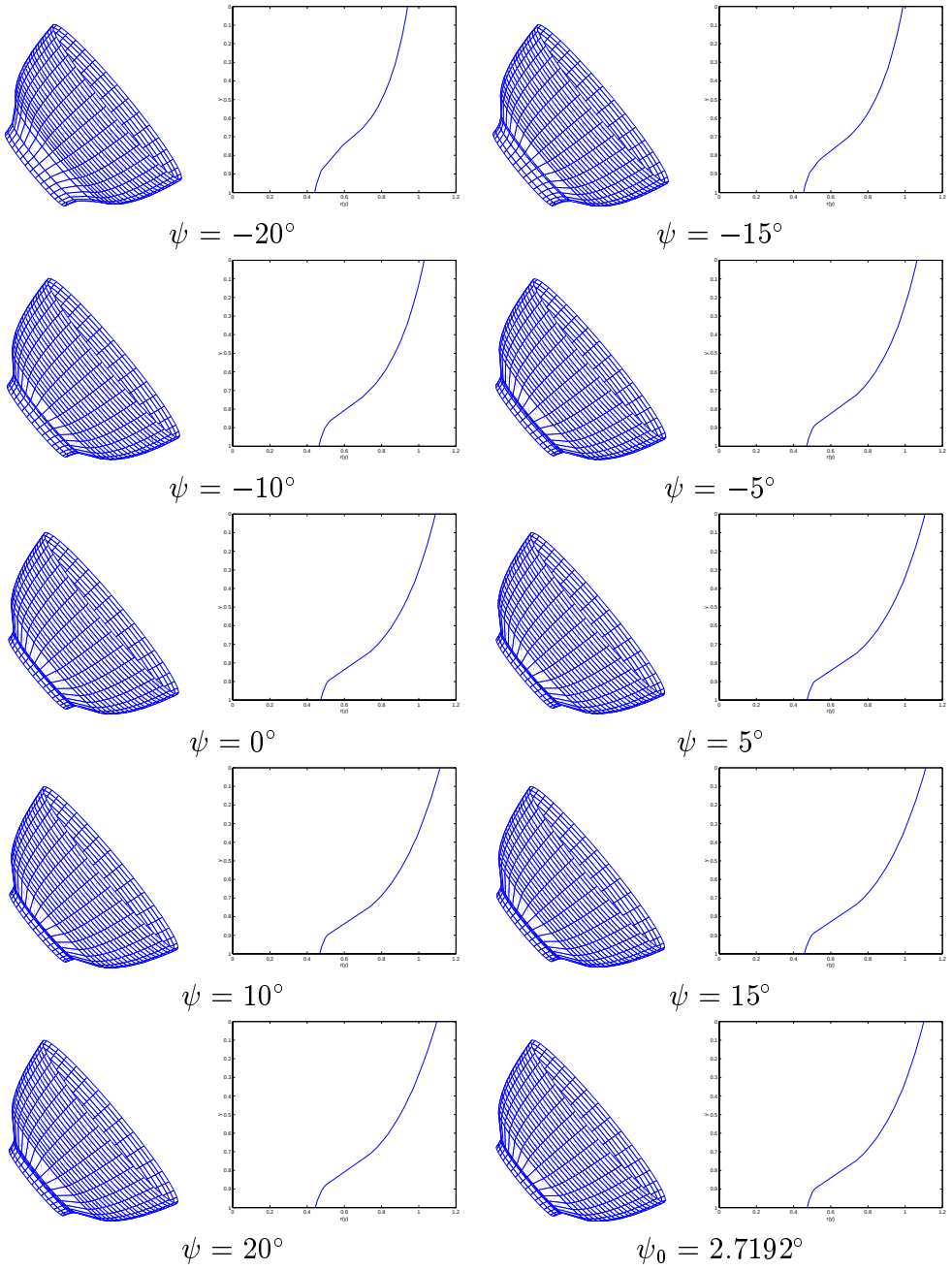


Figure 4.7: 3D models of the bowl estimated from a single view and the corresponding curves of revolution.

y-axis. This simplifies the analysis of the general camera configuration case to one in which the axis of revolution lies on the y-z plane of the camera coordinate system. The 1-parameter ambiguity in the reconstruction under general camera configuration, which cannot be described by a projection transformation, corresponds to the ambiguity of the orientation of the revolution axis on the y-z plane of the camera coordinate system. If the image of a latitude circle in the surface of revolution can be localized, the orientation of the revolution axis relative to the y-axis of the camera coordinate system can be estimated, which removes the ambiguity in the reconstruction. Alternatively, the ambiguity can also be removed by knowing the ratio of the radius of any latitude circle in the surface of revolution to the height of the surface of revolution. It is worth mentioning that sometimes due to self-occlusions, it might not be always possible to recover the whole surface of revolution from its silhouette. This situation is illustrated in figure 4.8, where part of the neck and the bottom of the vase cannot be reconstructed.

(a) (b)

Figure 4.8: Due to self-occlusions, it might not be always possible to recover the whole surface of revolution from its silhouette. (a) It is possible to recover the whole surface from the side view of a vase. (b) Part of the neck and the bottom of the vase cannot be reconstructed from this top view due to self-occlusions.

Chapter 5

Motion Estimation from Silhouettes

"Push on,—keep moving.."

- Thomas Morton, *A Cure for the Heartache*, Act ii, Sc. 1.

5.1 Introduction

Silhouettes are often a dominant image feature, and can be extracted relatively easily and reliably. They provide rich information about both the shape and motion of an object, and are indeed the only information available in the case of smooth textureless surfaces. Nonetheless, structure and motion from silhouettes has always been a challenging problem [68, 24, 22, 123, 125, 4, 30, 64]. Unlike corners, silhouettes are projections of contour generators which are viewpoint dependent, and hence they do not readily provide point correspondences (see Section 2.4). As a result, classical techniques for motion estimation and scene reconstruction [126, 9, 67, 44], based on point correspondences in the image sequence, cannot be applied.

In this chapter, a complete and practical system for generating high quality 3D models from 2D silhouettes is introduced. The system presented here employs a

novel technique [95, 96] for estimating the motion of an object undergoing circular motion from its silhouettes alone. An initial 3D model of the object can be obtained by an octree carving technique [124] using the silhouettes and the estimated motion. The system then allows the 3D model thus obtained to be refined incrementally by adding new arbitrary general views of the object and estimating the corresponding camera motion. This is achieved by registering the silhouette in the new view with the set of silhouettes generated by the now estimated circular motion [138]. The incorporation of arbitrary general views reveals information which is concealed under circular motion, and overcomes the drawbacks of using circular motion alone. Only the 2 outer epipolar tangents to the silhouettes are required in estimating both the circular and general motion, and no corner detection nor matching is needed. The system described is practical in almost all situations, and is capable of reconstructing virtually any kind of objects.

This chapter will concentrate on the problem of motion estimation from silhouettes, whereas the problem of model reconstruction from silhouettes will be studied in Chapter 6. Section 5.2 reviews existing techniques in the literature for motion estimation from silhouettes, and discusses their shortcomings. Section 5.3 studies the epipolar constraint between silhouettes from distinct viewpoints and introduces the use of outer epipolar tangents that simplifies the correspondence problem. Section 5.4 addresses the problem of estimating the motion of a rotating object and presents 2 useful parameterizations of the fundamental matrix specific to circular motion. The general motion case is then tackled in Section 5.5. The algorithms and implementations are described in Section 5.6, followed by a discussion of the degenerate case for the estimation of circular motion in Section 5.7. Results of real data experiments, demonstrating the practicality of the system, are

presented in Section 5.8. Finally discussions are given in Section 5.9.

5.2 Previous Works

The study of motion estimation from silhouettes was pioneered by Rieger [111], who showed that camera motion can be recovered from 3 *fixed points* of a deforming silhouette under orthographic projection. He also set forth the idea that under perspective projection, the epipole is constrained to the line spanned by the tangent vector to the silhouette at the fixed point (i.e. epipolar tangency constraint). In [107], it was noted that the intersection of 2 contour generators from 2 distinct viewpoints generates a point that is visible in both images as a fixed point. This point was identified as a *frontier point* in [45], where Giblin et al. developed an algorithm for motion estimation from the silhouettes of a rotating surface under orthographic projection.

The methods mentioned so far deal with the motion recovery problem under orthographic projection, which is a rather restrictive situation. The use of frontier points and epipolar tangents for motion recovery under perspective projection was introduced in [22, 4]. These techniques require the presence of at least 7 pairs of corresponding epipolar tangents in the image pair, which are localized by iterative methods. By using an affine approximation [100, 118, 110], a similar technique that only requires 4 pairs of corresponding epipolar tangents was developed in [93]. In [116], a non-iterative method was presented in the case of linear camera motion, where common tangents are used to determine both the frontier points and the epipoles. By combining the ideas in [4] and [116], Cross et al. [30] implemented a parallax-based technique in which images are registered using a

reference plane to "undo" the effect of rotation. Related work also includes [64] in which a calibrated trinocular stereo rig with known geometry was used.

This chapter tackles the problem of structure and motion from silhouettes observed under perspective projection using a single camera. The approach here is to first constrain the motion to be circular. This allows a trivial initialization of the parameters which all bear physical meanings (e.g. image of the rotation axis, the horizon and the angles of rotation). When there are 3 or more images in the circular motion sequence, a solution is possible by using only the 2 outer epipolar tangents to the silhouettes. In the case of complete circular motion with dense image sequence, the image of the rotation axis can be estimated conveniently and independently by exploiting the symmetry [147, 33, 45] associated with the image of the surface of revolution swept by the rotating object. The drawbacks of using circular motion alone are then overcome by incorporating new views from arbitrary general motion. The initialization of the general motion can be done relatively easily by using the model built from the estimated circular motion. By registering the silhouette in the new view with the set of silhouettes resulted from the circular motion, the camera motion can again be estimated using only the 2 outer epipolar tangents.

5.3 Epipolar Constraint between Silhouettes

Silhouettes are projections of contour generators which are viewpoint dependent, and hence they do not readily provide point correspondences. A frontier point is given by the intersection of 2 contour generators from 2 distinct viewpoints, and is visible in both images. A frontier point lies on an epipolar plane tangent to the

surface, and hence it will be projected onto a point in the silhouette which is also an epipolar tangent point (see Section 2.4). Epipolar tangent points thus provide point correspondences that satisfy the epipolar constraint, and can be exploited for motion estimation.

Theoretically, if 7 or more pairs of corresponding epipolar tangent points are available, the epipolar geometry between the 2 views can be estimated, and the camera intrinsic parameters can then be used to recover the relative motion [78, 39]. However, when the epipolar geometry is not known, the localization of the epipolar tangents involves a nonlinear optimization. Examples of this iterative approach can be found in [22, 4]. The need for a good but nontrivial initialization, the unrealistic demand for a large number of epipolar tangent points, and the presence of local minima all make this approach impractical.

In Section 5.4 and Section 5.5, 2 motion estimation algorithms which only require the 2 *outer epipolar tangents* are presented. The outer epipolar tangents correspond to the 2 epipolar tangent planes which touch the object (see figure 5.1). Except when the baseline passes through the object, the 2 outer epipolar tangents are always available in any pair of views and are guaranteed to be in correspondence. The use of the outer epipolar tangents avoids false matches due to self-occlusions and greatly simplifies the matching problem. This is illustrated in figure 5.2 which shows 2 silhouettes from 2 distinct viewpoints. The silhouette in the left image has 11 epipolar tangents, whereas the silhouette in the right image has only 6 epipolar tangents. A careful examination will show that not all 6 epipolar tangents in the right image have a correspondence in the left image. There are actually only 4 pairs of corresponding epipolar tangents, which are the 2 outer epipolar tangents and another 2 tangents at the front and back left legs. By con-

sidering only the outer epipolar tangents, possible false matches are eliminated and the problem is reduced to matching only 2 epipolar tangents against another 2, leaving only 2 possible cases.

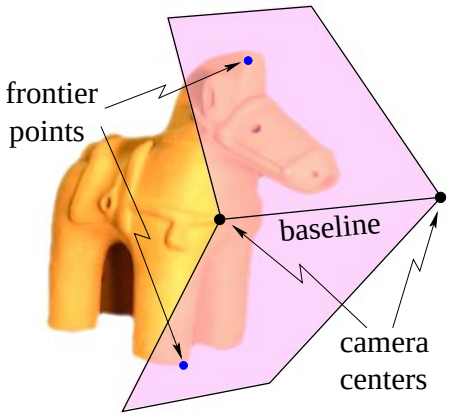


Figure 5.1: The outer epipolar tangents correspond to the 2 epipolar tangent planes which touch the object, and are always available in any pair of views except when the baseline passes through the object.

5.4 Circular Motion

5.4.1 Fixed Image Features under Circular Motion

Consider a pin-hole camera rotating about a fixed axis. Let $\mathbf{v}_x$ be the vanishing point corresponding to the normal direction $\mathbf{N}_x$ of the plane Π_s that contains the axis of rotation and the camera center, and $\mathbf{l}_\mathrm{h}$ be the *horizon* which is the image of the plane Π_h that contains the trajectory of the camera center. By definition, the epipoles are the projections of the camera center and must therefore lie on $\mathbf{l}_\mathrm{h}$. Besides, since $\mathbf{N}_x$ is parallel to the plane Π_h, it follows that $\mathbf{v}_x$ also lies on $\mathbf{l}_\mathrm{h}$. The plane Π_s will be projected onto the image plane as a line $\mathbf{l}_\mathrm{s}$, which is also the

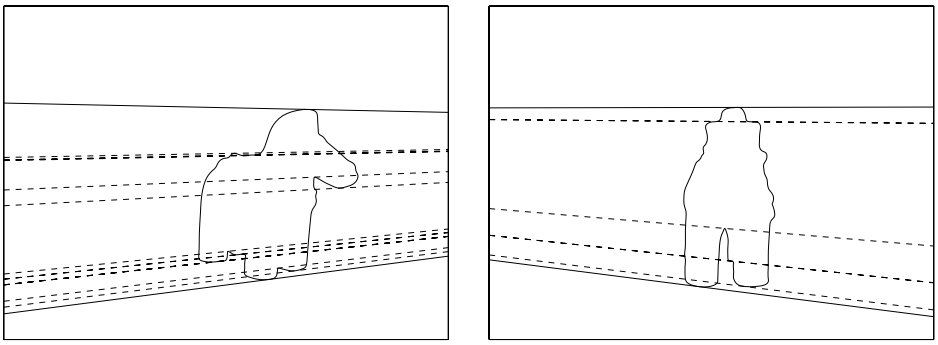


Figure 5.2: Two discrete views showing 17 epipolar tangents in total, of which only 4 pairs are in correspondence. The use of the 2 outer epipolar tangents (in solid lines), which are guaranteed to be in correspondence, avoids false matches due to self-occlusions, and greatly simplifies the matching problem.

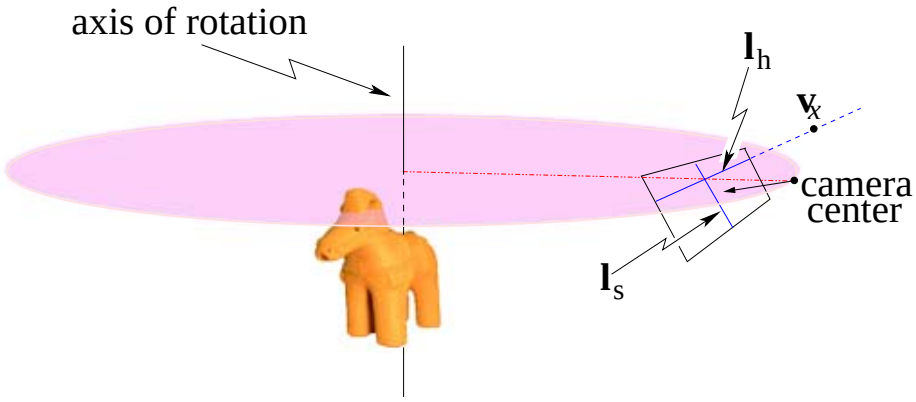


Figure 5.3: If the intrinsic parameters of the camera are assumed to be fixed, the image of the rotation axis $\mathbf{l}_s$, the horizon $\mathbf{l}_h$ and the vanishing point $\mathbf{v}_x$ corresponding to the normal direction of the plane that contains the rotation axis and the camera center, will be fixed throughout the image sequence.

image of the rotation axis. It has been shown in Section 3.5 that $\mathbf{l}_{\mathrm{s}}$ is related to $\mathbf{v}_x$ by the camera calibration matrix $\mathbf{K}$, given by

$$\mathbf{v}_x = \mathbf{K}\mathbf{K}^{\mathrm{T}}\mathbf{l}_{\mathrm{s}}. \tag{5.1}$$

If the intrinsic parameters of the camera are assumed to be fixed, due to symmetry in the configuration, $\mathbf{l}_{\mathrm{s}}$, $\mathbf{l}_{\mathrm{h}}$ and $\mathbf{v}_x$ will be fixed throughout the image sequence (see figure 5.3). The fundamental matrix associated with any pair of views in the circular motion sequence can be parameterized explicitly in terms of these fixed features [135, 44], and a simple derivation of this parameterization is given in the next section.

5.4.2 Parameterizations of the Fundamental Matrix

Parameterization via Fixed Image Features of Circular Motion

Consider 2 pin-hole cameras $\hat{\mathbf{P}}_1$ and $\hat{\mathbf{P}}_2$, given by

$$\hat{\mathbf{P}}_1 = [\mathbb{I}_3\ \mathbf{t}],\ \text{and} \tag{5.2}$$

$$\hat{\mathbf{P}}_2 = [\mathbf{R}_y(\theta)\ \mathbf{t}], \tag{5.3}$$

where $\mathbf{t} = [0\ 0\ 1]^{\mathrm{T}}$ and $\mathbf{R}_y(\theta)$ is a rotation by an angle $\theta \neq 0$ about the y-axis, given by

$$\mathbf{R}_y(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix}. \tag{5.4}$$

Under this camera configuration, the image of the rotation axis, the horizon and the special vanishing point are given by

$$\hat{\mathbf{l}}_{\mathrm{s}} = [1\ 0\ 0]^{\mathrm{T}}, \tag{5.5}$$

$$\hat{\mathbf{l}}_{\mathrm{h}} = [0\ 1\ 0]^{\mathrm{T}},\ \text{and} \tag{5.6}$$

$$\hat{\mathbf{v}}_x = [1\ 0\ 0]^{\mathrm{T}} \tag{5.7}$$

respectively. By substituting $\hat{\mathbf{P}}_1$ and $\hat{\mathbf{P}}_2$ into (2.15), the fundamental matrix associated with $\hat{\mathbf{P}}_1$ and $\hat{\mathbf{P}}_2$ is given by

$$\begin{aligned}\hat{\mathbf{F}} &= \begin{bmatrix} 0 & \tan\frac{\theta}{2} & 0 \\ \tan\frac{\theta}{2} & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} \\ &= \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}_{\times} + \tan\frac{\theta}{2}\left(\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} [0\ 1\ 0] + \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} [1\ 0\ 0] \right). \end{aligned} \tag{5.8}$$

By substituting (5.5), (5.6) and (5.7) into (5.8), the fundamental matrix can be rewritten in terms of the fixed image features under circular motion, and is given by [94, 96]

$$\hat{\mathbf{F}} = [\hat{\mathbf{v}}_x]_{\times} + \tan\frac{\theta}{2}(\hat{\mathbf{l}}_{\mathrm{s}}\hat{\mathbf{l}}_{\mathrm{h}}^{\mathrm{T}} + \hat{\mathbf{l}}_{\mathrm{h}}\hat{\mathbf{l}}_{\mathrm{s}}^{\mathrm{T}}). \tag{5.9}$$

Consider now a pair of camera $\mathbf{P}_1$ and $\mathbf{P}_2$ obtained by introducing the intrinsic parameters represented by the camera calibration matrix $\mathbf{K}$ to $\hat{\mathbf{P}}_1$ and $\hat{\mathbf{P}}_2$ respectively, and by applying the rotation $\mathbf{R}$ to $\hat{\mathbf{P}}_1$ and $\hat{\mathbf{P}}_2$ about their optical centers respectively. Hence $\mathbf{P}_1 = \mathbf{H}\hat{\mathbf{P}}_1$ and $\mathbf{P}_2 = \mathbf{H}\hat{\mathbf{P}}_2$, where $\mathbf{H} = \mathbf{K}\mathbf{R}$. The fundamental matrix associated with $\mathbf{P}_1$ and $\mathbf{P}_2$ is then given by [94, 96]

$$\begin{aligned}\mathbf{F} &= \mathbf{H}^{-\mathrm{T}}\hat{\mathbf{F}}\mathbf{H}^{-1} \\ &= \frac{1}{\det(\mathbf{H})}[\mathbf{v}_x]_{\times} + \tan\frac{\theta}{2}(\mathbf{l}_{\mathrm{s}}\mathbf{l}_{\mathrm{h}}^{\mathrm{T}} + \mathbf{l}_{\mathrm{h}}\mathbf{l}_{\mathrm{s}}^{\mathrm{T}}) \\ &= \frac{1}{\det(\mathbf{K})}[\mathbf{v}_x]_{\times} + \tan\frac{\theta}{2}(\mathbf{l}_{\mathrm{s}}\mathbf{l}_{\mathrm{h}}^{\mathrm{T}} + \mathbf{l}_{\mathrm{h}}\mathbf{l}_{\mathrm{s}}^{\mathrm{T}}), \end{aligned} \tag{5.10}$$

where $\mathbf{l}_{\mathrm{s}} = \mathbf{H}^{-\mathrm{T}}\hat{\mathbf{l}}_{\mathrm{s}}$, $\mathbf{l}_{\mathrm{h}} = \mathbf{H}^{-\mathrm{T}}\hat{\mathbf{l}}_{\mathrm{h}}$ and $\mathbf{v}_x = \mathbf{H}\hat{\mathbf{v}}_x$. Note that $\mathbf{l}_{\mathrm{s}}$, $\mathbf{l}_{\mathrm{h}}$ and $\mathbf{v}_x$ are the image of the rotation axis, the horizon and the special vanishing point, respectively, under this new camera configuration.

Equation (5.10) gives a simple parameterization of the fundamental matrix relating any pair of views in the circular motion sequence. This parameterization

allows a trivial initialization of the parameters which all bear physical meanings, and greatly reduces the dimension of the search space for the optimization problem in motion estimation. When the intrinsic parameters of the camera are fixed and known, 2 parameters are enough to fix $\mathbf{l}_\mathrm{s}$ and $\mathbf{v}_x$. Since $\mathbf{v}_x$ must also lie on $\mathbf{l}_\mathrm{h}$, only 1 further parameter is needed to fix $\mathbf{l}_\mathrm{h}$. As a result, a sequence of N images taken under circular motion can be described by $N+2$ motion parameters (2 for $\mathbf{l}_\mathrm{s}$ and $\mathbf{v}_x$, 1 for $\mathbf{l}_\mathrm{h}$ and the $N-1$ rotation angles). By exploiting the 2 outer epipolar tangents, the N images will provide $2N$ (or 2 when $N=2$) independent constraints on these parameters, and a solution will be possible when $N \geq 3$. The algorithm for estimating these $N+2$ motion parameters is given in Section 5.6.3.

Parameterization via Harmonic Homology

Consider again the pair of cameras $\hat{\mathbf{P}}_1$ and $\hat{\mathbf{P}}_2$ given in equations (5.2) and (5.3). The epipoles can be obtained by projecting the camera center of $\hat{\mathbf{P}}_2$ into $\hat{\mathbf{P}}_1$ and vice versa, and are given by

$$\hat{\mathbf{e}}_1 = \begin{bmatrix} \sin\theta \\ 0 \\ -\cos\theta + 1 \end{bmatrix} \text{ and } \hat{\mathbf{e}}_2 = \begin{bmatrix} -\sin\theta \\ 0 \\ -\cos\theta + 1 \end{bmatrix}. \tag{5.11}$$

Equation (5.11) shows that the epipoles are related by the transformation

$$\begin{aligned} \hat{\mathbf{e}}_2 &= \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \hat{\mathbf{e}}_1 \\ &= \mathbf{T}\hat{\mathbf{e}}_1, \end{aligned} \tag{5.12}$$

where $\mathbf{T}$ is the harmonic homology with axis $\hat{\mathbf{l}}_\mathrm{s}$ and center $\hat{\mathbf{v}}_x$ (see Section 3.4). Consider now the pair of cameras $\mathbf{P}_1 = \mathbf{H}\hat{\mathbf{P}}_1$ and $\mathbf{P}_2 = \mathbf{H}\hat{\mathbf{P}}_2$, where $\mathbf{H} = \mathbf{K}\mathbf{R}$. The epipoles can be obtained by transforming $\hat{\mathbf{e}}_1$ and $\hat{\mathbf{e}}_2$ by $\mathbf{H}$ respectively, and

are given by

$$\mathbf{e}_1 = \mathbf{H}\hat{\mathbf{e}}_1, \text{ and} \tag{5.13}$$

$$\mathbf{e}_2 = \mathbf{H}\hat{\mathbf{e}}_2. \tag{5.14}$$

Substituting (5.13) and (5.14) into (5.12) gives

$$\begin{aligned} \mathbf{H}^{-1}\mathbf{e}_2 &= \mathbf{T}\mathbf{H}^{-1}\mathbf{e}_1 \\ \mathbf{e}_2 &= \mathbf{H}\mathbf{T}\mathbf{H}^{-1}\mathbf{e}_1 \\ &= \mathbf{W}\mathbf{e}_1, \end{aligned} \tag{5.15}$$

where $\mathbf{W}$ is the harmonic homology with axis $\mathbf{l}_s$ and center $\mathbf{v}_x$ (see Section 3.4). Note that $\mathbf{W}$ is the harmonic homology associated with the image of the surface of revolution swept by the rotating object. Given a dense image sequence taken under complete circular motion, say the angles of rotation are less than 10°, the image of this surface of revolution can be approximated by overlapping all the images in the sequence and the associated harmonic homology $\mathbf{W}$ can be estimated from the resulting image using the algorithm described in Section 3.6.1. Since $\mathbf{l}_s$ is a point-wise fixed feature in the image sequence and is invariant to $\mathbf{W}$, it follows from equation (5.15) that corresponding epipolar lines $\mathbf{l}_1$ and $\mathbf{l}_2$ are related by

$$\mathbf{l}_2 = \mathbf{W}^{-\mathrm{T}}\mathbf{l}_1 \text{ and} \tag{5.16}$$

$$\mathbf{l}_1 = \mathbf{W}^{\mathrm{T}}\mathbf{l}_2. \tag{5.17}$$

In Section 2.3.3, it has been shown that the fundamental matrix can be written in a plane plus parallax representation, given by

$$\mathbf{F} = [\mathbf{e}_2]_\times \mathbf{M}, \tag{5.18}$$

where $\mathbf{M}$ is any plane induced homography such that corresponding epipolar lines are mapped by $\mathbf{M}^{-\mathrm{T}}$ and $\mathbf{M}^{\mathrm{T}}$ respectively. Hence, from equations (5.16) and (5.17), it follows that the fundamental matrix can be parameterized as [94, 96]

$$\mathbf{F} = [\mathbf{e}_2]_\times \mathbf{W}. \tag{5.19}$$

Note that $\mathbf{W}$ is the homography induced by the plane $\Pi_{\mathbf{W}}$ that contains the axis of rotation and bisects the line segment joining the 2 camera centers [95, 96]. Consider any point $\mathbf{X}$ on the plane $\Pi_{\mathbf{W}}$, given by

$$\mathbf{X} = \begin{bmatrix} \mu \sin\frac{\theta}{2} \\ \nu \\ -\mu\cos\frac{\theta}{2} \\ 1 \end{bmatrix}, \tag{5.20}$$

where μ and ν are some real numbers. Its image in $\hat{\mathbf{P}}_2$ is given by

$$\begin{aligned} \hat{\mathbf{x}}_2 &= \hat{\mathbf{P}}_2\mathbf{X} \\ &= \mathbf{R}_y(\theta)\begin{bmatrix} \mu \sin\frac{\theta}{2} \\ \nu \\ -\mu\cos\frac{\theta}{2} \end{bmatrix} + \mathbf{t} \\ &= \begin{bmatrix} -\mu \sin\frac{\theta}{2} \\ \nu \\ -\mu\cos\frac{\theta}{2} + 1 \end{bmatrix} \\ &= \mathbf{T}\hat{\mathbf{P}}_1\mathbf{X} \\ &= \mathbf{T}\hat{\mathbf{x}}_1, \end{aligned} \tag{5.21}$$

where $\hat{\mathbf{x}}_1 = \hat{\mathbf{P}}_1\mathbf{X}$ is the image of $\mathbf{X}$ in $\hat{\mathbf{P}}_1$. The images of $\mathbf{X}$ in $\mathbf{P}_1$ and $\mathbf{P}_2$ can be obtained by transforming $\hat{\mathbf{x}}_1$ and $\hat{\mathbf{x}}_2$ by $\mathbf{H}$ respectively, and are given by

$$\mathbf{x}_1 = \mathbf{H}\hat{\mathbf{x}}_1, \text{ and} \tag{5.22}$$

$$\mathbf{x}_2 = \mathbf{H}\hat{\mathbf{x}}_2. \tag{5.23}$$

Substituting (5.22) and (5.23) into (5.21) gives

$$\begin{aligned} \mathbf{H}^{-1}\mathbf{x}_2 &= \mathbf{T}\mathbf{H}^{-1}\mathbf{x}_1 \\ \mathbf{x}_2 &= \mathbf{H}\mathbf{T}\mathbf{H}^{-1}\mathbf{x}_1 \\ &= \mathbf{W}\mathbf{x}_1. \end{aligned} \tag{5.24}$$

Equation (5.24) implies that the homology $\mathbf{W}$ is induced by the plane $\Pi_{\mathbf{W}}$ that contains the axis of rotation and bisects the line segment joining the 2 camera centers. The plane plus parallax parameterization given in equation (5.19) suggests that the harmonic homology $\mathbf{W}$ can be used to register the images and the parallax-based technique introduced in [4, 30] can be applied to estimate the camera motion by locating the epipoles using common tangents. However, the epipoles obtained in this way are not constrained to lie on the horizon $\mathbf{l}_{\mathrm{h}}$, and hence a full optimization using the parameterization given in equation (5.10) is necessary to refine the solution so that the resulting camera motion is constrained to be circular.

5.5 General Motion

Circular motion allows a trivial initialization of the motion parameters which all bear physical meanings, and can be estimated accurately using only the 2 outer epipolar tangents. However, new views cannot be added easily at a later time, and part of the structure will always remain invisible under circular motion. These limit the usefulness of circular motion in model building from silhouettes. The drawbacks of using circular motion alone are overcome by the incorporation of arbitrary general views. In this section, it is shown that circular motion can be exploited for the registration of any arbitrary general view using only the 2 outer

epipolar tangents, and that the initial 3D model built from the circular motion can be used to aid the initialization of the motion parameters for the general motion.

Circular motion will generate a *web of contour generators* around the object (see figure 5.4), which can be used for registering any new arbitrary general view. Given an arbitrary general view, the associated contour generator will intersect with this web and form frontier points. If the camera intrinsic parameters are known, the 6 motion parameters (3 for rotation and 3 for translation) of the new view can be determined when there are 6 or more frontier points on the associated contour generator. This corresponds to having a minimum of 3 views under circular motion, each providing 2 outer epipolar tangents to the silhouette in the new general view (see figure 5.5). The motion parameters of the arbitrary general view can then be estimated by minimizing the reprojection errors of the 2 outer epipolar tangents resulting from each view in the estimated circular motion sequence.

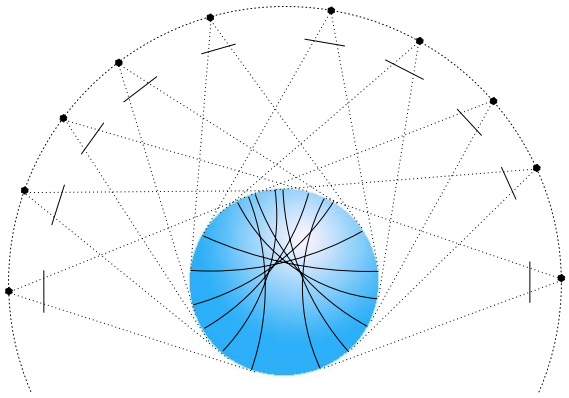


Figure 5.4: The circular motion will generate a web of contour generators around the object, which can be used for registering any new arbitrary general view.

The difficulty of nontrivial initialization, which exists in every algorithm for general motion estimation from silhouettes, is overcome by exploiting the 3D

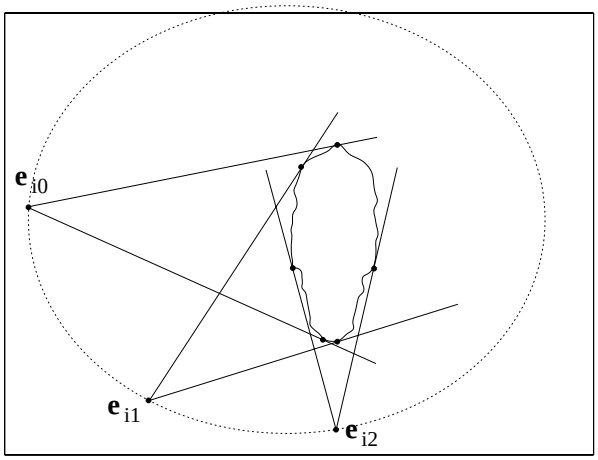


Figure 5.5: Three views from circular motion provide 6 outer epipolar tangents to the silhouette in the new general view for estimating its pose.

model built from the circular motion. After the estimation of the circular motion, a volumetric model of the object can be constructed by an octree carving technique [124] using the resulting camera configuration and the silhouettes. A triangulated mesh can then be extracted from the octree using the marching cubes algorithm [80]. The vertices in the mesh of the model are projected onto the new view whose pose is to be estimated. A very good initialization can be obtained by rotating and translating the camera (i.e. changing the 6 extrinsic parameters of the camera) until the projection of the initial 3D model roughly matches the silhouette in the new view (see figure 5.6).

5.6 Algorithms and Implementations

5.6.1 Extraction of Silhouettes

The cubic B-spline snake [23, 24] is chosen for the extraction of silhouettes from the image sequence. Cubic B-spline snake provides a compact representation for

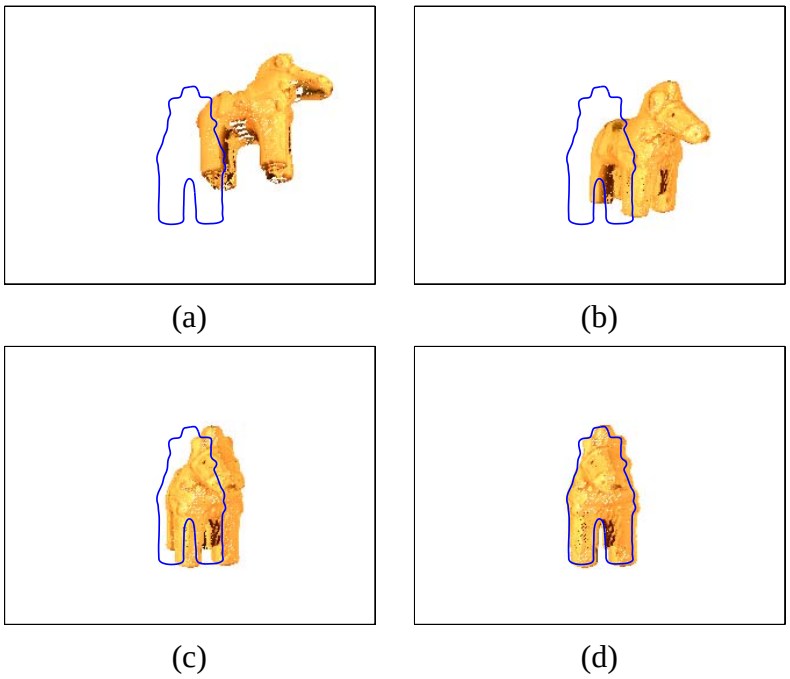


Figure 5.6: The arbitrary general motion can be initialized by rotating and translating the camera until the projection of the initial 3D model, built from the estimated circular motion, roughly matches the silhouette in the new view.

silhouettes of various complexity, and can achieve sub-pixel localization accuracy. Its parameterization also facilitates the localization of epipolar tangents.

The process of extracting a silhouette from an image using a B-spline snake is illustrated in figure 5.7. A B-spline snake is first initialized close to the target silhouette by selecting the control points manually. Points are then sampled along each spline segment and a search for intensity discontinuity (i.e. image edge) along the direction normal to the local tangent at each sample point is carried out. The control points of the B-Spline snake are then updated by a linear least-squares method, so that each sample point attaches to the location of intensity discontinuity found. In the implementation presented in this chapter, closed B-spline snake

is used to extract the complete silhouette of the object in the image.

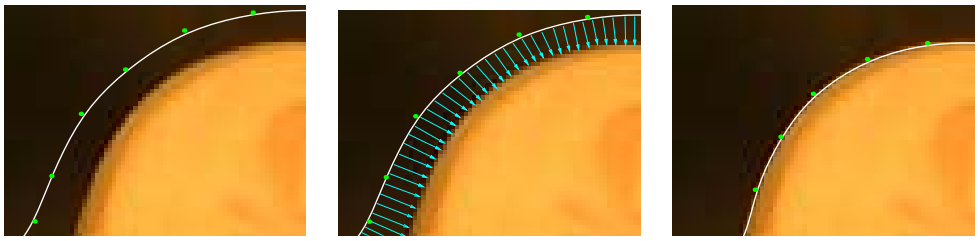


Figure 5.7: (a) A B-spline snake is initialized close to the target silhouette. (b) Points are sampled from each spline segment and a search for intensity discontinuity along the direction normal to the local tangent at each sample point is carried out. (c) The control points of the B-spline snake are updated so that each sample point attaches to the location of intensity discontinuity found.

5.6.2 Reprojection Errors of Epipolar Tangents

The motion estimation proceeds as an optimization which minimizes the reprojection errors of epipolar tangents. Given a pair of views i and j, the associated fundamental matrix $\mathbf{F}_{ij}$ is formed and the epipoles $\mathbf{e}_{ij}$ and $\mathbf{e}_{ji}$ are obtained from the right and left nullspaces of $\mathbf{F}_{ij}$ respectively. The outer epipolar tangent points $\mathbf{u}_{ij0}$, $\mathbf{u}_{ij1}$ and $\mathbf{u}_{ji0}$, $\mathbf{u}_{ji1}$ are located in view i and view j (see figure 5.8). The reprojection errors are then given by the geometric distances between the epipolar tangent points and their epipolar lines [81]

$$d_{ijk} = \frac{\mathbf{u}_{jik}^{\mathrm{T}}\mathbf{F}_{ij}\mathbf{u}_{ijk}}{\sqrt{(\mathbf{F}_{ij}^{\mathrm{T}}\mathbf{u}_{jik})_1^2 + (\mathbf{F}_{ij}^{\mathrm{T}}\mathbf{u}_{jik})_2^2}}, \text{ and} \tag{5.25}$$

$$d_{jik} = \frac{\mathbf{u}_{jik}^{\mathrm{T}}\mathbf{F}_{ij}\mathbf{u}_{ijk}}{\sqrt{(\mathbf{F}_{ij}\mathbf{u}_{ijk})_1^2 + (\mathbf{F}_{ij}\mathbf{u}_{ijk})_2^2}}, \tag{5.26}$$

where $(\mathbf{F}_{ij}^{\mathrm{T}}\mathbf{u}_{jik})_1$ and $(\mathbf{F}_{ij}^{\mathrm{T}}\mathbf{u}_{jik})_2$ indicate the 1^{st} and 2^{nd} coefficients of $(\mathbf{F}_{ij}^{\mathrm{T}}\mathbf{u}_{jik})$ respectively. Similarly, $(\mathbf{F}_{ij}\mathbf{u}_{ijk})_1$ and $(\mathbf{F}_{ij}\mathbf{u}_{ijk})_2$ indicate the 1^{st} and 2^{nd} coefficients of $(\mathbf{F}_{ij}\mathbf{u}_{ijk})$ respectively.

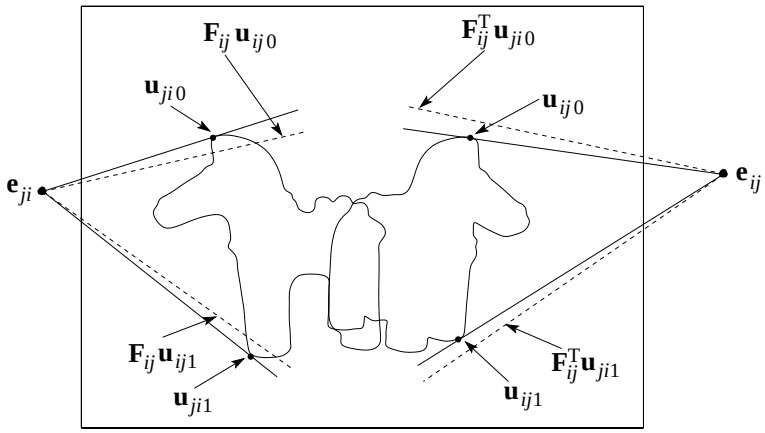


Figure 5.8: The motion parameters can be estimated by minimizing the reprojection errors of epipolar tangents, which are given by the geometric distances between the epipolar tangent points and their epipolar lines.

5.6.3 Estimation of the Circular Motion

For a sequence of N images taken under circular motion, the $N-1$ rotation angles are arbitrarily initialized. Usually by just inspecting the image sequence, a very good initialization for the image of the rotation axis $\mathbf{l}_{\mathrm{s}}$ can be obtained manually. The horizon $\mathbf{l}_{\mathrm{h}}$ is initialized manually by having a rough idea of the camera setup. Nonetheless, experimental results show that even with a poor initialization of $\mathbf{l}_{\mathrm{s}}$ and $\mathbf{l}_{\mathrm{h}}$, the algorithm always converges to the same solution (see Appendix G). As a result, $\mathbf{l}_{\mathrm{s}}$ and $\mathbf{l}_{\mathrm{h}}$ can be conveniently initialized as the vertical and horizontal lines through the image center respectively (see figure 5.9).

During each iteration of the optimization, a fundamental matrix $\mathbf{F}_{ij}$ between views i and j is computed from the current estimate of the motion parameters using equation (5.10) and the reprojection errors $d_{ij1}(\mathbf{m})$, $d_{ij2}(\mathbf{m})$, $d_{ji1}(\mathbf{m})$ and

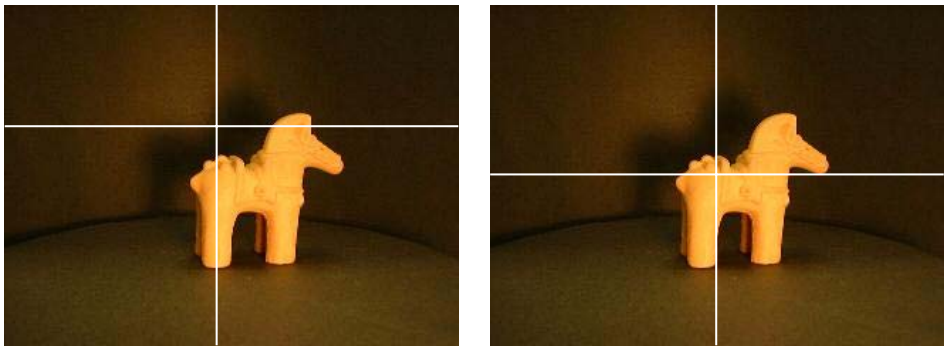


Figure 5.9: By inspecting the image sequence, or by having a rough idea of the camera setup, a very good initialization for the image of the rotation axis $\mathbf{l}_{\mathrm{s}}$ and the horizon $\mathbf{l}_{\mathrm{h}}$ can be obtained manually, as shown in the left image. Alternatively, $\mathbf{l}_{\mathrm{s}}$ and $\mathbf{l}_h$ can be conveniently initialized as the vertical and horizontal lines through the image center respectively, as shown in the right image.

$d_{ji2}(\mathbf{m})$ are determined. The cost function for the circular motion is then given by

$$\mathrm{Cost}_{\mathrm{cm}}(\mathbf{m}) = \sqrt{\frac{1}{4(2N-3)} \sum_{i=1}^{N} \sum_{j=i+1}^{min(i+2,N)} \sum_{k=1}^{2} (d_{ijk}(\mathbf{m})^2 + d_{jik}(\mathbf{m})^2)}, \quad (5.27)$$

where $\mathbf{m}$ consists of the $N+2$ motion parameters. Note that the cost $\mathrm{Cost}_{\mathrm{cm}}(\mathbf{m})$ to be minimized is the rms reprojection error of the epipolar tangents. The cost is minimized using the *conjugate gradient method* [109], with the gradient vector computed by finite differences using a delta change of 10^{-6} for each parameter. Typically, the cost is less than 0.2 (pixels) at the end of the optimization.

5.6.4 Registration of the General Motion

The 6 motion parameters for the optimization of the general motion are initialized by observing the projection of the 3D model built from the circular motion, as described in Section 5.5. This is achieved by using a user-friendly interface in which the rotation and translation of the camera are controlled by the mouse movement. Usually, an initialization obtained by this method gives a very small rms reprojec-

tion error of just a few pixels, and is good enough to avoid local minima and allow convergence to the global minimum in a few iterations.

During each iteration of the optimization, the projection matrix $\mathbf{P}_i$ of the arbitrary general view i is formed using the current estimate of the motion parameters. For each view j, with projection matrix $\mathbf{P}_j$, in the estimated circular motion sequence, a fundamental matrix $\mathbf{F}_{ij}$ is computed from $\mathbf{P}_i$ and $\mathbf{P}_j$ using equation (2.15) and the reprojection errors $d_{ij1}(\mathbf{m}')$, $d_{ij2}(\mathbf{m}')$, $d_{ji1}(\mathbf{m}')$ and $d_{ji2}(\mathbf{m}')$ are determined. The cost function of general motion for view i is then given by

$$\mathrm{Cost}_{\mathrm{gm}}(\mathbf{m}') = \sqrt{\frac{\sum_{j=1}^{N} f_{ij} \sum_{k=1}^{2} (d_{ijk}(\mathbf{m}')^2 + d_{jik}(\mathbf{m}')^2)}{4 \sum_{j=1}^{N} f_{ij}}}, \tag{5.28}$$

where N is the number of views in the estimated circular motion sequence, and $\mathbf{m}'$ consists of the 6 motion parameters for the arbitrary general view i whose pose is to be estimated. The coefficient f_{ij} is determined by the availability of the 2 outer epipolar tangents between views i and j (see Section 5.3). It is 0 if the baseline between views i and j passes through the object, otherwise it is 1. Similar to the circular motion case, the cost $\mathrm{Cost}_{\mathrm{gm}}(\mathbf{m}')$ is the rms reprojection error of the epipolar tangents, and is minimized using the conjugate gradient method with the gradient vector computed by finite differences.

The complete process of generating 3D model from 2D silhouettes is summarized in algorithm 5.1.

5.7 Degenerate Case

A degenerate case for the estimation of circular motion occurs when the object being viewed is a surface of revolution and is being rotated about its axis of revolution. In this situation, there will be no relative motion of the silhouettes. In fact,

Algorithm 5.1 Generation of 3D model from silhouettes.

extract the silhouettes using cubic B-spline snakes;

initialize $\mathbf{l}_{\mathrm{s}}$, $\mathbf{l}_{\mathrm{h}}$ and the $N-1$ angles for the circular motion;
while not converged **do**
  for each view i **do**
    form the fundamental matrices with the next 2 views;
    determine the reprojection errors of the epipolar tangents;
  end for
  compute the cost for the circular motion using (5.27);
  update the $N+2$ motion parameters to minimize the cost
   using conjugate gradient method;
end while

form the set of fundamental matrices from the estimated motion parameters;
upgrade the fundamental matrices to essential matrices
 using the camera calibration matrix;
decompose the essential matrices to form the projection matrices;
build an initial 3D model of the object
 by an octree carving technique (see Chapter 6 for details);
extract a triangulated mesh from the octree
 using the marching cubes algorithm (see Chapter 6 for details);

for each arbitrary general view i **do**
  initialize the 6 motion parameters with the aid of the initial 3D model;
  while not converged **do**
    for each view j in the estimated circular motion sequence **do**
      form the fundamental matrix $\mathbf{F}_{ij}$;
      determine the reprojection errors of the epipolar tangents;
    end for
    compute the cost for the general motion using (5.28);
    update the 6 motion parameters to minimize the cost
     using conjugate gradient method;
  end while
end for
refine the 3D model using the estimated general views;

the silhouettes observed from all viewpoints will be identical, and hence they provide no cues for motion. In [105], Pollick studied human perception of structure and motion from silhouettes, and reported similar results from the human subjects of his experiments.

This degenerate situation can be better understood by considering the parallax-based technique, where the silhouettes are first registered by the harmonic homology $\mathbf{W}$ associated with the circular motion, followed by the computation of the epipoles using common tangents to the registered silhouettes. When the axis of rotation coincides with the revolution axis of the surface of revolution, the silhouettes of the surface will be invariant to the harmonic homology $\mathbf{W}$ (see Section 3.4) and hence the registration has no effect on the silhouettes. All the silhouettes will remain being identical, and thus the epipoles can no longer be located using common tangents. Note that such a degenerate case also occurs when the object is just *locally* a surface of revolution at either ends.

The degenerate situation mentioned above can be easily avoided by simply ensuring that the revolution axis of the surface does not coincide with the axis of rotation. Experiments show that better and more stable results can be obtained by placing the object further from the rotation axis, and typically the degenerate case disappears when the images of the revolution axis and the rotation axis are separated by a distance of about 50 pixels [96].

5.8 Experiments and Results

The first experimental sequence consisted of 18 images of a polystyrene head model taken under controlled circular motion (see figure 5.10). Each image was

taken after rotating the model by 20° on a hand-operated turntable with a resolution of 0.01°. The circular motion was estimated using the algorithm described in Section 5.6.3. Note that neither the knowledge of the rotation angles nor the fact that it was a closed sequence was used in estimating the motion. Figure 5.11 shows the initial and final configurations of the image of the rotation axis and the horizon. Table 5.1 shows the estimated rotation angles between adjacent images and their errors. It can be seen from table 5.1 that the errors in the rotation angles ranged from 0.0077° to 0.4001°, and the rms error of the rotation angles was only 0.2131°. The resulting camera poses and the 3D model built from the estimated motion are shown in figure 5.12 and figure 5.13 respectively.

Table 5.1: Estimated rotation angles between adjacent images.

views	rotation angle	error	views	rotation angle	error
1–2	19.8856°	-0.1144°	10–11	20.1026°	$+0.1026^\circ$
2–3	19.9660°	-0.0340°	11–12	20.0241°	$+0.0241^\circ$
3–4	20.3055°	$+0.3055^\circ$	12–13	20.1651°	$+0.1651^\circ$
4–5	19.9707°	-0.0293°	13–14	20.2053°	$+0.2053^\circ$
5–6	20.0224°	$+0.0224^\circ$	14–15	20.1401°	$+0.1401^\circ$
6–7	19.8686°	-0.1314°	15–16	20.3132°	$+0.3132^\circ$
7–8	20.3860°	$+0.3860^\circ$	16–17	20.0292°	$+0.0292^\circ$
8–9	20.3708°	$+0.3708^\circ$	17–18	19.5999°	-0.4001°
9–10	19.9923°	-0.0077°			

The second experimental sequence consisted of 15 images of a Haniwa (large hollow baked clay sculpture placed on ancient Japanese burial mounds), of which the first 11 images were taken under unknown circular motion of the Haniwa, and the last 4 were taken under unknown general motion (see figure 5.14). The circular motion was first estimated using the algorithm described in Section 5.6.3. The 3D model built from the estimated circular motion alone is shown in figure 5.15.

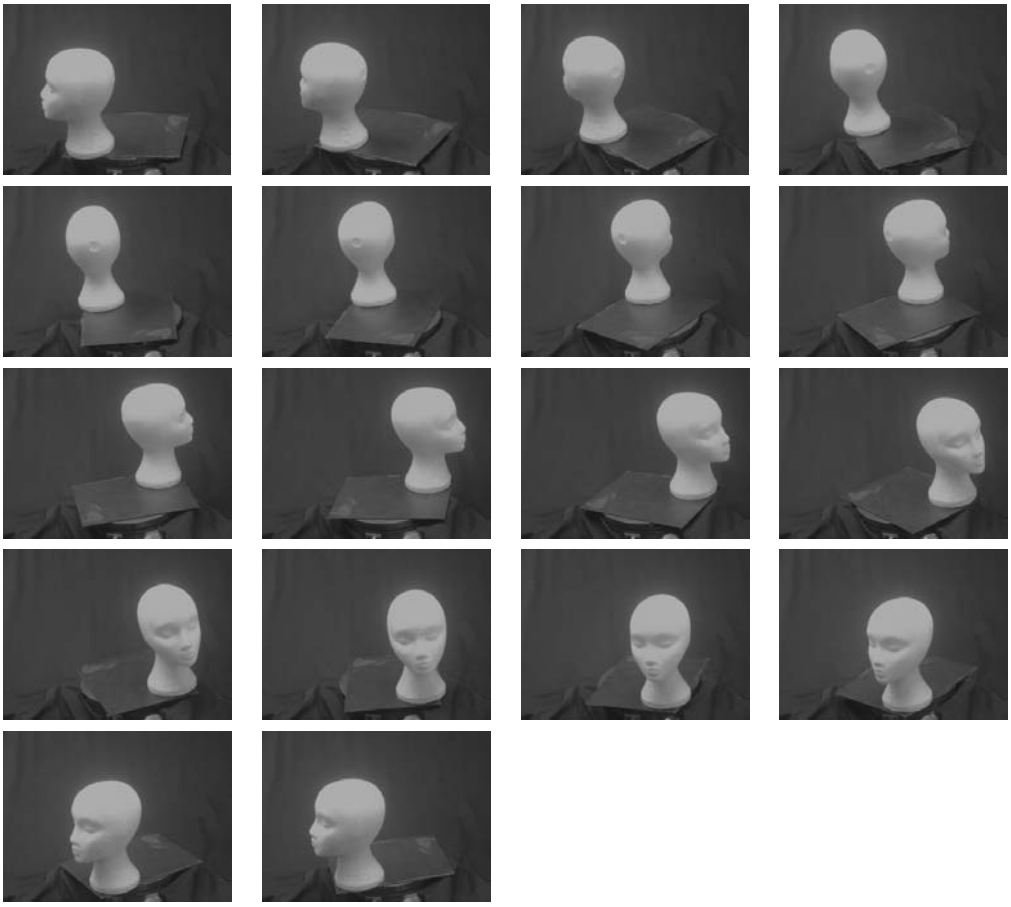


Figure 5.10: Eighteen images of a polystyrene head model under controlled circular motion. Each image was taken after rotating the model by 20° on a turntable with a resolution of 0.01°. Note that the head model is locally close to a surface of revolution at the top and bottom. In order to avoid the degenerate situation mentioned in Section 5.7, it was therefore placed further from the axis of rotation.

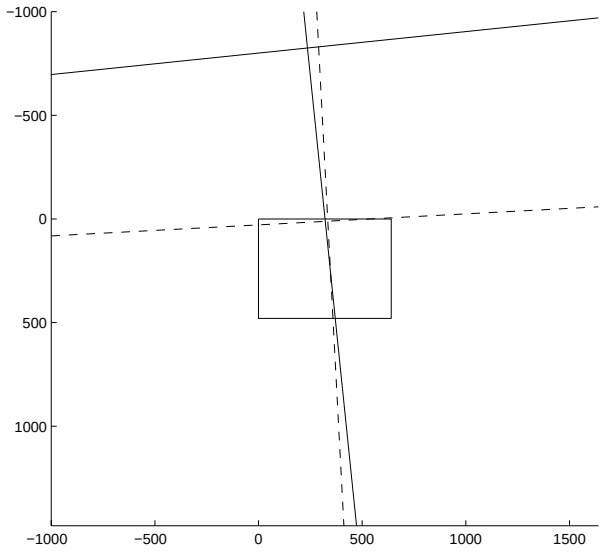


Figure 5.11: The initial (in dash lines) and final (in solid lines) configurations of the image of the rotation axis $\mathbf{l}_s$ and the horizon $\mathbf{l}_h$.

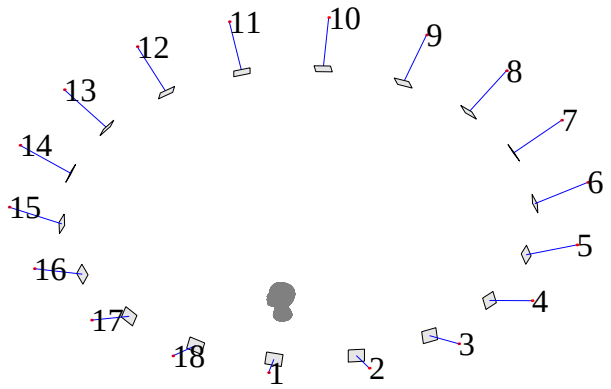


Figure 5.12: Camera poses estimated from the polystyrene head sequence.

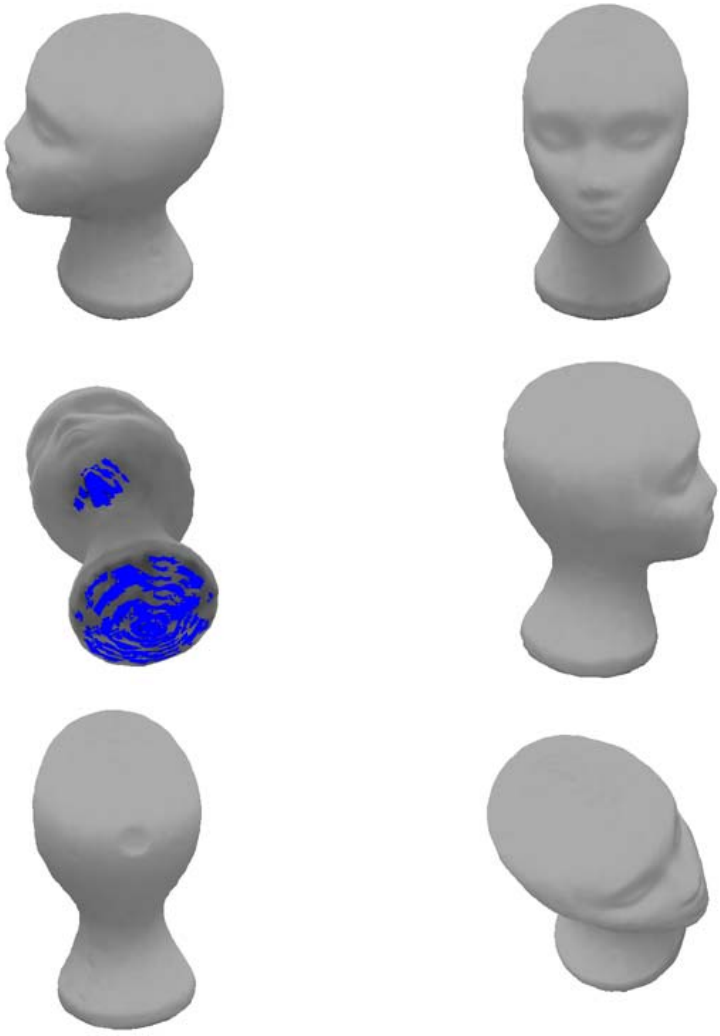


Figure 5.13: 3D model of the polystyrene head built from the estimated circular motion.

The gaps between the legs were not carved away since they never appeared as part of the silhouettes, and textures were missing in areas (top and bottom) which were invisible under circular motion. The last 4 views were then registered using the general motion algorithm described in Section 5.6.4. Figure 5.16 shows the refined model after incorporating the 4 arbitrary general views. The model was now fully covered with textures and showed great improvements in shape, especially in the front, back and top views. The resulting camera poses are shown in figure 5.17.

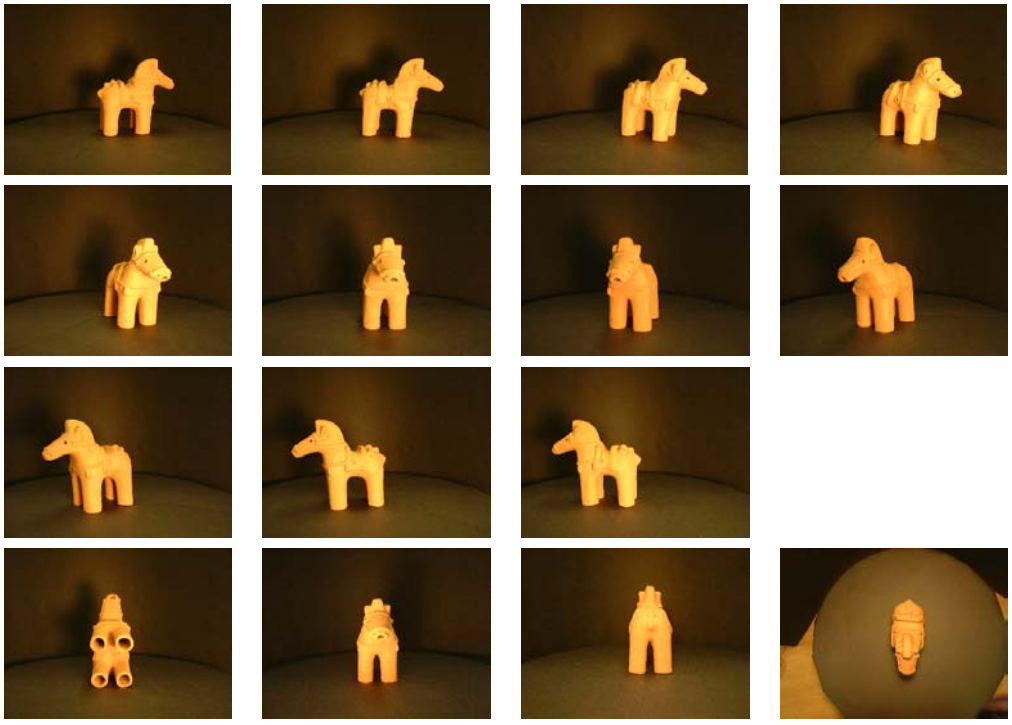


Figure 5.14: Fifteen images of a Haniwa, of which the first 11 images (top 3 rows) were taken under unknown circular motion of the Haniwa, and the last 4 images (bottom row) were taken under unknown general motion.

The third experimental sequence consisted of 13 images of a human head, of which the first 10 images were taken under unknown circular motion of the

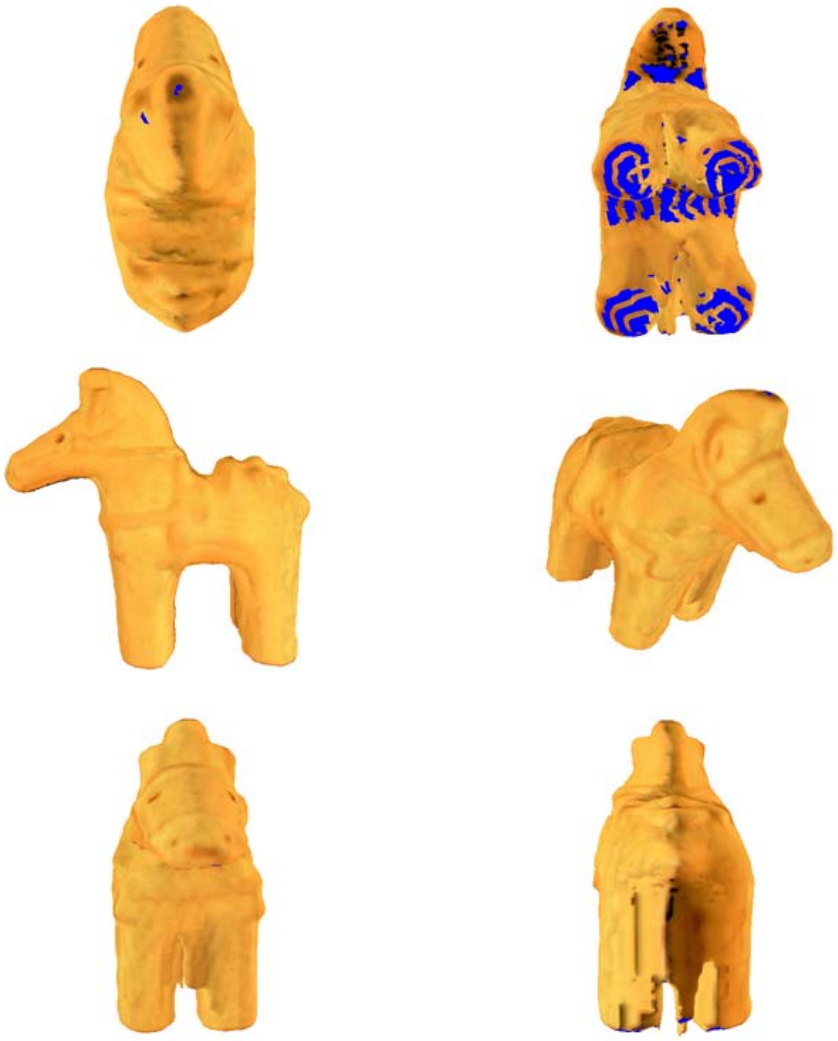


Figure 5.15: 3D model of the Haniwa built from the estimated circular motion alone. The gaps between the legs were not carved away since they never appeared as part of the silhouettes, and textures were missing in areas (top and bottom) which were invisible under circular motion.

Figure 5.16: Refined model of the Haniwa after incorporating the 4 arbitrary general views. The model was now fully covered with textures and showed great improvements in shape, especially in the front, back and top views.

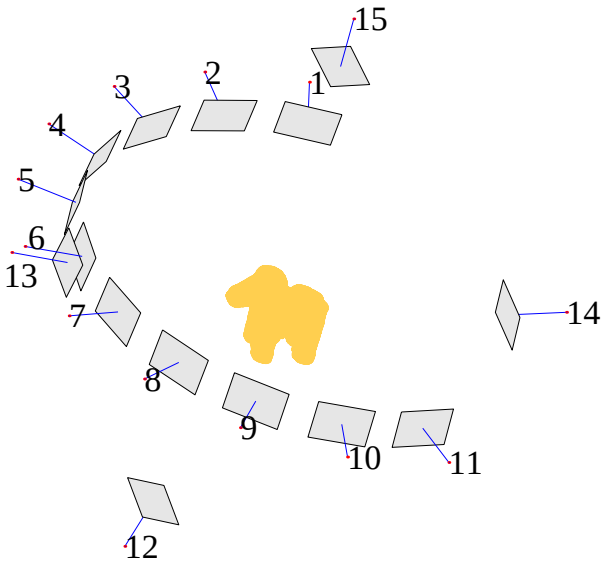


Figure 5.17: Camera poses estimated from the Haniwa sequence.

camera, and the last 3 were taken under unknown general motion (see figure 5.18). The circular motion was, again, first estimated using the algorithm described in Section 5.6.3. The 3D model built from the estimated circular motion alone is shown in figure 5.19, with textures missing at the top of the head and under the chin. The last 3 views were then registered using the general motion algorithm described in Section 5.6.4, and the refined model after incorporating the 3 arbitrary general views is shown in figure 5.20. The top of the head and the chin were now covered with textures. The resulting camera poses are shown in figure 5.21.

The fourth experimental sequence consisted of 9 images of a Haniwa taken in front of a calibration grid (see figure 5.22), and was used for quantitative evaluation. Each view in the sequence was calibrated independently using the DLT technique followed by an optimization which minimized the reprojection errors of the corner features from the calibration grid. The algorithm for estimating the

Figure 5.18: Thirteen images of a human head, of which the first 10 images (top 3 rows) were taken under unknown circular motion of the camera, and the last 3 images (bottom row) were taken under unknown general motion.

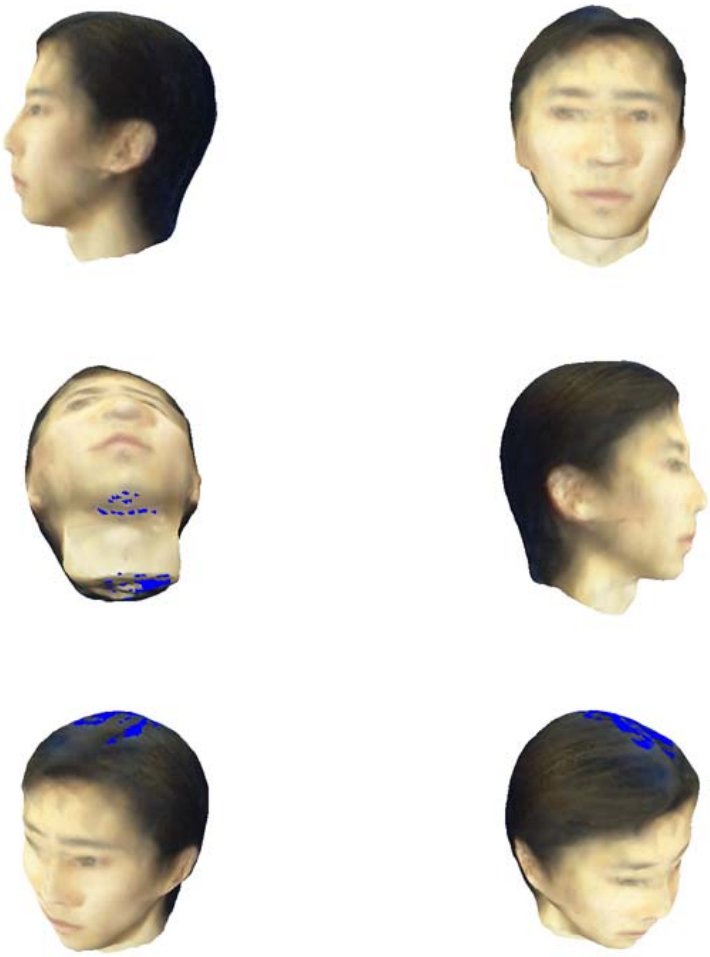


Figure 5.19: 3D model of the human head built from the estimated circular motion alone. Textures were missing at the top of the head and under the chin.

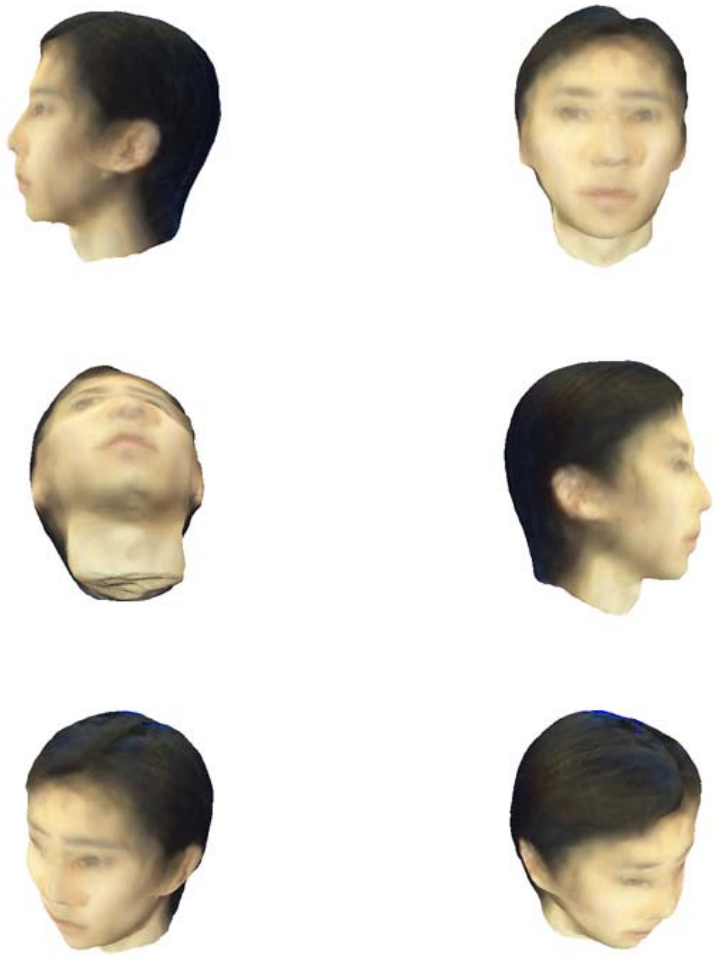


Figure 5.20: Refined model of the human head after incorporating the 3 arbitrary general views. The top of the head and the chin were now covered with textures.

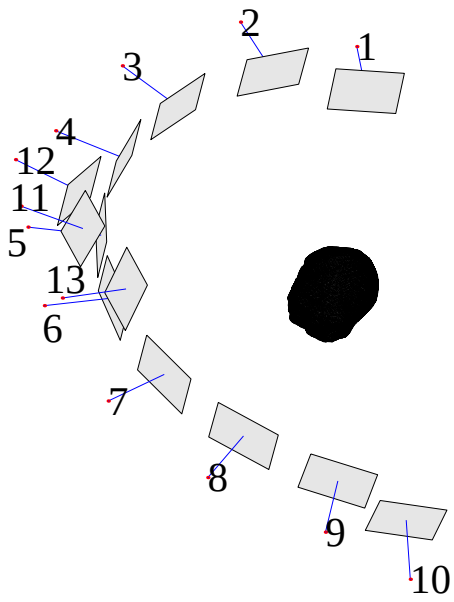


Figure 5.21: Camera poses estimated from the human head sequence.

pose of an arbitrary general view, as described in Section 5.6.4, was then used to register the 9^{th} view using different subsets of the first 8 images. The results are presented in table 5.2 which shows the rms reprojection errors of the corner features from the calibration grid in the 9^{th} view. Though the errors resulted from the motion estimated using epipolar tangents were not as small as that from the calibration using the calibration grid (which directly minimized the reprojection errors of those corner features), the results were indeed very good since only 6–16 epipolar tangent points had been used, compared with 192 corners used in the case of calibration using the calibration grid. Besides, the cameras were positioned relatively far from the Haniwa so as to keep the calibration grid visible inside the images. As a result, each silhouette of the Haniwa occupied only a very small region of the image and this limited the accuracy that could be achieved by the algorithm.

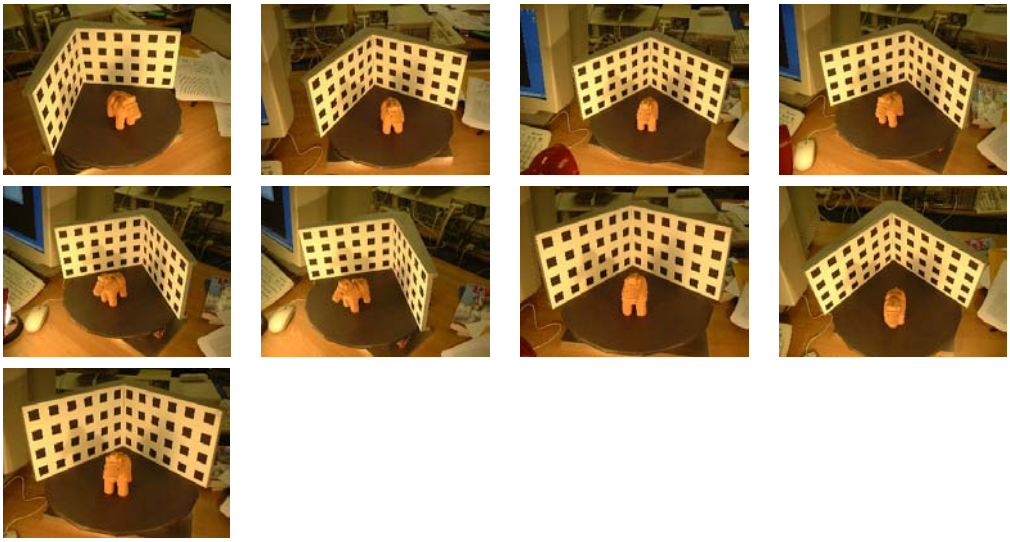


Figure 5.22: Nine images of a Haniwa in front of a calibration grid used for quantitative evaluation.

Table 5.2: Reprojection errors (in pixels) of the 192 corner features from the calibration grid.

motion estimated from: silhouettes in views	motion estimated from: no. of tangent pts	reprojection error: rms (in pixels)
1–3	6	1.1824
1–4	8	0.9407
1–5	10	0.8858
1–6	12	0.7311
1–7	14	0.7856
1–8	16	0.7963

ground truth	corners	rms (in pixels)
calib. grid	192	0.3853

The fifth experimental sequence consisted of 10 images of a human head acquired from an imperfect circular motion of the camera (see figure 5.23). Due to vibration of the rotating arm to which the camera was attached, the camera wobbled up and down during the circular motion. The camera poses, obtained by applying the algorithm described in Section 5.6.3 for circular motion, are shown in figure 5.24, and the resulting 3D model is shown in figure 5.25. As shown in figure 5.24, the camera was constrained to follow a perfect circular path. However, since the camera did not actually follow a circular path, the reconstructed head model was highly distorted. These camera poses were then iteratively refined by applying the general motion algorithm. Each view in the sequence was taken in turn and registered using the rest of the views, and the process was repeated until there were no further improvements in the reprojection errors of the epipolar tangents. The refined camera poses showed the wobbliness of the actual camera motion (see figure 5.27). The 3D model built from the refined motion is shown in figure 5.26, and it showed great improvements over the model shown in figure 5.25.

The last experimental sequence consisted of 14 images of an outdoor sculpture acquired by a hand-held camera (see figure 5.28). An approximate circular motion of the camera was achieved by using a string which was fixed to the ground by a peg at one end. A circular path on the ground was then obtained by rotating the free end of the string about its fixed end. Each image in the sequence was acquired by positioning the camera roughly at a fixed height above the free end of the (rotating) string, and pointing it towards the sculpture. Note that since the camera center, the string and the rotation axis were roughly coplanar, the image of the string in each image provided a very good estimate for the image of the rotation

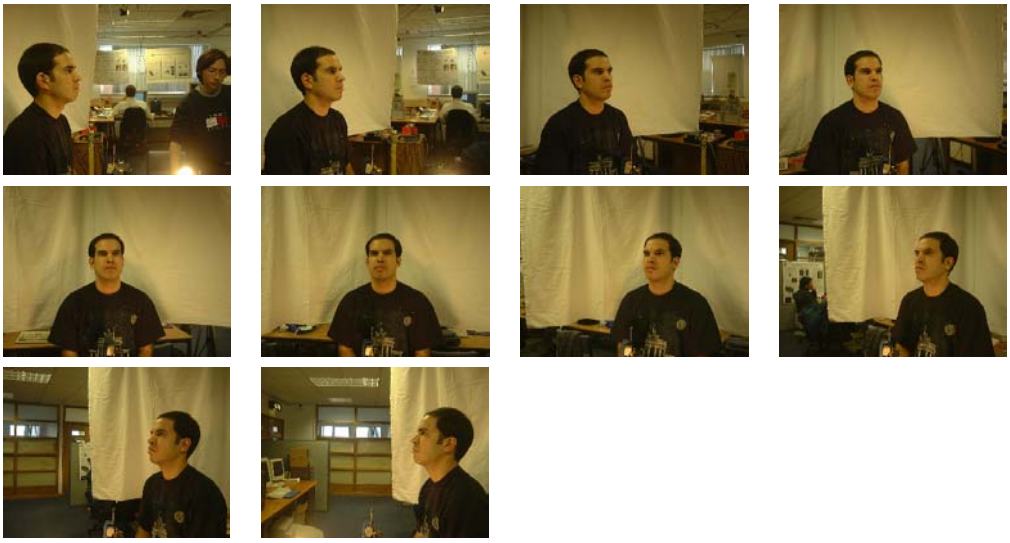


Figure 5.23: Ten images of a human head acquired from an imperfect circular motion of the camera. Due to vibration of the rotating arm to which the camera was attached, the camera wobbled up and down during the circular motion.

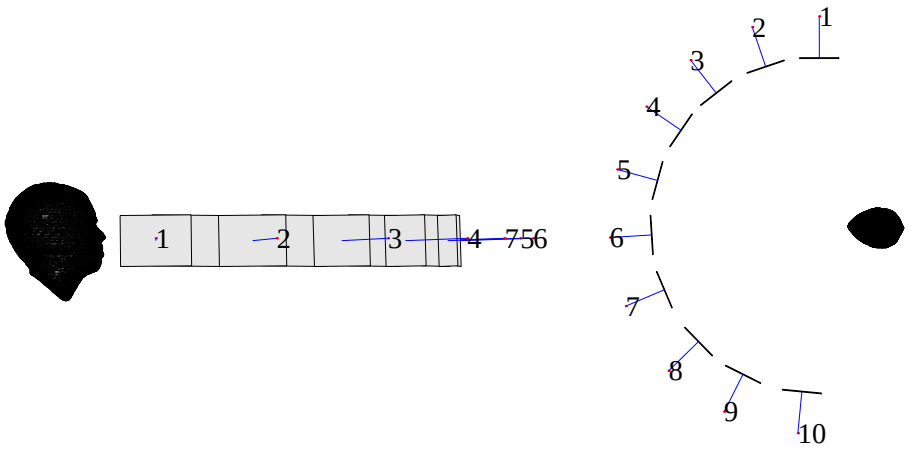


Figure 5.24: Camera poses obtained by assuming circular motion.

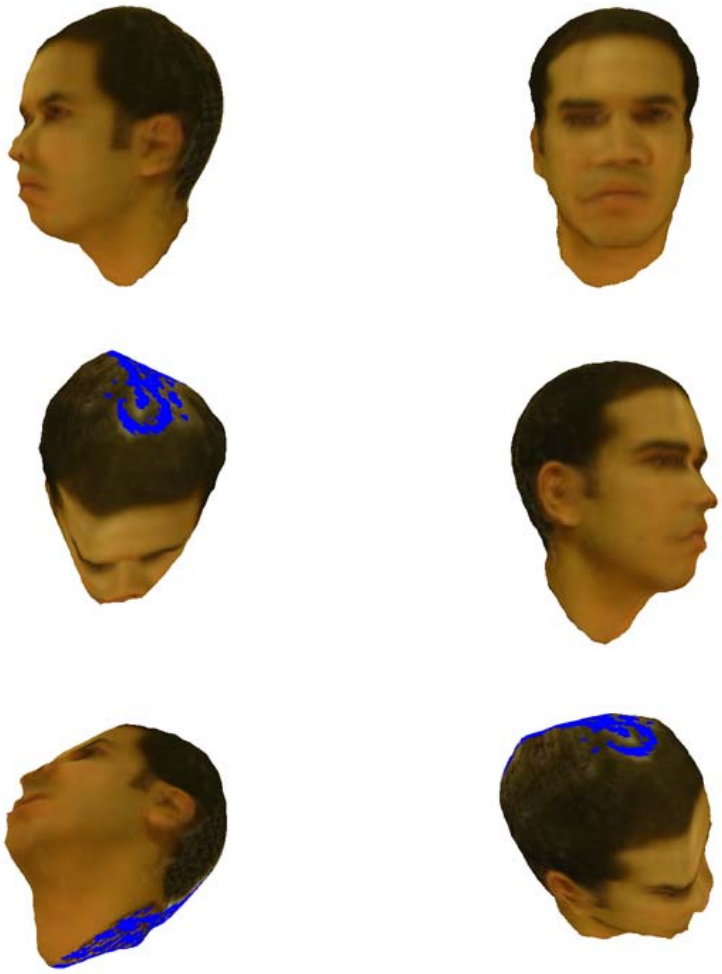


Figure 5.25: 3D model of the human head built from the estimated circular motion. Since the camera did not actually follow a circular path, the reconstructed head model was highly distorted.

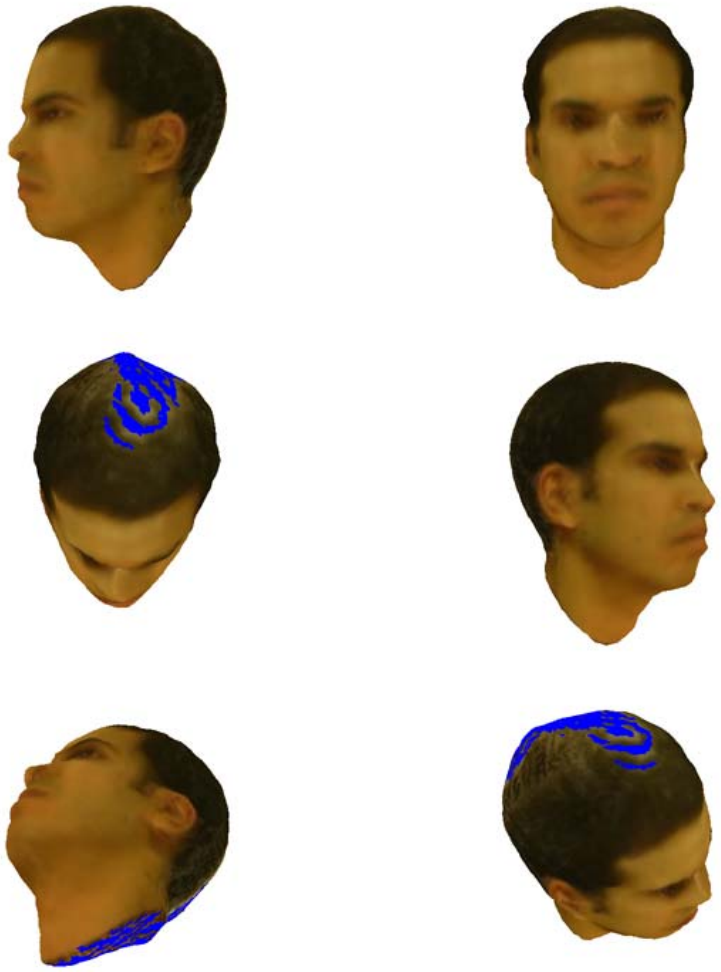


Figure 5.26: 3D model of the human head built from the refined motion.

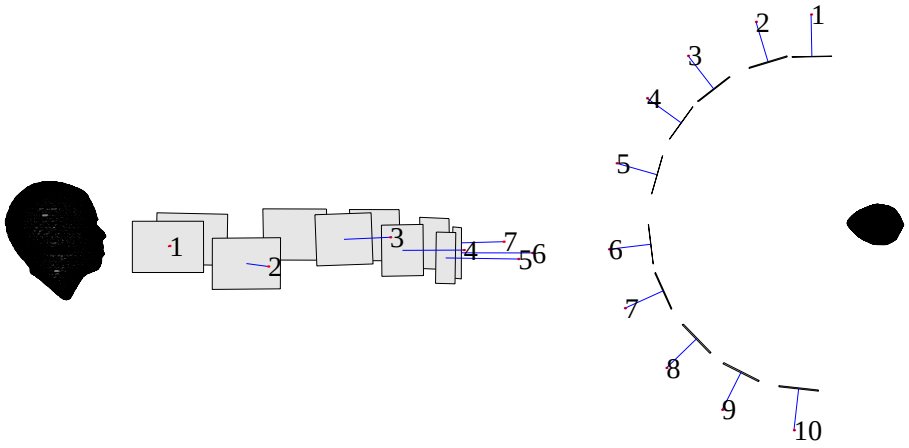


Figure 5.27: Camera poses obtained by iteratively refining the camera poses from the estimated circular motion using the general motion algorithm. The refined camera poses showed the wobbliness of the actual camera motion.

axis $\mathbf{l}_s$. Although the camera center roughly followed a circular path, the orientation of the camera was unconstrained and hence the image of the rotation axis $\mathbf{l}_s$ and the horizon $\mathbf{l}_h$ were not fixed throughout the image sequence. In order to allow the camera motion to be estimated using the circular motion algorithm described in Section 5.6.3, the images were first rectified using the technique described in Section 4.6.2 so that the image of the string (i.e. the image of the rotation axis) became a fixed vertical line passing through the principal point throughout the sequence. A transformation induced by a rotation about the x-axis of the camera was then applied to each image so that the image of the fixed end of the string became a fixed point on $\mathbf{l}_s$ throughout the rectified sequence (see figure 5.29). The resulting image sequence resembled a circular motion sequence, in which the horizon $\mathbf{l}_h$, the image of the rotation axis $\mathbf{l}_s$, and the special vanishing point $\mathbf{v}_x$ were fixed (see figure5.30). The algorithm for circular motion estimation was then applied to this rectified sequence, and the resulting camera poses were then

iteratively refined by applying the general motion algorithm. The final camera poses estimated from the rectified sequence are shown in figure 5.31, and the 3D model built from the estimated motion is shown in figure 5.32.

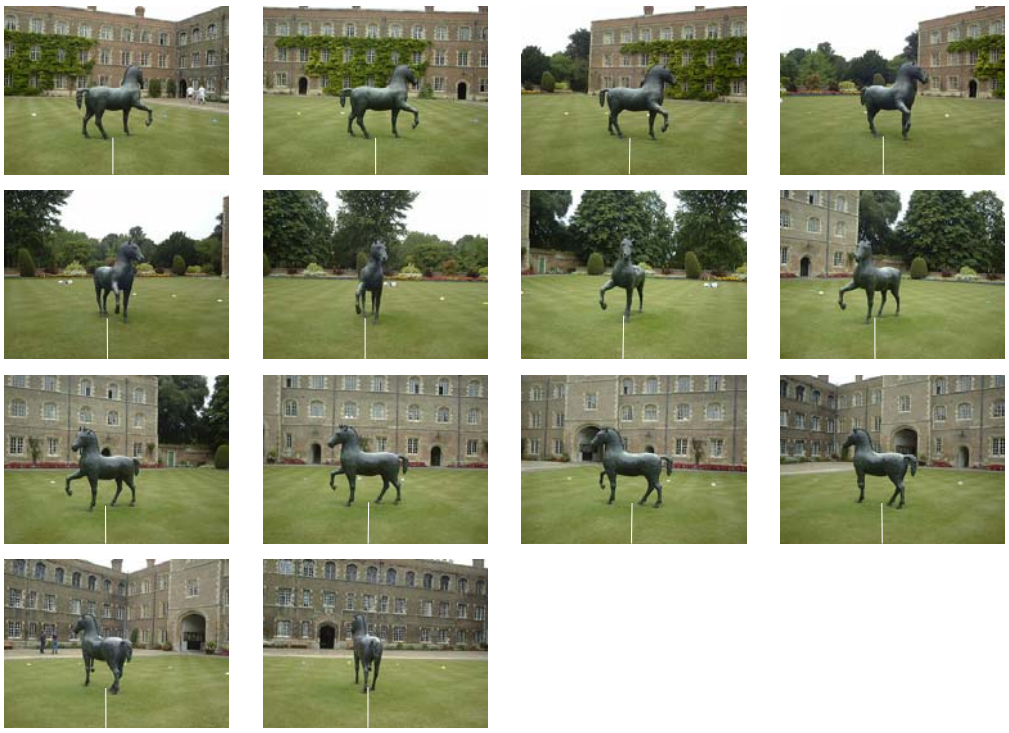
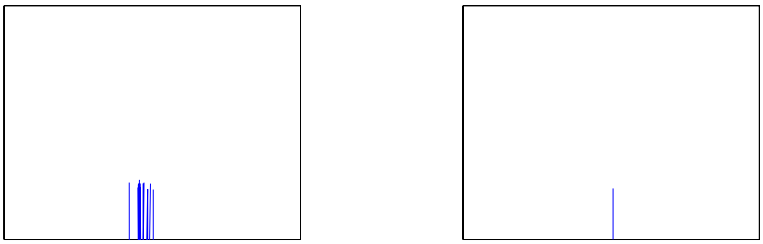


Figure 5.28: Fourteen images of an outdoor sculpture acquired by a hand-held camera. Although the camera center roughly followed a circular path, the orientation of the camera was unconstrained and hence the image of the rotation axis and the horizon were not fixed throughout the image sequence.

5.9 Discussions

In this chapter, a complete and practical system for generating high quality 3D models from 2D silhouettes is presented. The input to the system is an image sequence of an object under both unknown circular motion and unknown general motion. The circular motion is exploited to provide a simple parameterization of



(a) (b)

Figure 5.29: (a) The images of the string in the outdoor sequence did not coincide, and this implied that the image of the rotation axis was not fixed throughout the sequence. (b) The images were rectified so that the image of the string became a fixed vertical line passing through the principal point of the camera and the image of the fixed end of the string became a fixed point throughout the sequence.

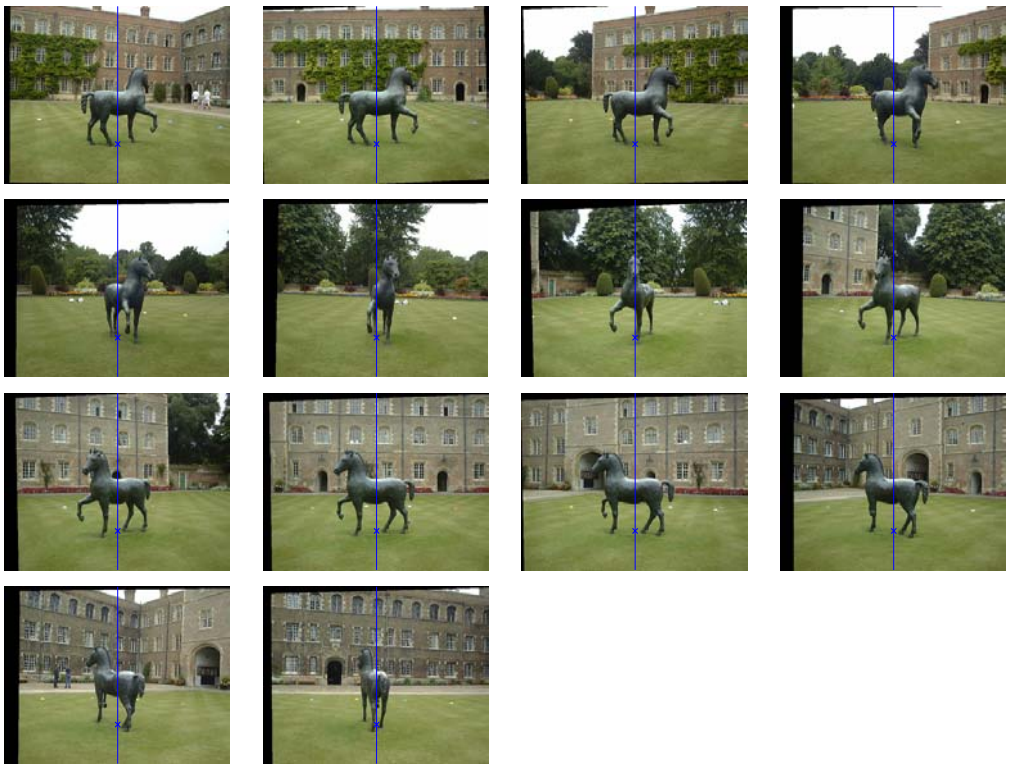


Figure 5.30: The rectified sequence resembled a circular motion sequence in which the horizon $\mathbf{l}_{\mathrm{h}}$, the image of the rotation axis $\mathbf{l}_{\mathrm{s}}$, and the special vanishing point $\mathbf{v}_x$ were fixed.

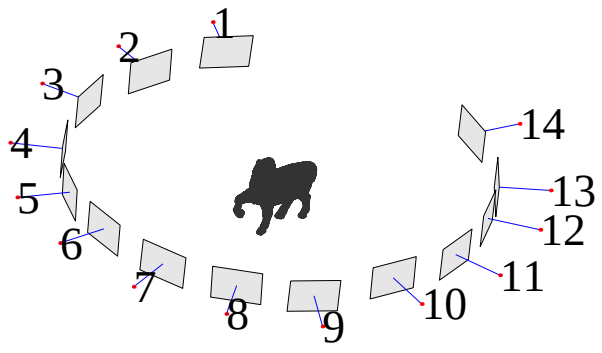


Figure 5.31: Camera poses estimated from the rectified outdoor sculpture sequence.

the fundamental matrix relating any pair of views in the circular motion sequence. Such a parameterization greatly reduces the dimension of the search space for the optimization problem, which can now be solved using only the 2 outer epipolar tangents. The parameterization also leads to a trivial initialization of the parameters which all bear physical meanings (i.e. image of rotation axis, horizon and rotation angles). In the case of complete circular motion with dense image sequence, the harmonic homology associated with the image of the surface of revolution swept by the rotating object can be exploited to obtain the image of the rotation axis conveniently and independently.

The incorporation of arbitrary general views reveals information which is concealed under circular motion, and greatly improves both the shape and textures of the 3D models. It also allows incremental refinement of the 3D models by adding new views at any time, without the need of setting up the exact, identical scene carefully. The registration of general motion using circular motion (or alternatively 3 or more known views) avoids the problems of local minima and nontrivial initialization, which exist in every algorithm for general motion esti-

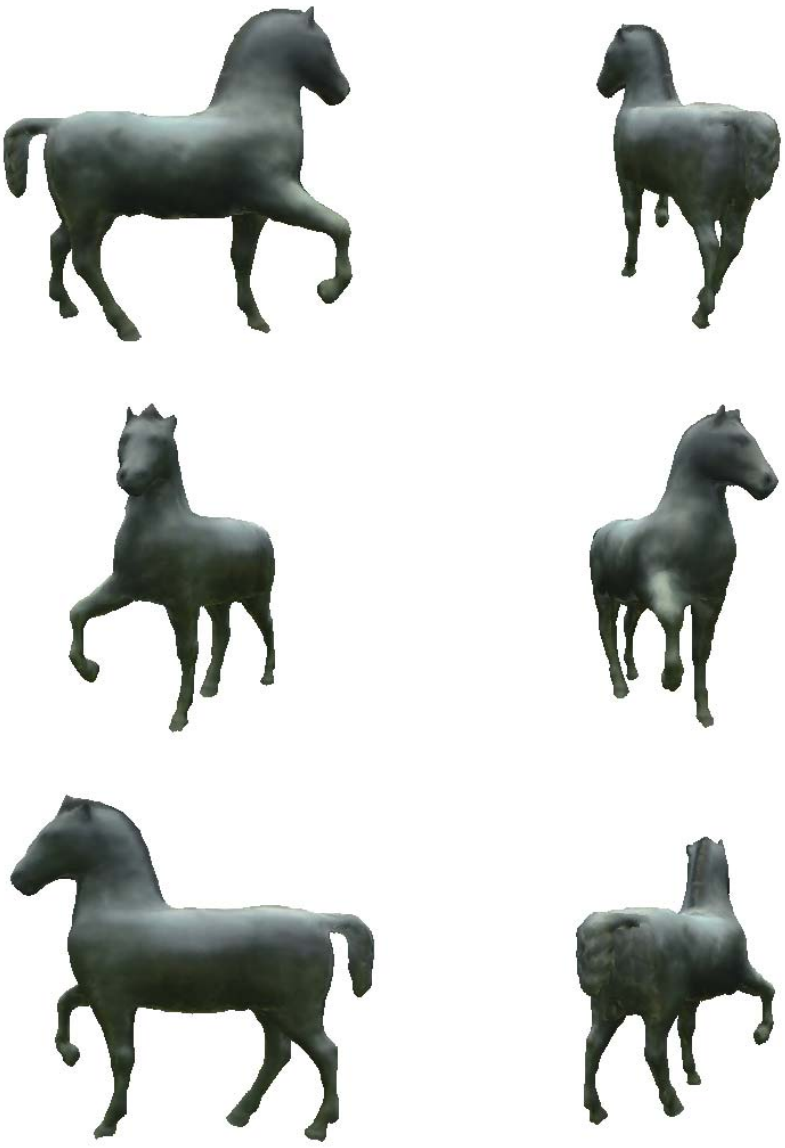


Figure 5.32: 3D model of the outdoor sculpture built from the estimated motion.

mation using silhouettes. Like the algorithm for circular motion estimation, the registration of the general motion requires only the 2 outer epipolar tangents. In the case of approximate (imperfect) circular motion, the motion can be estimated by first assuming circular motion. The camera poses obtained from the circular motion algorithm are then iteratively refined by using the general motion algorithm. Since only silhouettes have been used in both the motion estimation and model reconstruction, no corner detection nor matching is necessary. This means that the system is capable of reconstructing any kind of objects, including *smooth* and *textureless* surfaces. Experiments on various objects had produced convincing 3D models, demonstrating the practicality of the system.

Chapter 6

Reconstruction from Silhouettes: Implementation

"I can't work without a model."

- Vincent van Gogh.

6.1 Introduction

In Chapter 5, a complete and practical system for generating high quality 3D models from 2D silhouettes has been introduced. The system can be decomposed into 2 modules, namely the *motion module* and the *structure module*. The motion module is responsible for estimating the camera motion from the silhouettes, whereas the structure module is responsible for producing 3D models from the silhouettes and the estimated camera poses. The algorithms and implementations for motion estimation from silhouettes, which form the core of the motion module, have been presented in Chapter 5. This chapter studies the problem of model reconstruction from silhouettes, and gives the algorithms and implementation details for the structure module. Results on real data are presented, showing the quality of the reconstruction, as well as the quality of the motion estimated using the

techniques introduced in Chapter 5.

A survey of the literature on model reconstruction from silhouettes is given in Section 6.2. Section 6.3 briefly reviews the octree representation, and an efficient algorithm for constructing an octree using silhouettes from multiple views is presented in Section 6.4. Section 6.5 gives the implementation details for the silhouette extraction and intersection test. The extraction of a triangulated mesh from the octree is then described in Section 6.6. Experimental results on real data are presented in Section 6.7, followed by discussions in Section 6.8.

6.2 Previous Works

The surface reconstruction of smooth objects from silhouettes was pioneered by Giblin and Weiss [46]. Under the assumption of orthographic projection, they demonstrated that a surface can be reconstructed from the envelope of all its tangent planes computed directly from the family of silhouettes of the surface under planar viewer motion. Cipolla and Blake [24] extended the studies of Giblin and Weiss to curvilinear viewer motion under perspective projection, and developed the *osculating circle method* by introducing the *epipolar parameterization*. Vaillant and Faugeras [134] developed a similar technique in which the surface is parameterized by the *radial curves* instead of the *epipolar curves*. Based on the osculating circle method, Szeliski and Weiss [125] used a linear smoother to compute epipolar curves on the whole surface together with an estimate of uncertainty, and reported improvements in the reconstruction. In [12], Boyer and Berger derived a depth formulation from a local approximation of the surface up to order two for discrete motion. In [140], Wong et al. developed a simple technique based

on a finite-difference implementation of [24]. Despite its simplicity, the method developed in [140] was reported to produce results comparable to those in [24] and [12].

The volume intersection technique for constructing volumetric descriptions of objects from multiple views was first proposed by Martin and Aggarwal [90], who introduced the *volume segment* representation. In [21], Chien and Aggarwal presented an algorithm for generating an octree of an object from 3 orthogonal views under orthographic projection. Their work was further developed by Ahuja and Veenstra [2], who extended the algorithm to handle images from any subset of 13 standard viewing directions. In [56], Hong and Shneier introduced a technique for generating an octree from multiple arbitrary views under perspective projection. Their approach first constructs an octree for each image by projecting the octree cubes onto the image and intersecting their projections with the silhouette, and the final octree of the object is given by the intersection of the octrees obtained from all images. In [108], Potmesil described a similar approach in which the images are represented by *quadtrees* to facilitate the intersection of the projections of the cubes with the silhouettes. Other similar approaches also include [103] and [120], where the octree for each image is constructed by intersecting, in 3D space, the octree cubes with the polyhedral cone formed from the back-projection of the silhouette. In [124], Szeliski introduced an efficient algorithm which constructs an octree in a hierarchical coarse-to-fine fashion. His approach is similar to that of [108], except that only a single octree is constructed using all the images simultaneously.

In this chapter, the volume intersection approach is chosen due to its ability to describe objects with more complex topologies (e.g. object with holes). Based

on [124], an algorithm for generating an octree using silhouettes from multiple views is presented. The main difference between the work presented in [124] and the technique developed here is that instead of using a *background subtraction* technique as described in [124], the object/background binary images are computed directly from the B-spline snakes which are used to extract and represent the silhouettes during motion estimation (see Chapter 5). The sub-pixel accuracy of the B-spline snakes allows a binary image to have a resolution higher than the original image, and this may help to improve the cube classification when the object is relatively small in the image. For the sake of display, a triangulated mesh is extracted from the octree, and the colors of the vertices in mesh are estimated from the original images.

6.3 Octree Representation

An octree [63, 92] is a tree data structure in which each non-leaf node has at most 8 child nodes. It is commonly used in computer graphics to provide a volumetric representation of an object, where each node in the tree represents a *voxel* (volume element) in space. The root node of the octree consists of a single large voxel which defines the bounding volume of the object. The octree is constructed by recursively subdividing each voxel in the tree into 8 sub-voxels, which are represented by the 8 child nodes. Each node in the tree is assigned one of the 3 colors (black, gray and white) according to its occupancy. A *black node* represents a voxel which is totally occupied, a *gray node* represents a voxel which is partially occupied, and a *white node* represents a voxel which is completely empty. Note that both black and white nodes do not have any child node, and hence they are

leaf nodes in the tree. A gray node is an interior node in the tree, which has child nodes with different colors. It represents a voxel which lies on the boundary (surface) of the object. Figure 6.1 shows a simple volume represented by an octree and the corresponding tree structure with colored nodes. Further details on octree representations, constructions and manipulations can be found in [18, 20, 113].

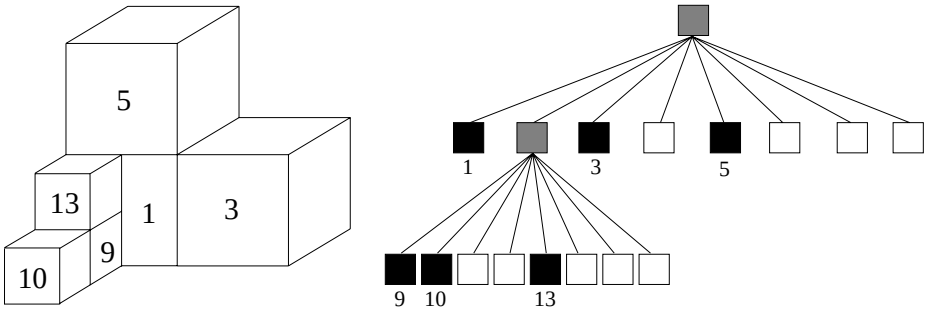


Figure 6.1: A simple volume represented by an octree and the corresponding tree structure with colored nodes.

In the implementation presented in this chapter, the voxels are cubes and each subdivision produces 8 identical sub-cubes. Each node in the octree stores the color (occupancy), the length, and the 3D coordinates of the center of the cube it represents. It also contains pointers to its child nodes, if there are any. In addition, each node also stores an 8-bit index which represents the occupancy of the 8 corners of the cube (see figure 6.2). This 8-bit index is used to index into a lookup table during the marching cubes algorithm [80] for extracting a triangulated mesh from the octree. In order to allow fast access to the cubes in a particular level, all cubes in the same level are stored in a level-list and all the level-lists are kept in an array.

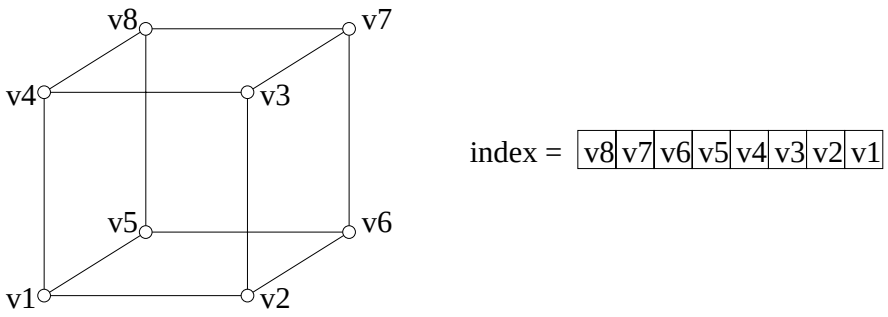


Figure 6.2: An 8-bit index indicating the occupancy of the 8 corners of a cube. A bit is set to 1 if the corresponding corner is occupied, otherwise it is set to 0.

6.4 Octree Construction from Multiple Views

For the purpose of generating an octree from silhouettes, the interpretation of the colors used in the octree representation is slightly modified. A black node here represents a cube which lies completely inside the object, a white node represents a cube which lies completely outside the object, and a gray node represents a cube which is ambiguous.

The root node of the octree is first initialized as a single gray cube which completely encloses the object. To refine the octree, a new level is first formed by subdividing each gray cube in the finest level into 8 sub-cubes. Note that black and white cubes do not need to be subdivided since all their child nodes will have the same classifications as their parent nodes. Initially, all the cubes in the new level are assumed to be completely inside the object and are assigned the black color, and the 8-bit index of each cube is set to 255. The cubes are then projected onto each image and tested for intersection with the silhouette so as to determine their occupancy and have their colors updated accordingly.

Each cube in the new level is projected onto each image in the sequence using

the associated projection matrix. If its projection lies completely outside the silhouette in the current image, it must lie completely outside the object. It is then assigned the white color and further checking against other images is not necessary. If its projection lies partially inside the silhouette in the current image, it must lie close to the boundary of the object. Its color is then updated to gray to indicate that its occupancy is ambiguous, and thus further refinement is needed. Finally, if its projection lies completely inside the silhouette in the current image, its occupancy cannot be determined and hence it just keeps its current color. If the cube remains being black after checking its projections against all the silhouettes, it must then lie completely inside the object. Note that the color of a cube can only change from black to gray, from back to white, or from gray to white. This is because cubes can only be removed or "carved away" from the octree.

The above refinement process is repeated until there is no gray cube in the finest level, or, in practice, a preset resolution level is reached. The algorithm for generating an octree using silhouettes from multiple views is summarized in algorithm 6.1. This algorithm is very efficient in that white cubes are identified in the earliest possible stage, and thus it avoids all unnecessary cube projections, intersection tests and cube subdivisions. Since only gray cubes in the finest level are being considered and refined during each iteration of the refinement process, care must be taken not to classify a cube as black (i.e. completely inside) or white (i.e. completely outside) unless this is certain. On the contrary, if a black or white cube is wrongly assigned the gray color, it only indicates that the occupancy of the cube is ambiguous and the cube will be reconsidered and refined in the next level. The implementation details for the silhouette extraction and intersection test are given in the next section.

Algorithm 6.1 Octree construction using silhouettes from multiple views.

```
initialize the root node of the octree as a single gray cube
 that completely encloses the object;

while max level not reached do
  if no gray cube in the finest level then
    break the while-loop;
  end if
  for each gray cube in the finest level do
    subdivide it into 8 sub-cubes;
    for each sub-cube do
      set its color to black;
      set its 8-bit index to 255;
      for each image in the sequence do
        project the cube onto the image
         using the associated projection matrix;
        if the projection lies completely outside the silhouette then
          update the cube's color to white;
          break the inner for-loop;
        else if the projection lies partially inside the silhouette then
          update the cube's color to gray;
        else
          keep the current color of the cube;
        end if
      end for
    end for
  end for
end while
```

6.5 Silhouette Extraction and Intersection Test

In Chapter 5, closed cubic B-spline snakes are used to extract and represent the silhouettes from the image sequence. In this chapter, for the sake of the intersection test, the silhouettes are represented by object/background binary images where object pixels and background pixels are represented by 1s and 0s respectively. Instead of using background subtraction techniques [121, 127, 62], the binary images are computed directly from the B-spline snakes which are obtained during the motion estimation stage (see Chapter 5). The sub-pixel accuracy of the B-spline snakes allows a binary image to have a resolution higher than the original image, and this may help to improve the cube classification when the object is relatively small in the image. For each B-spline snake in a image, a binary image at a chosen resolution, with the region enclosed by the snake filled with 1s, is constructed using some conventional graphics drawing routines. The object/background binary image is then obtained by combining these binary images using "xor" (see figure 6.3).

To classify a cube in the octree, the 8 corners of the cube are first projected onto the binary image, and the 8-bit index of the cube is updated by setting the bits corresponding to those corners which are projected onto background pixels to 0s. The projection of the cube, which is a hexagon in general, is then approximated by its bounding box computed from the projections of the 8 corners, and the pixels of the binary image within the bounding box are examined. If the bounding box is completely occupied (i.e. all pixels are 1s), the projection of the cube must also be completely occupied. Similarly, if the bounding box is completely empty (i.e. all pixels are 0s), the projection of the cube must also be completely empty.

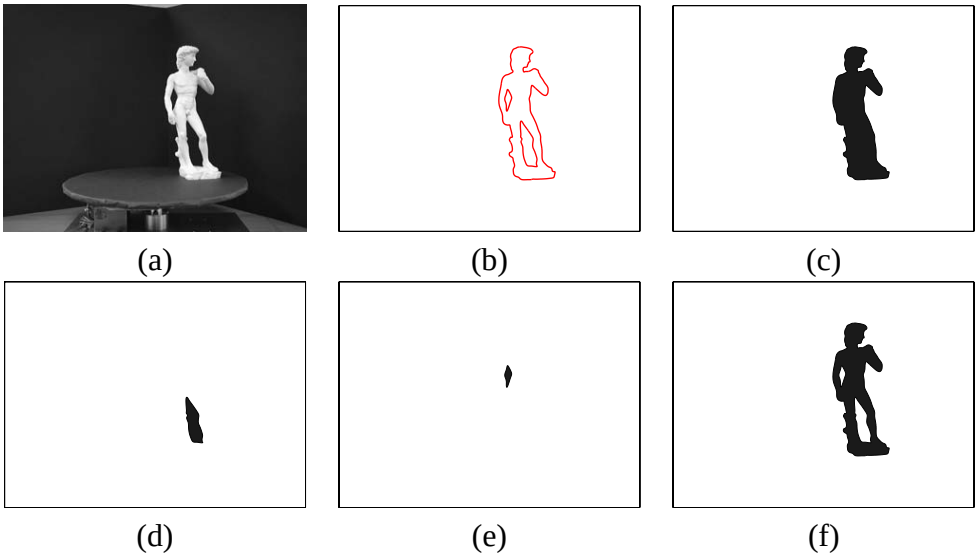


Figure 6.3: (a) The original image of a miniature David statue. (b) The silhouette is extracted and represented by 3 closed B-spline snakes. (c)–(e) A binary image is formed from each of the B-spline snakes. (f) The object/background binary image is obtained by combining the binary images in (c)–(e) using "xor".

Since the bounding box is always bigger than or equal to the actual projection of the cube, there maybe chances when the bounding box is only partially occupied whereas the actual projection of the cube is completely occupied or completely empty (see figure 6.4). In such situations, the cube will be classified as ambiguous. Nonetheless, this only postpones the classification of the cube and causes no harm to the algorithm (see Section 6.4).

6.6 Surface Extraction and Coloring

In order to allow the reconstructed 3D model to be displayed efficiently with conventional graphics rendering algorithms (implemented either in hardware or software), a triangulated mesh is extracted from the octree using standard marching cubes algorithm [80]. Due to its practicality and simplicity, the marching cubes

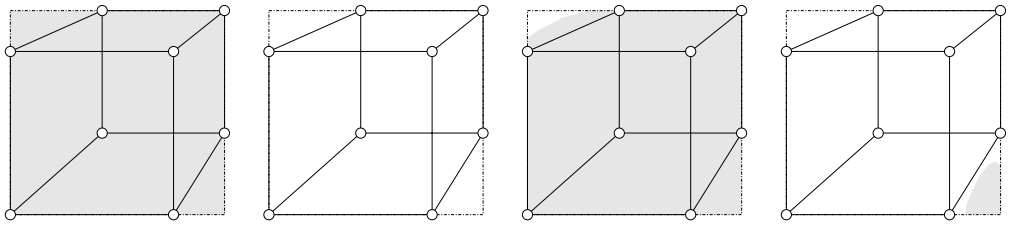


Figure 6.4: (a) If the bounding box is completely occupied, so is the projection of the cube. (b) If the bounding box is completely empty, so is the projection of the cube. (c) The bounding box is partially occupied but the projection of the cube is completely occupied. (d) The bounding box is partially occupied but the projection of the cube is completely empty.

algorithm has been widely using for visualizing volumetric data like those produced from computed tomography (CT), magnetic resonance (MR) and single-photon emission computed tomography (SPECT). The marching cubes algorithm uses the occupancy information of the 8 corners of a cube to determine how the surface intersects the edges of the cube, and produces triangle patches that best approximate the surface. Since there are 8 corners in a cube and each corner can either be inside or outside the surface, there are totally $2^8 = 256$ ways a surface can intersect the cube. By complementary symmetry and rotational symmetry considerations, Lorensen and Cline [80] showed that these 256 cases can be reduced to 15 patterns for which they developed explicit triangulations (see figure 6.5). A lookup table consisting of the triangulation information for the 256 cases was then built from the permutation of these 15 basic patterns. An 8-bit index, constructed from the occupancy information of the 8 corners of a cube (see figure 6.2), is used to index into this lookup table to produce triangle patches for that cube.

To extract surface triangles from the octree, the 8-bit index of each gray cube

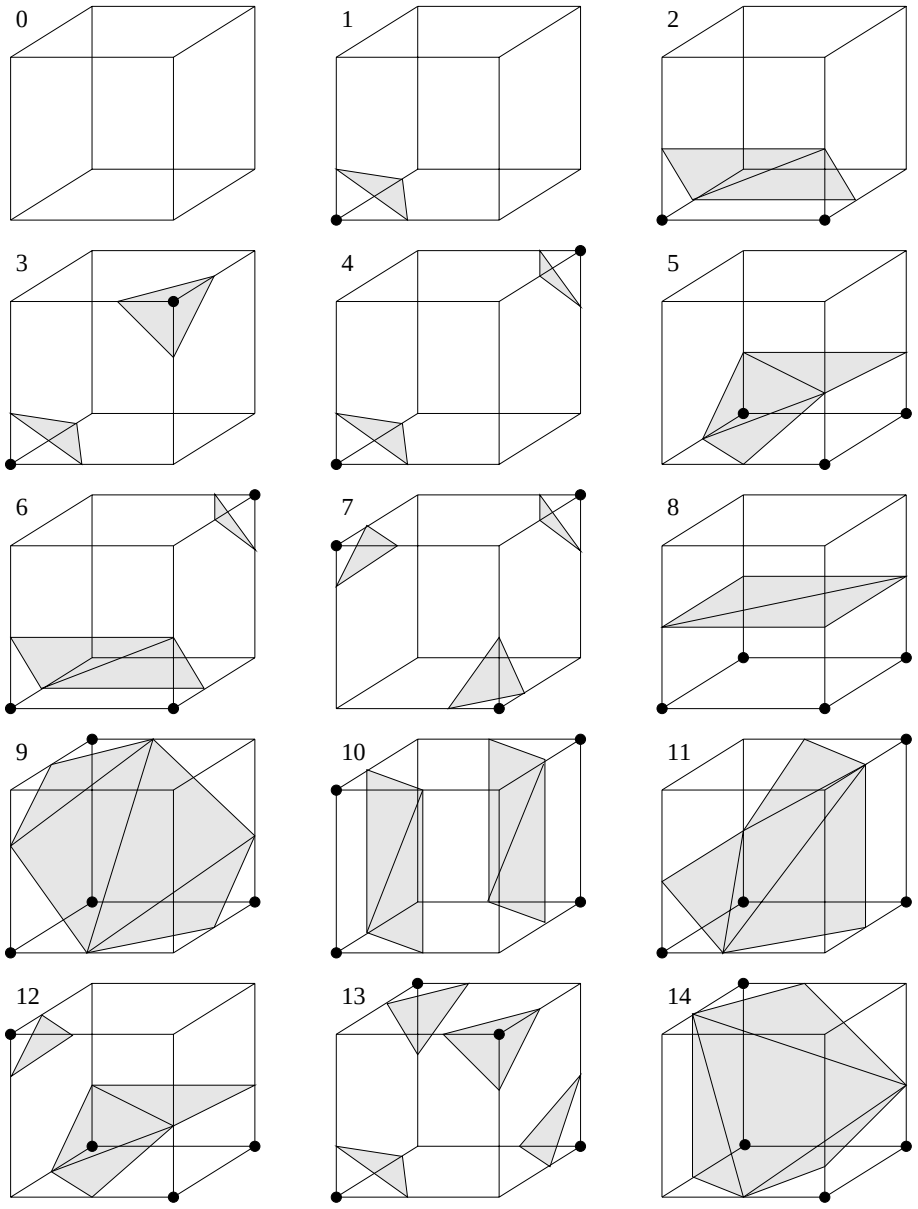


Figure 6.5: Fifteen patterns of triangulated cubes for the marching cubes algorithm.

in a particular level can be used to index directly into the lookup table to produce triangle patches for that cube. However, this approach makes it difficult to maintain the connectivity information between vertices, edges and triangles in the mesh. In the implementation presented in this chapter, a 3D voxel array is first constructed from the octree and the marching cubes algorithm, with a modified lookup table [98] which prevents the creation of holes in the surface [37], is then applied to produce a triangulated mesh. In the original implementation [80] of the marching cubes algorithm, the surface intersection along each edge of a cube is obtained by linear interpolation using the data at the 2 corners. Since the octree only contains binary data, linear interpolation is not necessary and the surface intersection is simply approximated by the midpoint of the edge [97]. In order to reduce the jaggedness in appearance resulting from the midpoint approximation, each vertex in the mesh is smoothed locally by taking the mean position of its directly connected neighboring vertices. The normal vector of each vertex is then taken as the mean of the normal vectors of those triangles which contain that vertex, and the color of the vertex is computed as the weighted average of the color values of its projections on all views. The weighting factor w_i is given by

$$w_i = \begin{cases} -\mathbf{n} \cdot \mathbf{d}_i & (\text{if visible}) \\ 0 & (\text{otherwise}) \end{cases} , \tag{6.1}$$

where $\mathbf{n}$ is the unit normal vector of the vertex (pointing outwards) and $\mathbf{d}_i$ is the unit viewing direction of view i.

6.7 Experiments and Results

The experimental sequence consisted of 19 images of a miniature David statue, of which the first 18 images were taken under unknown circular motion of the statue,

and the last image was taken under unknown general motion (see figure 6.6). The camera motion was estimated from the silhouettes using the algorithms presented in Chapter 5 for circular and general motion, and the resulting camera poses are shown in figure 6.7. An octree was constructed from the silhouettes and the estimated camera motion using the algorithms presented in this chapter. Figure 6.8 shows the resulting octree at different levels, together with the number of cubes in each level. It can be seen from figure 6.8 that the number of cubes grew roughly by a factor of 4 after each level of refinement. This was consistent with the findings of Meagher [92] and Szeliski [124] that the number of cubes is proportional to the surface area of the object measured in units of the finest resolution. Two surface models obtained by applying the marching cubes algorithm to level 7 and level 8 of the octree are shown in figure 6.9 and figure 6.10 respectively. The model extracted from level 7 of the octree had only 27,720 triangles and was suitable for real time rendering, whereas the model extracted from level 8 of the octree was composed of 113,384 triangles and hence had a much higher resolution. The difference in resolution of the 2 surface models can be seen more clearly in figure 6.11, which shows 2 close up views of the 2 models.

Other experimental results on model reconstruction from silhouettes can be found in Chapter 5, and the triangulated meshes of those 3D models are shown in figures 6.12–6.16.

6.8 Discussions

In this chapter, an algorithm for model reconstruction using silhouettes from multiple views is presented. The implementation is based on an octree carving tech-

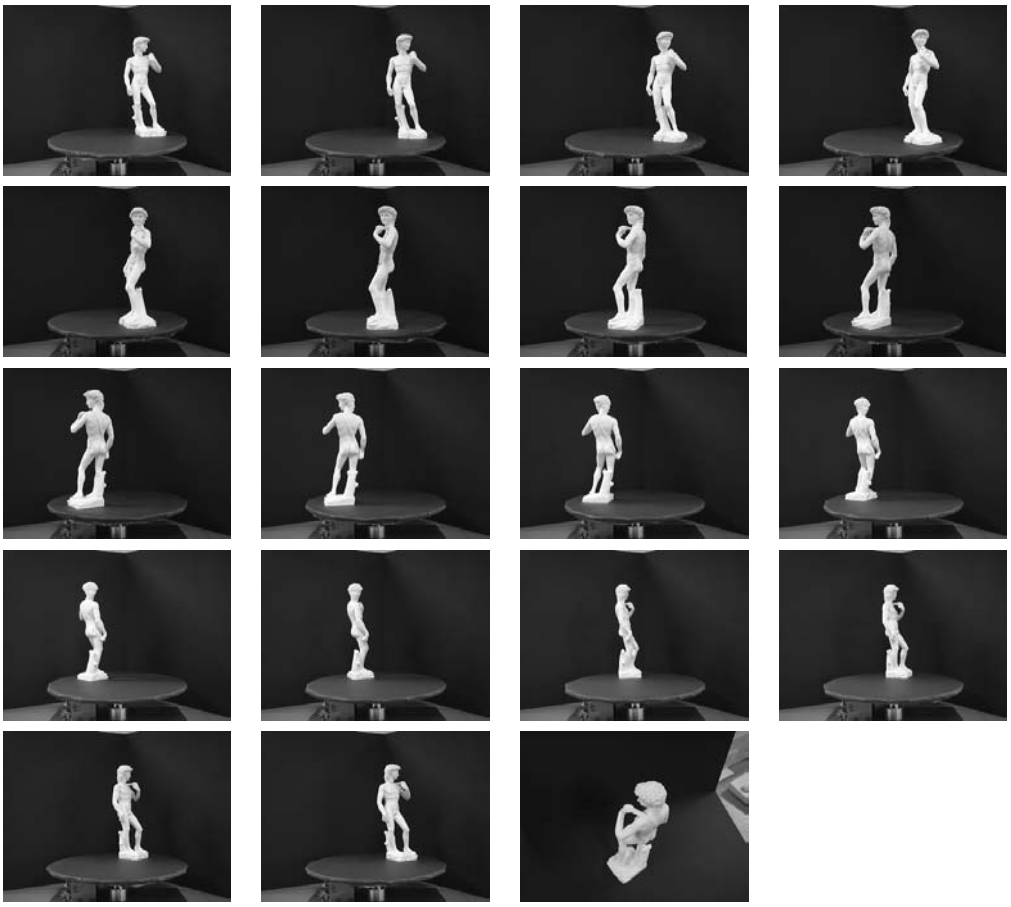


Figure 6.6: Nineteen images of a miniature David statue, of which the first 18 images were taken under unknown circular motion of the statue, and the last image was taken under unknown general motion.

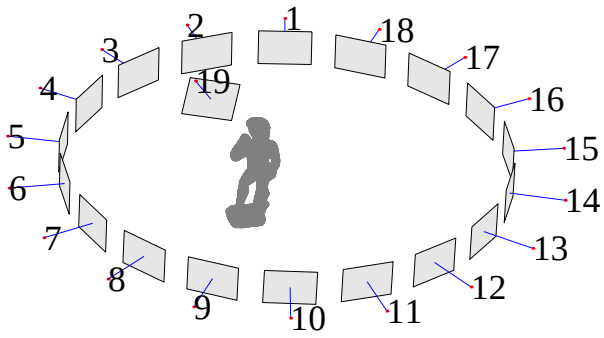


Figure 6.7: Camera poses estimated from the miniature David statue sequence.

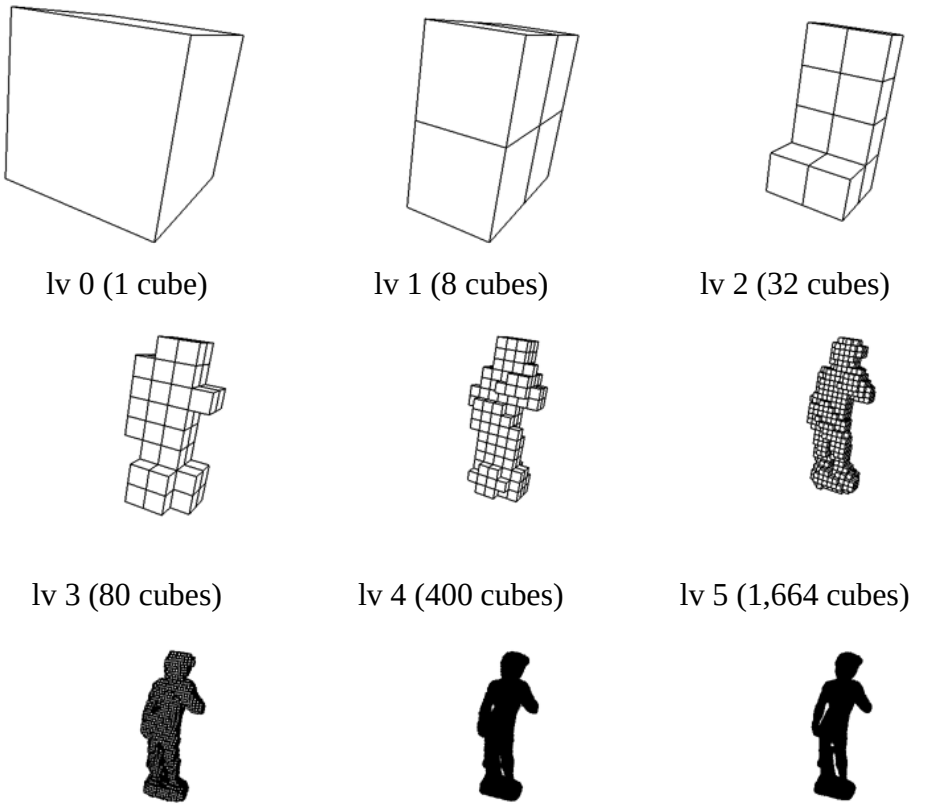
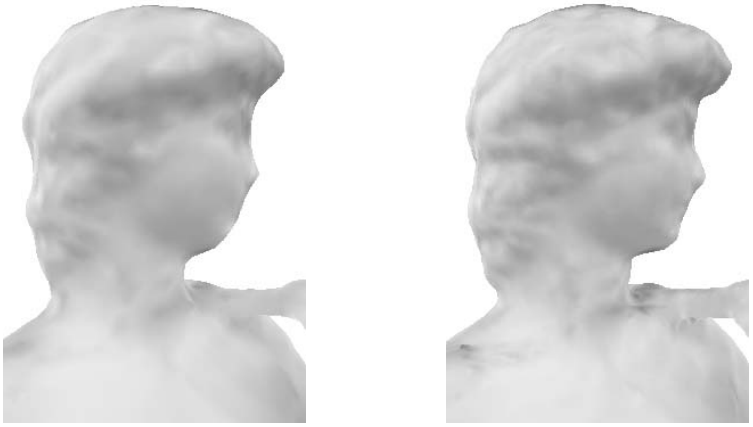


Figure 6.8: An octree constructed from the silhouettes and the estimated camera motion of the miniature David statue sequence. The number of cubes in each level includes all black, gray and white cubes, whereas only gray cubes are drawn.

Figure 6.9: Surface model of the miniature David statue extracted from level 7 of the octree. This model was composed of 27,720 triangles.

Figure 6.10: Surface model of the miniature David statue extracted from level 8 of the octree. This model was composed of 113,384 triangles.



level 7 level 8

Figure 6.11: Two close up views of the surface models shown in figure 6.9 and figure 6.10. It can been seen that the model extracted from level 8 of the octree had a higher resolution and showed more details of the surface.

nique introduced in [124], with a few modifications to make it fit into the framework of the model building system introduced in Chapter 5. In particular, instead of using background subtraction techniques, the object/background binary images for the intersection tests are computed directly from the B-spline snakes, which are used to extract and represent the silhouettes during motion estimation. In addition to the colors of the cubes, the occupancy information of the 8 corners of each cube in the octree has been computed during the octree construction. This information allows a triangulated mesh to be extracted from the octree conveniently using standard marching cubes algorithm. Such a triangulated mesh can then be displayed efficiently with conventional graphics rendering algorithms. Experimental results show that the algorithm is capable of reconstructing objects with relatively complex topologies (like object with holes). Like any other silhouette-based reconstruction technique, the model produced here is only the visual hull [73, 74] of

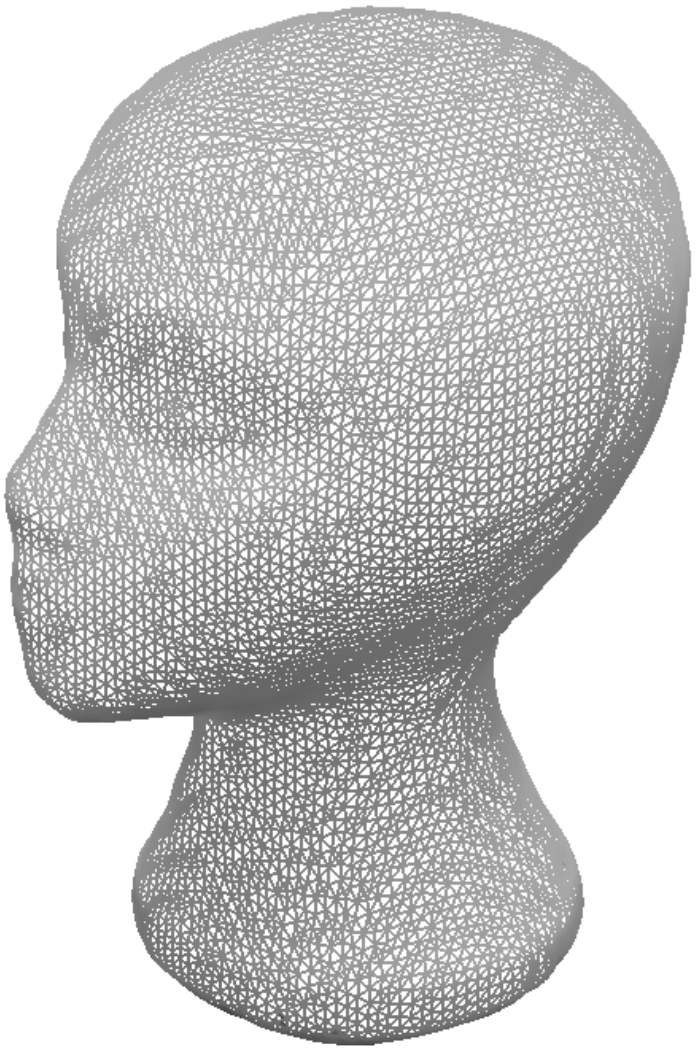


Figure 6.12: Triangulated mesh of the polystyrene head model. This model was composed of 26,696 triangles.

Figure 6.13: Triangulated mesh of the Haniwa model. This model was composed of 12,028 triangles.

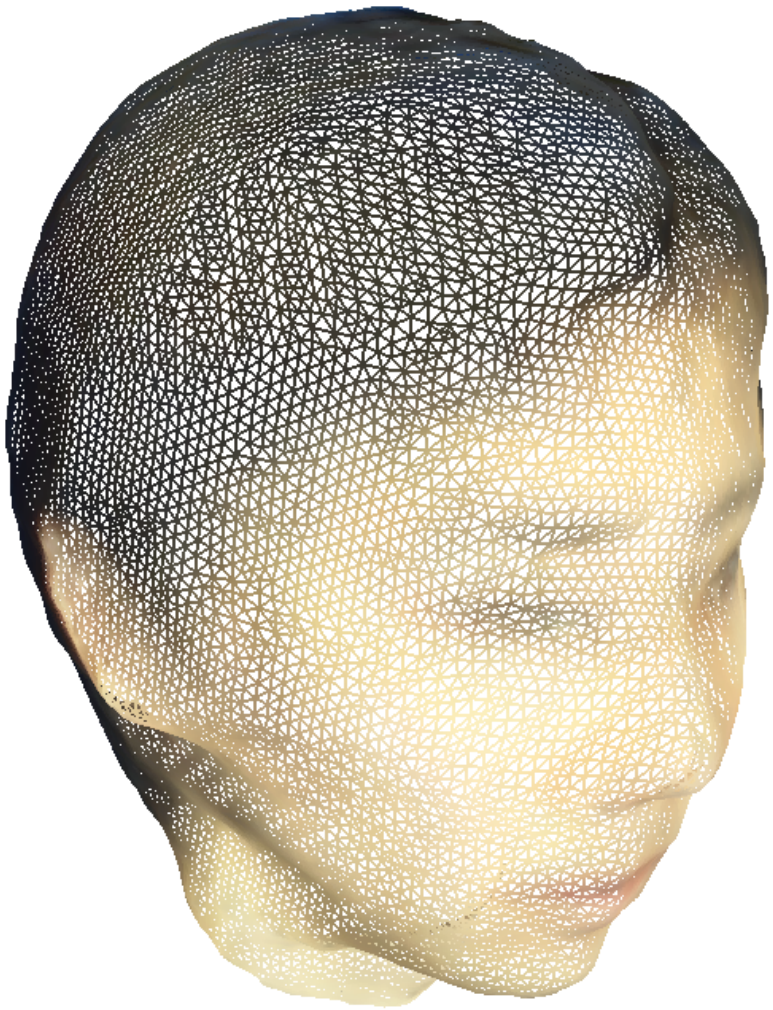


Figure 6.14: Triangulated mesh of the 1^{st} human head model. This model was composed of 34,348 triangles.

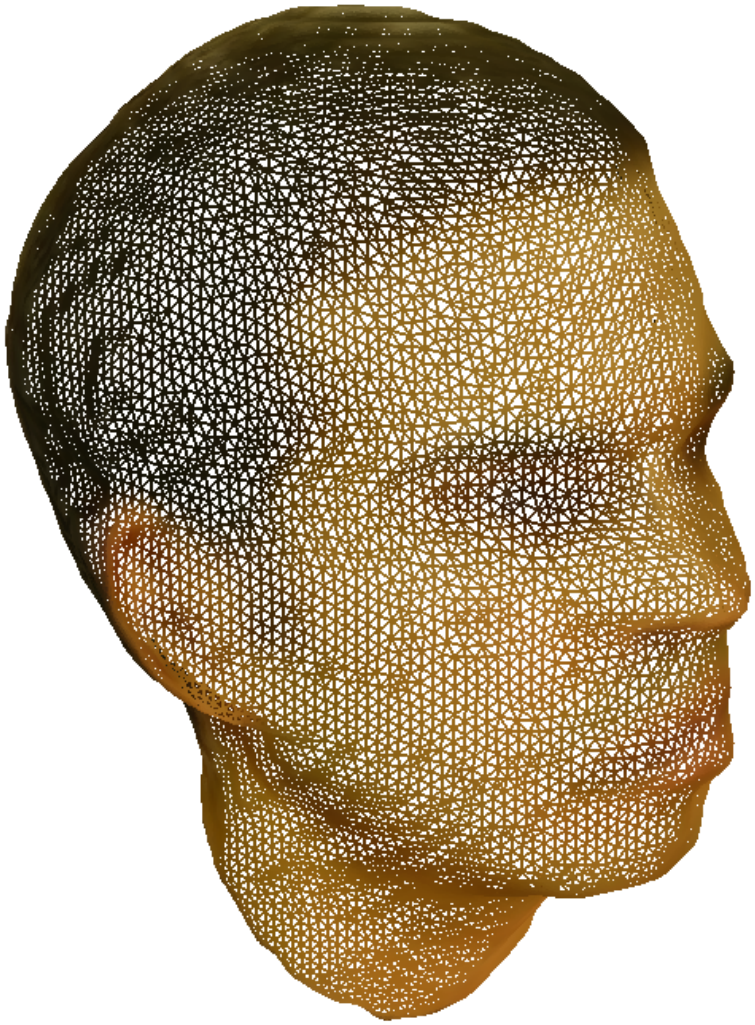


Figure 6.15: Triangulated mesh of the 2^{nd} human head model. This model was composed of 37,404 triangles.

Figure 6.16: Triangulated mesh of the outdoor sculpture model. This model was composed of 29,672 triangles.

the object with respect to the set of viewpoints from which the image sequence is acquired. In order to reconstruct concavities in the object, techniques like *space carving* [72, 14] which exploit texture information should be used. An even better approach is to use both silhouettes and texture information, as proposed by Cross and Zisserman [32]. In spite of that, the system introduced in Chapter 5 aims at recovering the structure and motion of an object from its silhouettes alone. By not depending on texture information, the system is capable of reconstructing any kind of objects, including *smooth* and *textureless* surfaces.

Chapter 7

Conclusions

"*This is not the beginning of the end, but it is the end of the beginning.*"

- Winston Churchill.

7.1 Summary

This thesis has presented theoretical and practical solutions to the problem of structure and motion from silhouettes. Novel algorithms for camera calibration, motion estimation and shape recovery have been developed from the analysis of the projective invariant of surfaces of revolution and the epipolar constraint between the silhouettes of an arbitrary object. Based on these algorithms, a complete, practical and easy-to-use system has been built for generating high quality 3D models from 2D silhouettes. A brief summary of the algorithms and techniques introduced is given below.

In this thesis, the projective invariant of surfaces of revolution has been studied. It has been shown that under perspective projection, the silhouette of a surface of revolution will be invariant to a harmonic homology. Such a harmonic homology can be exploited in 2 ways. First, it has been shown that the axis and the

center of such a harmonic homology are related by the dual image of the absolute conic, and hence they provide 2 constraints on the intrinsic parameters of a camera. Based on this observation, a simple technique for camera calibration has been developed in Chapter 3, which allows a camera to be calibrated from 2 or more silhouettes of surfaces of revolution. Second, the intrinsic parameters of the camera and the harmonic homology can be exploited to rectify the image so that the silhouette becomes bilaterally symmetric about the y-axis. This corresponds to normalizing and rotating the camera until the axis of the surface of revolution lies on the y-z plane of the camera coordinate system. A simple algorithm, based on the coplanarity constraint between the surface normal and the revolution axis, has been developed in Chapter 4 for recovering the 3D shape of a surface of revolution from its rectified silhouette up to an 1-parameter ambiguity. This 1-parameter ambiguity in the reconstruction corresponds to the 1-parameter ambiguity in the orientation of the revolution axis on the y-z plane.

The problem of motion estimation has been tackled in Chapter 5. In the case of circular motion, the 3 main image features, namely the image of the rotation axis, the horizon and a special vanishing point, are fixed throughout the sequence, and the fundamental matrix can be parameterized explicitly in terms of these features. Such a parameterization allows a trivial initialization of the parameters which all bear physical meanings (i.e. image of rotation axis, horizon and rotation angles). It also greatly reduces the dimension of the search space for the optimization problem, which can now be solved using only the 2 outer epipolar tangents. The drawbacks of using circular motion alone for model building are then overcome by the incorporation of arbitrary general views, which reveals information that is concealed under circular motion. It has been shown that the web of contour gen-

erators generated by the circular motion can be exploited to register any arbitrary general view. The coarse model built from the circular motion has been used to aid the initialization of the registration, and again only the 2 outer epipolar tangents are needed for the estimation of the general motion. This 2-stage motion estimation technique avoids the common problems that exist in virtually every algorithm for motion estimation from silhouettes, namely the need for a good but nontrivial initialization, the unrealistic demand for a large number of epipolar tangent points, and the presence of local minima.

Finally, based on an octree carving technique and the marching cubes algorithm, a simple method for constructing a triangulated mesh of the surface from the silhouettes has been described in Chapter 6. Together with the techniques developed for camera calibration and motion estimation, a complete and practical system for generating 3D models from 2D silhouettes has been implemented.

7.2 Future Work

Though the model building system presented in this thesis is very practical and produces high quality 3D models, there are certainly rooms for improvements:

- Surface Reflectance

  In the implementation presented in this thesis, the texture of the model has been computed using an ad hoc method. In order to produce a photo-realistic 3D model under different lighting conditions, it would be desirable to develop algorithms for recovering the surface reflectance of the model.

- Surface Representation

  The marching cubes algorithm has been employed to extract a triangulated

mesh from the octree. Despite its extensive use in many applications, the marching cubes algorithm often produces excessive output data fragmentation which prevents interactive rendering. A more efficient representation, especially for smooth surfaces, is required.

- Viewpoint Control

  The coarse 3D model built from the circular motion contains information about which part of the object has not yet been fully explored. This information can be used to develop strategies to determine the viewpoints of the new arbitrary views for model refinement.

- Fusion of Information

  The work described in this thesis has only used information from the silhouettes to solve the structure and motion problem. Nonetheless, other image features like corners, edges, shadows, textures and specularities also provide strong cues to surface shape and orientation. An ideal approach would be to design a system that exploits all the information available to provide a robust solution to the structure and motion problem.

- Self-Calibration

  The motion estimation algorithms presented in this work depend on off-line camera calibration. It would be desirable if self-calibration techniques can be incorporated into the algorithms.

Appendix A

Definition of the Harmonic Homology

A *perspective collineation* [29], with center $\mathbf{x}_c$ and axis $\mathbf{l}_a$, is a collineation which leaves all the lines through $\mathbf{x}_c$ and points of $\mathbf{l}_a$ invariant. If the center $\mathbf{x}_c$ and the axis $\mathbf{l}_a$ are not incident, the perspective collineation is called a *homology* [29]; otherwise it is called an *elation* [29]. Consider a point $\mathbf{x}$ which is mapped by a homology with center $\mathbf{x}_c$ and axis $\mathbf{l}_a$ to the point $\mathbf{x}'$. Let $\mathbf{x}'_c$ be the point of intersection between the axis $\mathbf{l}_a$ and a line passing through the points $\mathbf{x}$ and $\mathbf{x}'$. The homology is said to be harmonic if the points $\mathbf{x}$ and $\mathbf{x}'$ are harmonic conjugates with respect to $\mathbf{x}_c$ and $\mathbf{x}'_c$ (i.e. the cross-ratio $\{\mathbf{x}_c,\mathbf{x}'_c;\mathbf{x},\mathbf{x}'\}$ equals -1). The matrix $\mathbf{W}$ representing a *harmonic homology* [29] with centre $\mathbf{x}_c$ and axis $\mathbf{l}_a$, in homogeneous coordinates, is given by

$$\mathbf{W} = \mathbb{I}_3 - 2\frac{\mathbf{x}_c\mathbf{l}_a^T}{\mathbf{x}_c^T\mathbf{l}_a}. \tag{A.1}$$

More details on harmonic homology can be found in [117, 29].

Appendix B

Bilateral Symmetry and Surfaces of Revolution

Let $\tilde{\mathbf{C}}_{\mathrm{r}}(s) = [r(s)\ y(s)\ 0]^{\mathrm{T}}$ be a regular planar curve on the x-y plane where $\exists r_{\max}$ such that $0 < r(s) < r_{\max}\ \forall s$. A surface of revolution can be generated by rotating $\tilde{\mathbf{C}}_{\mathrm{r}}$ about the y-axis, and is given by

$$\tilde{\mathbf{S}}_{\mathrm{r}}(s,\theta) = \begin{bmatrix} r(s)\cos\theta \\ y(s) \\ r(s)\sin\theta \end{bmatrix}, \tag{B.1}$$

where θ is the angle parameter for a complete circle. The tangent plane basis vectors are given by

$$\frac{\partial\tilde{\mathbf{S}}_{\mathrm{r}}}{\partial s} = \begin{bmatrix} \dot{r}(s)\cos\theta \\ \dot{y}(s) \\ \dot{r}(s)\sin\theta \end{bmatrix} \text{ and } \frac{\partial\tilde{\mathbf{S}}_{\mathrm{r}}}{\partial\theta} = \begin{bmatrix} -r(s)\sin\theta \\ 0 \\ r(s)\cos\theta \end{bmatrix} \tag{B.2}$$

respectively. Since the surface normal must be orthogonal to both tangent plane basis vectors, it is thus given by

$$\begin{aligned} \mathbf{n}(s,\theta) &= \frac{\partial\tilde{\mathbf{S}}_{\mathrm{r}}}{\partial s} \times \frac{\partial\tilde{\mathbf{S}}_{\mathrm{r}}}{\partial\theta} \\ &= \begin{bmatrix} r(s)\dot{y}(s)\cos\theta \\ -r(s)\dot{r}(s) \\ r(s)\dot{y}(s)\sin\theta \end{bmatrix}. \end{aligned} \tag{B.3}$$

Consider now a pin-hole camera $\hat{\mathbf{P}} = [\mathbb{I}_3 \ \mathbf{t}]$, where $\mathbf{t} = [0 \ 0 \ d_z]$ and $d_z > r_{\max}$. The silhouette $\hat{\rho}$ of $\tilde{\mathbf{S}}_\mathrm{r}$ formed on the image plane of $\hat{\mathbf{P}}$ is the projection of the locus of points on $\tilde{\mathbf{S}}_\mathrm{r}$ at which the line of sight is orthogonal to the surface normal. This constraint can be expressed as

$$\begin{aligned}
(\tilde{\mathbf{S}}_\mathrm{r}(s,\theta) + \mathbf{t}) \cdot \mathbf{n}(s,\theta) &= 0 \\
\begin{bmatrix} r(s)\cos\theta \\ y(s) \\ r(s)\sin\theta + d_z \end{bmatrix} \cdot \begin{bmatrix} r(s)\dot{y}(s)\cos\theta \\ -r(s)\dot{r}(s) \\ r(s)\dot{y}(s)\sin\theta \end{bmatrix} &= 0 \\
r(s)\dot{y}(s) - \dot{r}(s)y(s) + d_z\dot{y}(s)\sin\theta &= 0 \\
\frac{\dot{r}(s)y(s) - r(s)\dot{y}(s)}{d_z\dot{y}(s)} &= \sin\theta. \qquad \text{(B.4)}
\end{aligned}$$

Now by projecting $\tilde{\mathbf{S}}_\mathrm{r}$ using $\hat{\mathbf{P}}$, the image of $\tilde{\mathbf{S}}_\mathrm{r}$, in homogeneous coordinates, is given by

$$\begin{aligned}
\hat{\mathbf{s}}_\mathrm{r} &= \hat{\mathbf{P}}[\tilde{\mathbf{S}}_\mathrm{r}^\mathrm{T} \ 1]^\mathrm{T} \\
&= \begin{bmatrix} r(s)\cos\theta \\ y(s) \\ r(s)\sin\theta + d_z \end{bmatrix}. \qquad \text{(B.5)}
\end{aligned}$$

Finally, by removing the dependency of θ from $\hat{\mathbf{s}}_\mathrm{r}$ using (B.4), the silhouette is then given by

$$\hat{\rho}(s) = \begin{bmatrix} \pm r(s)\sqrt{1 - \left(\frac{\dot{r}(s)y(s) - r(s)\dot{y}(s)}{d_z\dot{y}(s)}\right)^2} \\ y(s) \\ r(s)\frac{\dot{r}(s)y(s) - r(s)\dot{y}(s)}{d_z\dot{y}(s)} + d_z \end{bmatrix}. \qquad \text{(B.6)}$$

It follows from equation (B.6) that the silhouette of $\tilde{\mathbf{S}}_\mathrm{r}$, formed on the image plane of $\hat{\mathbf{P}}$, is bilaterally symmetric about the image of the revolution axis $\hat{\mathbf{l}}_\mathrm{s} = [1 \ 0 \ 0]^\mathrm{T}$ (i.e. the y-axis in the image).

Appendix C

Ambiguity in Reconstruction of Surfaces of Revolution

Consider a surface of revolution $\mathbf{S}_{\mathrm{r}}$ whose axis of revolution coincides with the y-axis, and a pin-hole camera $\hat{\mathbf{P}} = [\mathbb{I}_3 \ \mathbf{t}]$ where $\mathbf{t} = [0 \ 0 \ d_z]^{\mathrm{T}}$ and $d_z > 0$. The silhouette $\hat{\rho}$ of $\mathbf{S}_{\mathrm{r}}$, formed on the image plane of $\hat{\mathbf{P}}$, will be bilaterally symmetric about the image of the revolution axis $\hat{\mathbf{l}}_{\mathrm{s}} = [1 \ 0 \ 0]^{\mathrm{T}}$ (see Appendix B) and invariant to the harmonic homology $\mathbf{T}$, given by

$$\mathbf{T} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{C.1}$$

Given a point $\hat{\mathbf{x}}(s) = [x(s) \ y(s) \ 1]^{\mathrm{T}}$ in the silhouette $\hat{\rho}$, its associated surface normal $\mathbf{n}(s)$ is given by

$$\mathbf{n}(s) = \begin{bmatrix} -\dot{y}(s) \\ \dot{x}(s) \\ x(s)\dot{y}(s) - \dot{x}(s)y(s) \end{bmatrix} = \begin{bmatrix} n_1(s) \\ n_2(s) \\ n_3(s) \end{bmatrix}, \tag{C.2}$$

and its depth $\lambda(s)$ along the optical axis is given by

$$\lambda(s) = \frac{d_z n_1(s)}{n_1(s) - n_3(s)x(s)} \tag{C.3}$$

(see Chapter 4 for details).

Consider now a pin-hole camera $\mathbf{P}^{\psi}$ obtained by rotating $\hat{\mathbf{P}}$ about its x-axis by an angle $-\psi$. Hence $\mathbf{P}^{\psi} = \mathbf{R}_x(\psi)\hat{\mathbf{P}}$, where

$$\mathbf{R}_x(\psi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\psi & -\sin\psi \\ 0 & \sin\psi & \cos\psi \end{bmatrix} = \begin{bmatrix} \mathbf{r}_1^{\mathrm{T}} \\ \mathbf{r}_2^{\mathrm{T}} \\ \mathbf{r}_3^{\mathrm{T}} \end{bmatrix}. \tag{C.4}$$

The silhouette ρ^{ψ} of $\mathbf{S}_{\mathrm{r}}$, formed on the image plane of $\mathbf{P}^{\psi}$, can be obtained by transforming every point in $\hat{\rho}$ by $\mathbf{R}_x(\psi)$ (i.e. $\rho^{\psi} = \mathbf{R}_{\mathbf{x}}(\psi)\hat{\rho}$). Let $\hat{\mathbf{x}}$ and $\hat{\mathbf{x}}'$ be a pair of symmetric points in $\hat{\rho}$, and $\mathbf{x}^{\psi} = \mathbf{R}_x(\psi)\hat{\mathbf{x}}$ and $\mathbf{x}'^{\psi} = \mathbf{R}_x(\psi)\hat{\mathbf{x}}'$ be their correspondences in ρ^{ψ}. The symmetry between $\hat{\mathbf{x}}$ and $\hat{\mathbf{x}}'$ is given by

$$\hat{\mathbf{x}}' = \mathbf{T}\hat{\mathbf{x}}. \tag{C.5}$$

Substituting $\hat{\mathbf{x}}$ and $\hat{\mathbf{x}}'$ in (C.5) by $\mathbf{R}_x^{-1}(\psi)\mathbf{x}^{\psi}$ and $\mathbf{R}_x^{-1}(\psi)\mathbf{x}'^{\psi}$, respectively, gives

$$\begin{aligned}
\mathbf{R}_x^{-1}(\psi)\mathbf{x}'^{\psi} &= \mathbf{T}\mathbf{R}_x^{-1}(\psi)\mathbf{x}^{\psi} \\
\mathbf{x}'^{\psi} &= \mathbf{R}_x(\psi)\mathbf{T}\mathbf{R}_x^{-1}(\psi)\mathbf{x}^{\psi} \\
&= \begin{bmatrix} -1 & 0 & 0 \\ 0 & \cos\psi & -\sin\psi \\ 0 & \sin\psi & \cos\psi \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\psi & \sin\psi \\ 0 & -\sin\psi & \cos\psi \end{bmatrix} \mathbf{x}^{\psi} \\
&= \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \mathbf{x}^{\psi} \\
&= \mathbf{T}\mathbf{x}^{\psi}.
\end{aligned} \tag{C.6}$$

Equation (C.6) implies that the silhouette ρ^{ψ} is also invariant to $\mathbf{T}$. As a result, given a silhouette ρ of a surface of revolution $\mathbf{S}_{\mathrm{r}}$ which is invariant to $\mathbf{T}$, one can only infer that the revolution axis of $\mathbf{S}_{\mathrm{r}}$ lies on the y-z plane of the camera coordinate system. However, the relative orientation of the revolution axis with respect to the y-axis of the camera cannot be deduced. This results in an 1-parameter ambiguity in the reconstruction of the surface of revolution by applying equation (C.3) to the rectified silhouette of $\mathbf{S}_{\mathrm{r}}$ that is invariant to $\mathbf{T}$. Consider again the

point $\hat{\mathbf{x}}(s)$ in $\hat{\rho}$, it is transformed by $\mathbf{R}_x(\psi)$ to the point $\mathbf{x}^\psi(s)$ in ρ^ψ, given by

$$\mathbf{x}^\psi(s) = \frac{\mathbf{R}_x(\psi)\hat{\mathbf{x}}(s)}{\mathbf{r}_3^{\mathrm{T}}\hat{\mathbf{x}}(s)}. \tag{C.7}$$

The denominator $\mathbf{r}_3^{\mathrm{T}}\hat{\mathbf{x}}(s)$ in equation (C.7) is used to normalize $\mathbf{x}^\psi(s)$ so that its 3^{rd} coefficient is 1. The surface normal associated with $\mathbf{x}^\psi(s)$ can be obtained by transforming $\mathbf{n}(s)$ by $\mathbf{R}_x(\psi)$, and is given by

$$\mathbf{n}^\psi(s) = \mathbf{R}_x(\psi)\mathbf{n}(s). \tag{C.8}$$

Substituting (C.7) and (C.8) into (C.3) yields

$$\lambda^\psi(s) = \frac{d_z\mathbf{r}_1^{\mathrm{T}}\mathbf{n}(s)}{\mathbf{r}_1^{\mathrm{T}}\mathbf{n}(s) - \mathbf{r}_3^{\mathrm{T}}\mathbf{n}(s)\frac{\mathbf{r}_1^{\mathrm{T}}\hat{\mathbf{x}}(s)}{\mathbf{r}_3^{\mathrm{T}}\hat{\mathbf{x}}(s)}}, \tag{C.9}$$

and the resulting contour generator Γ^ψ, with the 1-parameter ambiguity in ψ, is then given by

$$\begin{aligned}
\Gamma^\psi(s) &= \begin{bmatrix} -\mathbf{t} + \lambda^\psi(s)\mathbf{x}^\psi(s) \\ 1 \end{bmatrix} \\
&= \begin{bmatrix} -\mathbf{t} + \frac{d_z\mathbf{r}_1^{\mathrm{T}}\mathbf{n}(s)}{\mathbf{r}_1^{\mathrm{T}}\mathbf{n}(s) - \mathbf{r}_3^{\mathrm{T}}\mathbf{n}(s)\frac{\mathbf{r}_1^{\mathrm{T}}\hat{\mathbf{x}}(s)}{\mathbf{r}_3^{\mathrm{T}}\hat{\mathbf{x}}(s)}}\frac{\mathbf{R}_x(\psi)\hat{\mathbf{x}}(s)}{\mathbf{r}_3^{\mathrm{T}}\hat{\mathbf{x}}(s)} \\ 1 \end{bmatrix} \\
&= \begin{bmatrix} -\mathbf{t} + \frac{d_z n_1(s)}{\mathbf{r}_3^{\mathrm{T}}\hat{\mathbf{x}}(s)n_1(s) - \mathbf{r}_3^{\mathrm{T}}\mathbf{n}(s)x(s)}\mathbf{R}_x(\psi)\hat{\mathbf{x}}(s) \\ 1 \end{bmatrix} \\
&= \begin{bmatrix} d_z\dot{y}(s)x(s) \\ d_z\dot{y}(s)(y(s)\cos\psi - \sin\psi) \\ d_z\alpha_\Gamma^\psi(s) \\ \dot{y}(s)(y(s)\sin\psi + \cos\psi) - \alpha_\Gamma^\psi(s) \end{bmatrix},
\end{aligned} \tag{C.10}$$

where $\alpha_\Gamma^\psi(s) = \{(\dot{x}(s)y(s) - x(s)\dot{y}(s))\cos\psi - \dot{x}(s)\sin\psi\}x(s)$.

Appendix D

Estimation of the Orientation of the Revolution Axis

Consider a surface of revolution $\mathbf{S}_{\mathrm{r}}$ whose revolution axis lies on the y-z plane of a pin-hole camera $\hat{\mathbf{P}} = [\mathbb{I}_3 \ \mathbb{O}_3]$. In general, a latitude circle C in $\mathbf{S}_{\mathrm{r}}$ will be projected onto the image plane of $\hat{\mathbf{P}}$ as an ellipse which is bilaterally symmetric about the y-axis. Such an ellipse can be represented by a 3×3 symmetric matrix $\hat{\mathbf{C}}$, given by

$$\hat{\mathbf{C}} = \begin{bmatrix} e & 0 & 0 \\ 0 & f & g \\ 0 & g & h \end{bmatrix}, \tag{D.1}$$

such that every point $\mathbf{x}$ on the ellipse satisfies $\mathbf{x}^{\mathrm{T}}\hat{\mathbf{C}}\mathbf{x} = 0$. Consider now a pin-hole camera $\mathbf{P}$ obtained by rotating $\hat{\mathbf{P}}$ about its x-axis by an angle $-\sigma$. Hence $\mathbf{P} = \mathbf{R}_x(\sigma)\hat{\mathbf{P}}$, where

$$\mathbf{R}_x(\sigma) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\sigma & -\sin\sigma \\ 0 & \sin\sigma & \cos\sigma \end{bmatrix}. \tag{D.2}$$

The image $\mathbf{C}$ of the latitude circle C, formed on the image plane of $\mathbf{P}$, can be obtained by transforming the ellipse $\hat{\mathbf{C}}$ by $\mathbf{R}_x(\sigma)$, and is given by

$$\begin{aligned}\mathbf{C} &= \mathbf{R}_x^{-\mathrm{T}}(\sigma)\hat{\mathbf{C}}\mathbf{R}_x^{-1}(\sigma) \\ &= \mathbf{R}_x(\sigma)\hat{\mathbf{C}}\mathbf{R}_x^{\mathrm{T}}(\sigma) \\ &= \begin{bmatrix} e & 0 & 0 \\ 0 & fc^2 - 2gsc + hs^2 & fsc + g(c^2 - s^2) - hsc \\ 0 & fsc + g(c^2 - s^2) - hsc & fs^2 + 2gsc + hc^2 \end{bmatrix}, \quad \text{(D.3)}\end{aligned}$$

where $c = \cos\sigma$ and $s = \sin\sigma$. If the revolution axis of $\mathbf{S}_\mathrm{r}$ is parallel to the optical axis (i.e. z-axis) of $\mathbf{P}$, then the image $\mathbf{C}$ of the latitude circle C will be a circle, i.e.

$$e = f\cos^2\sigma - 2g\sin\sigma\cos\sigma + h\sin^2\sigma. \quad \text{(D.4)}$$

Hence by locating and fitting an ellipse to the image of a latitude circle in $\mathbf{S}_\mathrm{r}$, the angle σ can be obtained by solving equation (D.4) and the orientation of the revolution axis of $\mathbf{S}_\mathrm{r}$ follows. Note that equation (D.4) is quadratic in $\sin\sigma$ and $\cos\sigma$, and hence in general there will be 2 distinct solutions of which only one is correct. Such an ambiguity originates from the symmetry of the ellipse, and can be resolved either manually or by fitting 2 ellipses to the images of 2 distinct latitude circles which in general share only one common solution for σ.

Appendix E

Projective Transformations and Surfaces of Revolution

Consider a 4×4 nonsingular matrix

$$\mathbf{H}_{\mathrm{SOR}} = \begin{bmatrix} h_{11} & h_{12} & h_{13} & h_{14} \\ h_{21} & h_{22} & h_{23} & h_{24} \\ h_{31} & h_{32} & h_{33} & h_{34} \\ h_{41} & h_{42} & h_{43} & h_{44} \end{bmatrix} \tag{E.1}$$

representing a projective transformation that maps a surface of revolution

$$\mathbf{S}_{\mathrm{r}}(s, \theta) = \begin{bmatrix} r(s) \cos\theta \\ y(s) \\ r(s) \sin\theta \\ 1 \end{bmatrix} \tag{E.2}$$

to another surface of revolution

$$\mathbf{S}'_{\mathrm{r}}(s', \theta') = \begin{bmatrix} r'(s') \cos\theta' \\ y'(s') \\ r'(s') \sin\theta' \\ 1 \end{bmatrix}, \tag{E.3}$$

both with the y-axis as their axes of revolution. Note that $\mathbf{H}_{\mathrm{SOR}}$ has the property that it maps a latitude circle of a surface of revolution to a latitude circle of another surface of revolution, as a latitude circle is by itself a surface of revolution in the limiting case.

The projective transformation $\mathbf{S}'_{\mathrm{r}}(s',\theta') = \mathbf{H}_{\mathrm{SOR}}\mathbf{S}_{\mathrm{r}}(s,\theta)$ is represented, in Cartesian coordinates, by the set of equations

$$r'(s')\cos\theta' = \frac{r(s)h_{11}\cos\theta + h_{12}y + r(s)h_{13}\sin\theta + h_{14}}{r(s)h_{41}\cos\theta + h_{42}y + r(s)h_{43}\sin\theta + h_{44}}, \quad \text{(E.4)}$$

$$y'(s') = \frac{r(s)h_{21}\cos\theta + h_{22}y + r(s)h_{23}\sin\theta + h_{24}}{r(s)h_{41}\cos\theta + h_{42}y + r(s)h_{43}\sin\theta + h_{44}}, \quad \text{(E.5)}$$

$$r'(s')\sin\theta' = \frac{r(s)h_{31}\cos\theta + h_{32}y + r(s)h_{33}\sin\theta + h_{34}}{r(s)h_{41}\cos\theta + h_{42}y + r(s)h_{43}\sin\theta + h_{44}}. \quad \text{(E.6)}$$

Since $\mathbf{H}_{\mathrm{SOR}}$ maps a latitude circle to a latitude circle, $y'(s')$ should therefore be independent of θ (i.e. $\frac{d\,y'(s')}{d\,\theta} = 0$). Hence differentiating (E.5) with respect to θ gives

$$\begin{aligned} 0 = {} & (h_{23}h_{41} - h_{21}h_{43})r(s) + \\ & (h_{22}h_{41} - h_{21}h_{42})y(s)\sin\theta + (h_{23}h_{42} - h_{22}h_{43})y(s)\cos\theta + \\ & (h_{24}h_{41} - h_{21}h_{44})\sin\theta + (h_{23}h_{44} - h_{24}h_{43})\cos\theta, \end{aligned} \quad \text{(E.7)}$$

which holds for all values of $r(s)$, $y(s)$ and θ. Equation (E.7) thus yields the following 5 constraints,

$$h_{23}h_{41} - h_{21}h_{43} = 0, \quad \text{(E.8)}$$

$$h_{22}h_{41} - h_{21}h_{42} = 0, \quad \text{(E.9)}$$

$$h_{23}h_{42} - h_{22}h_{43} = 0, \quad \text{(E.10)}$$

$$h_{24}h_{41} - h_{21}h_{44} = 0, \text{ and} \quad \text{(E.11)}$$

$$h_{23}h_{44} - h_{24}h_{43} = 0. \quad \text{(E.12)}$$

Consider now the sum of the squares of (E.4) and (E.6),

$$\begin{aligned} r'(s')^2 = {} & \frac{(r(s)h_{11}\cos\theta + h_{12}y + r(s)h_{13}\sin\theta + h_{14})^2}{(r(s)h_{41}\cos\theta + h_{42}y + r(s)h_{43}\sin\theta + h_{44})^2} + \\ & \frac{(r(s)h_{31}\cos\theta + h_{32}y + r(s)h_{33}\sin\theta + h_{34})^2}{(r(s)h_{41}\cos\theta + h_{42}y + r(s)h_{43}\sin\theta + h_{44})^2}. \end{aligned} \quad \text{(E.13)}$$

Since the transformation $\mathbf{H}_{\mathrm{SOR}}$ is a point-to-point mapping, $r'(s')$ must be zero whenever $r(s)$ is zero. Substituting $r(s) = 0$ and $r'(s') = 0$ into (E.13) gives

$$0 = \frac{(h_{12}y(s) + h_{14})^2 + (h_{32}y(s) + h_{34})^2}{(h_{42}y(s) + h_{44})^2}. \tag{E.14}$$

Equation (E.14) yields 2 constraints,

$$h_{12}y(s) + h_{14} = 0, \text{ and} \tag{E.15}$$

$$h_{32}y(s) + h_{34} = 0, \tag{E.16}$$

which hold for all values of $y(s)$. Equations (E.15) and (E.16) imply that

$$h_{12} = h_{14} = h_{32} = h_{34} = 0, \tag{E.17}$$

and the projective transformation $\mathbf{H}_{\mathrm{SOR}}$ thus has the form

$$\mathbf{H}_{\mathrm{SOR}} = \begin{bmatrix} h_{11} & 0 & h_{13} & 0 \\ h_{21} & h_{22} & h_{23} & h_{24} \\ h_{31} & 0 & h_{33} & 0 \\ h_{41} & h_{42} & h_{43} & h_{44} \end{bmatrix}. \tag{E.18}$$

Substituting (E.17) into (E.13) gives

$$\begin{aligned} r'(s')^2 = {} & \frac{(r(s)h_{11}\cos\theta + r(s)h_{13}\sin\theta)^2}{(r(s)h_{41}\cos\theta + h_{42}y + r(s)h_{43}\sin\theta + h_{44})^2} + \\ & \frac{(r(s)h_{31}\cos\theta + r(s)h_{33}\sin\theta)^2}{(r(s)h_{41}\cos\theta + h_{42}y + r(s)h_{43}\sin\theta + h_{44})^2}. \end{aligned} \tag{E.19}$$

Without loss of generality, let

$$\begin{array}{rclcrcl} h_{11} & = & \mathcal{A}\cos\varrho & \text{and} & h_{13} & = & \mathcal{A}\sin\varrho, \\ h_{31} & = & -\mathcal{B}\sin\varphi & \text{and} & h_{33} & = & \mathcal{B}\cos\varphi, \\ h_{41} & = & \mathcal{C}\cos\vartheta & \text{and} & h_{43} & = & \mathcal{C}\sin\vartheta, \end{array} \tag{E.20}$$

where $\mathcal{A} \neq 0$ and $\mathcal{B} \neq 0$. Substituting (E.20) into (E.19) gives

$$r'(s')^2 = r(s)^2 \frac{\mathcal{A}^2\cos^2(\theta - \varrho) + \mathcal{B}^2\sin^2(\theta - \varphi)}{(\mathcal{C}r(s)\cos(\theta - \vartheta) + h_{42}y(s) + h_{44})^2}. \tag{E.21}$$

Since $r'(s')$ should be independent of θ (i.e. $\frac{d\, r'(s')}{d\, \theta} = 0$), hence differentiating (E.21) with respect to θ gives

$$\begin{aligned} 0 &= 2\mathcal{A}^2\mathcal{C}r(s)\sin(\varrho-\vartheta)\cos(\theta-\varrho) + \\ &\quad 2\mathcal{B}^2\mathcal{C}r(s)\sin(\theta-\varphi)\cos(\varphi-\vartheta) + \\ &\quad [\mathcal{B}^2\sin(2\theta-2\varphi) - \mathcal{A}^2\sin(2\theta-2\varrho)](h_{42}y(s)+h_{44}), \end{aligned} \tag{E.22}$$

which holds for all values of $r(s)$, $y(s)$ and θ. Equation (E.22) yields 2 constraints,

$$\mathcal{C}\left[\mathcal{A}^2\sin(\varrho-\vartheta)\cos(\theta-\varrho) + \mathcal{B}^2\sin(\theta-\varphi)\cos(\varphi-\vartheta)\right] = 0, \tag{E.23}$$

$$\left[\mathcal{B}^2\sin(2\theta-2\varphi) - \mathcal{A}^2\sin(2\theta-2\varrho)\right](h_{42}y(s)+h_{44}) = 0. \tag{E.24}$$

Solving equation (E.23) yields 2 possible cases:

- case i

$$\begin{aligned} \varrho &= \vartheta + k_1\pi, \qquad \text{and} \\ \varphi &= \vartheta + \tfrac{\pi}{2} + k_2\pi, \end{aligned} \tag{E.25}$$

  where k_1 and k_2 are any integers. Solving equation (E.24) then gives

$$h_{42} = h_{44} = 0. \tag{E.26}$$

  The projective transformation $\mathbf{H}_{\mathrm{SOR}}$ thus has the form

$$\mathbf{H}_{\mathrm{SOR}} = \begin{bmatrix} h_{11} & 0 & h_{13} & 0 \\ h_{21} & h_{22} & h_{23} & h_{24} \\ h_{31} & 0 & h_{33} & 0 \\ h_{41} & 0 & h_{43} & 0 \end{bmatrix}. \tag{E.27}$$

  For $\mathbf{H}_{\mathrm{SOR}}$ to be nonsingular, obviously both h_{22} and h_{24} cannot be zeros. However, it then follows from equations (E.9)–(E.12) that $h_{41} = h_{43} = 0$, causing $\mathbf{H}_{\mathrm{SOR}}$ to be singular.

- case ii

$$\mathcal{C} = 0. \tag{E.28}$$

Substituting (E.28) into (E.20) gives

$$h_{41} = h_{43} = 0. \tag{E.29}$$

The projective transformation $\mathbf{H}_{\mathrm{SOR}}$ thus has the form

$$\mathbf{H}_{\mathrm{SOR}} = \begin{bmatrix} h_{11} & 0 & h_{13} & 0 \\ h_{21} & h_{22} & h_{23} & h_{24} \\ h_{31} & 0 & h_{33} & 0 \\ 0 & h_{42} & 0 & h_{44} \end{bmatrix}. \tag{E.30}$$

For $\mathbf{H}_{\mathrm{SOR}}$ to be nonsingular, h_{42} and h_{44} cannot be both zeros at the same time. It then follows from equations (E.9)–(E.12) that

$$h_{21} = h_{23} = 0. \tag{E.31}$$

Finally, solving equation (E.24) gives

$$\varrho = \varphi, \text{ and} \tag{E.32}$$

$$\mathcal{A} = \pm\mathcal{B}. \tag{E.33}$$

Hence the projective transformation $\mathbf{H}_{\mathrm{SOR}}$ has the form

$$\mathbf{H}_{\mathrm{SOR}} = \begin{bmatrix} \cos\varrho & 0 & \sin\varrho & 0 \\ 0 & h_1 & 0 & h_2 \\ \mp\sin\varrho & 0 & \pm\cos\varrho & 0 \\ 0 & h_3 & 0 & h_4 \end{bmatrix}, \text{ where } \begin{vmatrix} h_1 & h_2 \\ h_3 & h_4 \end{vmatrix} \neq 0. \tag{E.34}$$

Appendix F

Cubic B-splines

A cubic B-spline [43] is specified by N control points $\{\tilde{\mathbf{q}}_i\}_{i=1}^{N}$ and comprises $N-3$ cubic polynomial curve segments $\{\tilde{\mathbf{w}}_i\}_{i=1}^{N-3}$ (see figure F.1). The equation for each curve segment $\tilde{\mathbf{w}}_i$ is given by

$$\tilde{\mathbf{w}}_i(s) = \frac{1}{6}[s^3 \; s^2 \; s \; 1] \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{q}}_i \\ \tilde{\mathbf{q}}_{i+1} \\ \tilde{\mathbf{q}}_{i+2} \\ \tilde{\mathbf{q}}_{i+3} \end{bmatrix}, \tag{F.1}$$

where $0 \leq s \leq 1$ and $1 \leq i \leq N-3$.

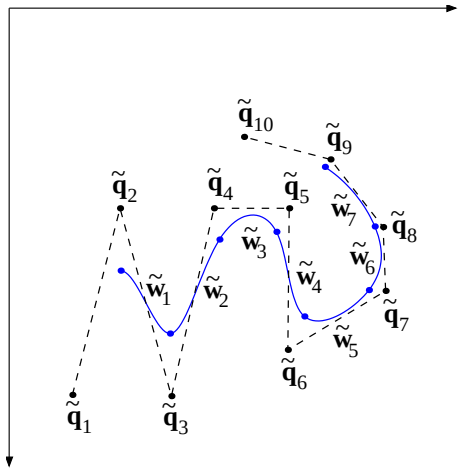


Figure F.1: A cubic B-spline with 10 control points.

B-splines are defined with C^2 continuity at each joining point (knot) between adjacent curve segments, though multiple knots may be used to reduce the continuity at knots. Each additional control point allows one more inflection, and the B-splines may be open or closed as required. For a closed B-spline, the N control points are used in a cyclic manner in equation (F.1) to produce N curve segments. Unlike a single high order polynomial curve, B-splines exhibit local control. This means that modifying the position of one control point causes only a small part of the curve to change, making it particularly suitable for edge fitting (see figure F.2).

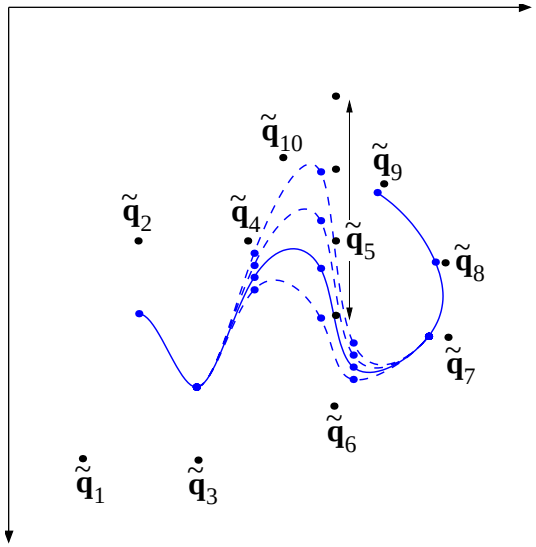


Figure F.2: When the control point $\tilde{\mathbf{q}}_5$ is being moved up and down, only 4 out of the 7 segments of the B-spline change.

Appendix G

Behaviour of the Cost Functions for Motion Estimation

In Chapter 5, algorithms and implementations for motion estimation from silhouettes have been presented. The motion estimation proceeds as an optimization which minimizes the rms reprojection error of the epipolar tangents to the silhouettes. The $N + 2$ motion parameters for a sequence of N images under circular motion are given by

$$\mathbf{m} = [\theta_{\mathrm{s}}\ d_{\mathrm{s}}\ d_{\mathrm{h}}\ \theta_{1,2}\ \theta_{2,3}\ \cdots\ \theta_{N-1,N}], \tag{G.1}$$

where θ_{s} and d_{s} define the image of the rotation axis $\mathbf{l}_{\mathrm{s}}$, and d_{h} defines the intersection $\mathbf{x}_{\mathrm{h}}$ of the horizon $\mathbf{l}_{\mathrm{h}}$ with $\mathbf{l}_{\mathrm{s}}$ (see figure G.1). Given the camera calibration matrix $\mathbf{K}$, the special vanishing point $\mathbf{v}_x = \mathbf{K}\mathbf{K}^{\mathrm{T}}\mathbf{l}_{\mathrm{s}}$ and the horizon $\mathbf{l}_{\mathrm{h}} = \mathbf{v}_x \times \mathbf{x}_{\mathrm{h}}$ can then be determined. The remaining $N-1$ parameters correspond to the $N-1$ angles between the N images, and $\theta_{i,j}$ indicates the rotation angle between image i and image j. Figure G.2 shows 4 different plots of the cost function (5.27) for the Haniwa sequence under circular motion (see figure 5.14), when different pairs of the motion parameters were varied. It can be seen from figure G.2 that though local minima did occur when the 2 consecutive rotation angles were both

close to 90°, the cost function was smooth in most of the search space and had a well-defined global minimum. This explains why the algorithm for circular motion estimation always converges roughly to the same solution even with a poor initialization of $\mathbf{l}_\mathrm{s}$ and $\mathbf{l}_\mathrm{h}$ (see Section 5.6.3).

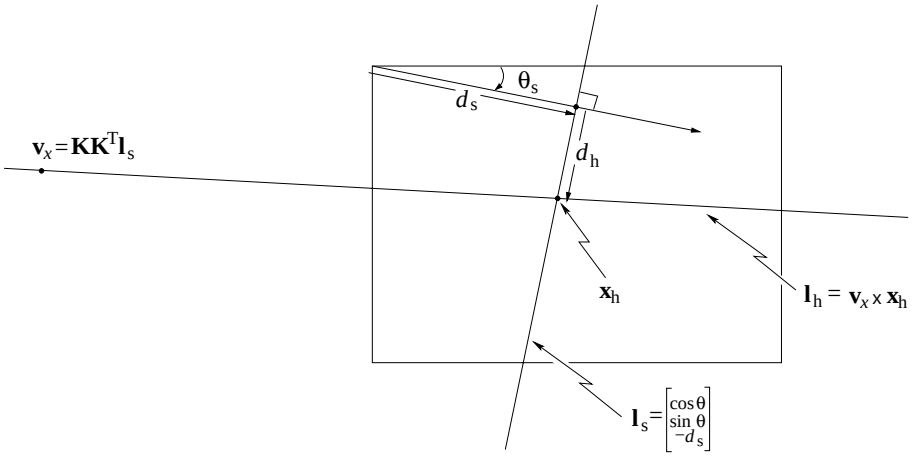


Figure G.1: The 3 parameters defining $\mathbf{l}_\mathrm{s}$, $\mathbf{v}_\mathrm{x}$ and $\mathbf{l}_\mathrm{h}$.

For the registration of a general view with the circular motion, the 6 motion parameters are given by

$$\mathbf{m}' = [\theta_\mathrm{r}\ AE\ EL\ X\ Y\ Z], \tag{G.2}$$

where $\mathbf{T} = [X\ \ Y\ \ Z]^\mathrm{T}$ represents a translation vector, and the angles θ_r, AE and EL define a rotation matrix $\mathbf{R}$ (see figure G.3). Note that $\mathbf{R}$ and $\mathbf{t}$ represent the extrinsic parameters of the projection matrix of the general view. Figure G.4 shows 4 different plots of the cost function (5.28) for registering the 12^{th} view in the Haniwa sequence with the first 11 views (see figure 5.14). It can be seen from figure G.4 that the cost function was not as smooth as that for the circular motion, and that there were lots of local minima around the true solution. As

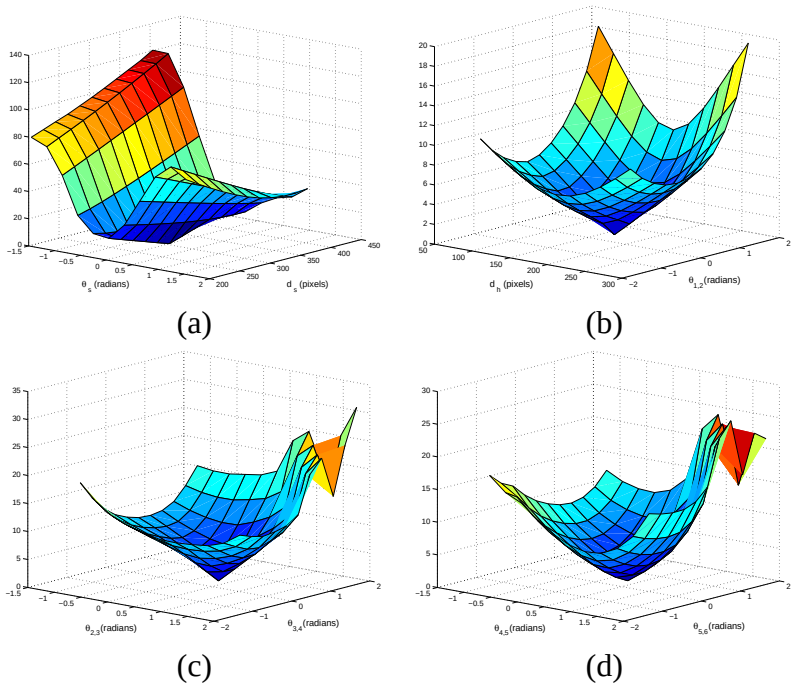


Figure G.2: Plots of the cost function for the Haniwa sequence under circular motion, when different pairs of the motion parameters were varied.

a result, a very good initialization is required for the algorithm to converge to the true solution. This is achieved by rotating and translating the camera, using a user-friendly mouse-controlled interface, until the projection of the initial 3D model built from the circular motion roughly matches the silhouette in the new view (see Section 5.5).

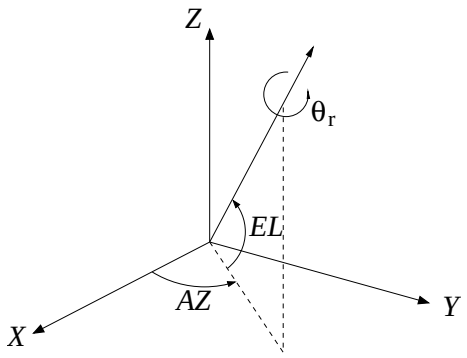


Figure G.3: The 3 parameters of the rotation matrix $\mathbf{R}$ are the rotation angle θ_{r}, the azimuth AE and the elevation EL.

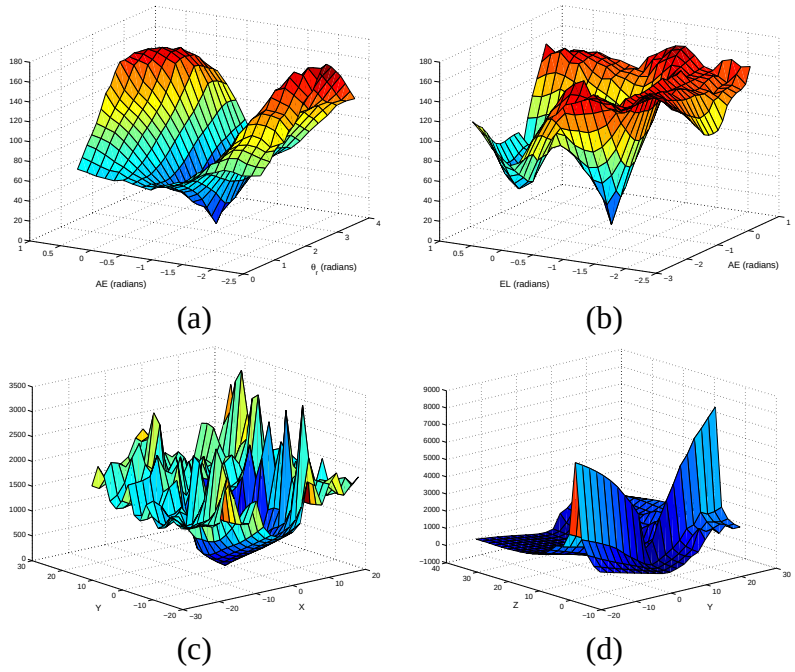


Figure G.4: Plots of the cost function for registering the 12^{th} view of the Haniwa sequence with the first 11 views, when different pairs of the motion parameters were varied. Note that the cost function was not as smooth as that for the circular motion, and that there were lots of local minima around the true solution. As a result, a very good initialization is required for the algorithm to converge to the true solution.

Bibliography

[1] Y. I. Abdel-Aziz and H. M. Karara. Direct linear transformation from comparator coordinates into object space coordinates in close-range photogrammetry. In *Proc. ASP/UI Symp. Close-Range Photogrammetry*, pages 1–18, Urbana, IL, January 1971.

[2] N. Ahuja and J. Veenstra. Generating octrees from object silhouettes in orthographic views. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 11(2):137–149, February 1989.

[3] M. Armstrong, A. Zisserman, and R. Hartley. Self-calibration from image triplets. In B. Buxton and R. Cipolla, editors, *Proc. 4th European Conf. on Computer Vision*, volume 1064 of *Lecture Notes in Computer Science*, pages 3–16, Cambridge, UK, April 1996. Springer–Verlag.

[4] K. Åström, R. Cipolla, and P. Giblin. Generalised epipolar constraints. *Int. Journal of Computer Vision*, 33(1):51–72, September 1999.

[5] H. H. Baker and T. O. Binford. Depth from edge and intensity based stereo. In *Proc. 7th Int. Joint Conf. on Artificial Intelligence*, pages 631–636, Vancouver, BC, Canada, August 1981.

[6] S. T. Barnard and M. A. Fischler. Computational stereo. *ACM Computing Surveys*, 14(4):553–572, December 1982.

[7] H. G. Barrow and J. M. Tenenbaum. Recovering intrinsic scene characteristics from images. In A. R. Hanson and E. M. Riseman, editors, *Computer Vision Systems*, pages 3–26. Academic Press, New York, 1978.

[8] H. G. Barrow and J. M. Tenenbaum. Interpreting line drawings as three-dimensional surfaces. *Artificial Intelligence*, 17:75–116, August 1981.

[9] P. A. Beardsley, A. Zisserman, and D. W. Murray. Sequential updating of projective and affine structure from motion. *Int. Journal of Computer Vision*, 23(3):235–259, June 1997.

[10] T. O. Binford. Visual perception by computer. In *Proc. IEEE Conf. Systems and Control*, Miami, FL, December 1971.

[11] A. Blake and C. Marinos. Shape from texture: Estimation, isotropy and moments. *Artificial Intelligence*, 45(3):323–380, October 1990.

[12] E. Boyer and M. O. Berger. 3d surface reconstruction using occluding contours. *Int. Journal of Computer Vision*, 22(3):219–233, March 1997.

[13] M. Brady, J. Ponce, A. L. Yuille, and H. Asada. Describing surfaces. *Computer Vision, Graphics and Image Processing*, 32(1):1–28, October 1985.

[14] A. Broadhurst, T. W. Drummond, and R. Cipolla. A probabilistic framework for space carving. In *Proc. 8th Int. Conf. on Computer Vision*, volume I, pages 388–393, Vancouver, BC, Canada, July 2001.

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

[16] J. Canny. A computational approach to edge detection. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 8(6):679–698, November 1986.

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

[18] I. Carlbom, I. Chakravarty, and D. Vanderschel. A hierarchical data structure for representing the spatial decomposition of 3-d objects. *IEEE Computer Graphics and Applications*, 5(4):24–31, 1985.

[19] T. J. Cham and R. Cipolla. Geometric saliency of curve correspondences and grouping of symmetric contours. In B. Buxton and R. Cipolla, editors, *Proc. 4th European Conf. on Computer Vision*, volume 1064 of *Lecture Notes in Computer Science*, pages 385–398, Cambridge, UK, April 1996. Springer–Verlag.

[20] H. H. Chen and T. S. Huang. A survey of construction and manipulation of octrees. *Computer Vision, Graphics and Image Processing*, 43(3):409–431, September 1988.

[21] C. H. Chien and J. K. Aggarwal. Volume/surface octrees for the representation of three-dimensional objects. *Computer Vision, Graphics and Image Processing*, 36(1):100–113, October 1986.

[22] R. Cipolla, K. E. Åström, and P. J. Giblin. Motion from the frontier of curved surfaces. In *Proc. 5th Int. Conf. on Computer Vision*, pages 269–275, Cambridge, MA, USA, June 1995.

[23] R. Cipolla and A. Blake. The dynamic analysis of apparent contours. In *Proc. 3rd Int. Conf. on Computer Vision*, pages 616–623, Osaka, Japan, December 1990.

[24] R. Cipolla and A. Blake. Surface shape from the deformation of apparent contours. *Int. Journal of Computer Vision*, 9(2):83–112, November 1992.

[25] R. Cipolla, T. W. Drummond, and D. Robertson. Camera calibration from vanishing points in images of architectural scenes. In T. Pridmore and D. Elliman, editors, *Proc. British Machine Vision Conference*, volume 2, pages 382–391, Nottingham, UK, September 1999.

[26] R. Cipolla, G. Fletcher, and P. J. Giblin. Following cusps. *Int. Journal of Computer Vision*, 23(2):115–129, June 1997.

[27] R. Cipolla and P. J. Giblin. *Visual Motion of Curves and Surfaces*. Cambridge University Press, Cambridge, UK, 1999.

[28] M. B. Clowes. On seeing things. *Artificial Intelligence*, 2(1):79–116, 1971.

[29] H. S. M. Coxeter. *Introduction to Geometry*. Wiley and Sons, New York, 2^{nd} edition, 1989.

[30] G. Cross, A. W. Fitzgibbon, and A. Zisserman. Parallax geometry of smooth surfaces in multiple views. In *Proc. 7th Int. Conf. on Computer Vision*, pages 323–329, Corfu, Greece, September 1999.

[31] G. Cross and A. Zisserman. Quadric reconstruction from dual-space geometry. In *Proc. 6th Int. Conf. on Computer Vision*, pages 25–31, Bombay, India, January 1998.

[32] G. Cross and A. Zisserman. Surface reconstruction from multiple views using apparent contours and surface texture. In A. Leonardis, F. Solina, and R. Bajcsy, editors, *NATO Advanced Research Workshop on Confluence of Computer Vision and Computer Graphics*, pages 25–47, Ljubljana, Slovenia, 2000.

[33] R. W. Curwen, C. V. Stewart, and J. L. Mundy. Recognition of plane projective symmetry. In *Proc. 6th Int. Conf. on Computer Vision*, pages 1115–1122, Bombay, India, January 1998.

[34] L. S. Davis, L. Janos, and S. M. Dunn. Efficient recovery of shape from texture. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 5(5):485–492, September 1983.

[35] L. de Agapito, R. I. Hartley, and E. Hayman. Linear calibration of a rotating and zooming camera. In *Proc. Conf. Computer Vision and Pattern Recognition*, volume I, pages 15–21, Fort Collins, CO, June 1999.

[36] A. R. Dick, P. H. S. Torr, S. J. Ruffle, and R. Cipolla. Combining single view recognition and multiple view stereo for architectural scenes. In *Proc. 8th Int. Conf. on Computer Vision*, volume I, pages 268–274, Vancouver, BC, Canada, July 2001.

[37] M. J. Dürst. Letters: Additional reference to “marching cubes”. *ACM Computer Graphics*, 22(2):72–73, April 1988.

[38] W. Faig. Calibration of close-range photogrammetry system: Mathematical formulation. *Photogrammetric Engineering and Remote Sensing*, 41(12):1479–1486, December 1975.

[39] O. D. Faugeras. What can be seen in three dimensions with an uncalibrated stereo rig. In G. Sandini, editor, *Proc. 2nd European Conf. on Computer Vision*, volume 588 of *Lecture Notes in Computer Science*, pages 563–578, Santa Margherita Ligure, Italy, May 1992. Springer–Verlag.

[40] O. D. Faugeras. *Three-Dimensional Computer Vision: a Geometric Viewpoint*. MIT Press, Cambridge, MA, 1993.

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

[42] O. D. Faugeras and G. Toscani. The calibration problem for stereo. In *Proc. Conf. Computer Vision and Pattern Recognition*, pages 15–20, Miami, FL, June 1986.

[43] I. D. Faux and M. J. Pratt. *Computational Geometry for Design and Manufacture*. Ellis Horwood, New York, 1979.

[44] A. W. Fitzgibbon, G. Cross, and A. Zisserman. Automatic 3D model construction for turn-table sequences. In R. Koch and L. Van Gool, editors, *3D Structure from Multiple Images of Large-Scale Environments, European Workshop SMILE'98*, volume 1506 of *Lecture Notes in Computer Science*, pages 155–170, Freiburg, Germany, June 1998. Springer–Verlag.

[45] P. J. Giblin, F. E. Pollick, and J. E. Rycroft. Recovery of an unknown axis of rotation from the profiles of a rotating surface. *Journal of Optical Soc. of America A*, 11(7):1976–1984, July 1994.

[46] P. J. Giblin and R. S. Weiss. Reconstructions of surfaces from profiles. In *Proc. 1st Int. Conf. on Computer Vision*, pages 136–144, London, UK, June 1987.

[47] P. J. Giblin and R. S. Weiss. Epipolar curves on surfaces. *Image and Vision Computing*, 13(1):33–44, February 1995.

[48] J. J. Gibson. *The Ecological Approach to Visual Perception*. Houghton Mifflin, Boston, MA, 1979.

[49] A. D. Gross and T. E. Boult. Recovery of SHGCs from a single intensity view. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 18(2):161–180, February 1996.

[50] A. Guzman-Arenas. Computer recognition of three-dimensional objects in a visual scene. Technical Report MAC-TR-59, MIT, December 1968.

[51] C. Harris and M. J. Stephens. A combined corner and edge detector. In *4th Alvey Conference*, pages 147–152, Manchester, UK, August 1988.

[52] R. Hartley and R. Gupta. Computing matched-epipolar projections. In *Proc. Conf. Computer Vision and Pattern Recognition*, pages 549–555, New York, NY, June 1993.

[53] R. I. Hartley. Estimation of relative camera positions for uncalibrated cameras. In G. Sandini, editor, *Proc. 2nd European Conf. on Computer Vision*,

volume 588 of *Lecture Notes in Computer Science*, pages 579–587, Santa Margherita Ligure, Italy, May 1992. Springer–Verlag.

[54] R. I. Hartley. Projective reconstruction and invariants from multiple images. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 16(10):1036–1041, October 1994.

[55] R. I. Hartley and P. Sturm. Triangulation. *Computer Vision and Image Understanding*, 68(2):146–157, November 1997.

[56] T. H. Hong and M. O. Shneier. Describing a robot's workspace using a sequence of views from a moving camera. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 7(6):721–726, November 1985.

[57] R. Horaud and M. Brady. On the geometric interpretation of image contours. *Artificial Intelligence*, 37:333–353, December 1988.

[58] B. K. P. Horn. Understanding image intensities. *Artificial Intelligence*, 8(2):201–231, 1977.

[59] B. K. P. Horn. *Robot Vision*. MIT Press, Cambridge, MA, 1986.

[60] D. A. Huffman. Impossible objects as non-sense sentences. *Machine Intelligence*, 6:295–323, 1971.

[61] K. Ikeuchi and B. K. P. Horn. Numerical shape from shading and occluding boundaries. *Artificial Intelligence*, 17:141–184, 1981.

[62] Y. A. Ivanov, A. F. Bobick, and J. Liu. Fast lighting independent background subtraction. *Int. Journal of Computer Vision*, 37(2):199–207, June 2000.

[63] C. L. Jackins and S. L. Tanimoto. Oct-trees and their use in representing three-dimensional objects. *Computer Graphics Image Processing*, 14(3):249–270, November 1980.

[64] T. Joshi, N. Ahuja, and J. Ponce. Structure and motion estimation from dynamic silhouettes under perspective projection. *Int. Journal of Computer Vision*, 31(1):31–50, February 1999.

[65] K. I. Kanatani and T. C. Chou. Shape from texture: General principle. *Artificial Intelligence*, 38(1):1–48, 1989.

[66] J. R. Kender and Kanade. Mapping image properties into shape constraints: Skewed symmetry, affine-transformable patterns, and the shape-from-texture paradigm. In J. Beck, B. Hope, and A. Rosenfeld, editors, *Human and Machine Vision*, pages 237–257. Academic Press, New York, 1983.

[67] R. Koch, M. Pollefeys, and L. van Gool. Multi viewpoint stereo from uncalibrated video sequences. In H. Burkhardt and B. Neumann, editors, *Proc. 5th European Conf. on Computer Vision*, volume 1406 of *Lecture Notes in Computer Science*, pages 55–71, Freiburg, Germany, June 1998. Springer–Verlag.

[68] J. J. Koenderink. What does the occluding contour tell us about solid shape? *Perception*, 13:321–330, 1984.

[69] J. J. Koenderink. *Solid Shape*. MIT Press, Cambridge, MA, 1990.

[70] J. J. Koenderink and A. J. van Doorn. Geometry of binocular vision and a model for stereopsis. *Biological Cybernetics*, 21:29–35, 1976.

[71] J. J. Koenderink and A. J. van Doorn. The singularities of the visual mapping. *Biological Cybernetics*, 24(1):51–59, 1976.

[72] K. N. Kutulakos and S. M. Seitz. A theory of shape by space carving. *Int. Journal of Computer Vision*, 38(3):197–216, July 2000.

[73] A. Laurentini. The visual hull concept for silhouette-based image understanding. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 16(2):150–162, February 1994.

[74] A. Laurentini. How far 3D shapes can be understood from 2D silhouettes. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 17(2):188–195, February 1995.

[75] R. K. Lenz and R. Y. Tsai. Techniques for calibration of the scale factor and image center for high accuracy 3D machine vision metrology. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 10(5):713–720, September 1988.

[76] D. Liebowitz and A. Zisserman. Combining scene and auto-calibration constraints. In *Proc. 7th Int. Conf. on Computer Vision*, pages 293–300, Corfu, Greece, September 1999.

[77] J. Liu, J. L. Mundy, D. A. Forsyth, A. Zisserman, and C. A. Rothwell. Efficient recognition of rotationally symmetric surface and straight homo-

geneous generalized cylinders. In *Proc. Conf. Computer Vision and Pattern Recognition*, pages 123–129, New York, NY, June 1993.

[78] H. C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135, September 1981.

[79] H. C. Longuet-Higgins and K. Prazdny. The interpretation of a moving retinal image. *Proc. Royal Soc. London B*, 208:385–397, 1980.

[80] W. E. Lorensen and H. E. Cline. Marching cubes: a high resolution 3D surface construction algorithm. *ACM Computer Graphics*, 21(4):163–169, July 1987.

[81] Q. T. Luong and O. D. Faugeras. The fundamental matrix: Theory, algorithm, and stability analysis. *Int. Journal of Computer Vision*, 17(1):43–75, January 1996.

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

[83] A. K. Mackworth. Interpreting pictures of polyhedral scenes. *Artificial Intelligence*, 4(2):121–139, 1973.

[84] J. Malik. Interpreting line drawings of curved objects. *Int. Journal of Computer Vision*, 1(1):73–103, 1987.

[85] J. Malik and R. Rosenholtz. Computing local surface orientation and shape from texture for curved surfaces. *Int. Journal of Computer Vision*, 23(2):149–168, June 1997.

[86] C. Marinos and A. Blake. Shape from texture: The homogeneity hypothesis. In *Proc. 3rd Int. Conf. on Computer Vision*, pages 350–353, Osaka, Japan, December 1990.

[87] D. Marr. Analysis of occluding contour. *Proc. Royal Soc. London B*, 197:441–475, 1977.

[88] D. Marr. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. Freeman, San Francisco, 1982.

[89] D. Marr and T. A. Poggio. Cooperative computation of stereo disparity. *Science*, 194(4262):283–287, October 1976.

[90] W. N. Martin and J. K. Aggarwal. Volumetric descriptions of objects from multiple views. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 5(2):150–158, March 1983.

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

[92] D. J. Meagher. Geometric modeling using octree encoding. *Computer Graphics Image Processing*, 19(2):129–147, June 1982.

[93] P. R. S. Mendonça and R. Cipolla. Estimation of epipolar geometry from apparent contours: Affine and circular motion cases. In *Proc. Conf. Computer Vision and Pattern Recognition*, volume I, pages 9–14, Fort Collins, CO, 1999.

[94] P. R. S. Mendonça, K.-Y. K. Wong, and R. Cipolla. Recovery of circular motion from profiles of surfaces. In B. Triggs, A. Zisserman, and R. Szeliski, editors, *Vision Algorithms: Theory and Practice*, volume 1883 of *Lecture Notes in Computer Science*, pages 151–167, Corfu, Greece, September 1999. Springer–Verlag.

[95] P. R. S. Mendonça, K.-Y. K. Wong, and R. Cipolla. Camera pose estimation and reconstruction from image profiles under circular motion. In D. Vernon, editor, *Proc. 6th European Conf. on Computer Vision*, volume 1843 of *Lecture Notes in Computer Science*, pages 864–877, Dublin, Ireland, June 2000. Springer–Verlag.

[96] P. R. S. Mendonça, K.-Y. K. Wong, and R. Cipolla. Epipolar geometry from profiles under circular motion. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 23(6):604–616, June 2001.

[97] C. Montani, R. Scateni, and R. Scopigno. Discretized marching cubes. In R. D. Bergeron and A. E. Kaufman, editors, *Proc. Visualization '94*, pages 281–287, Washington, DC, October 1994.

[98] C. Montani, R. Scateni, and R. Scopigno. A modified look-up table for implicit disambiguation of marching cubes. *The Visual Computer*, 10(6):353–355, 1994.

[99] D. P. Mukherjee, A. Zisserman, and J. M. Brady. Shape from symmetry–detecting and exploiting symmetry in affine images. *Phil. Trans. Royal Soc. London A*, 351:77–106, 1995.

[100] J. L. Mundy and A. Zisserman. *Geometric Invariance in Computer Vision*. MIT Press, Cambridge, MA, 1992.

[101] J. L. Mundy and A. Zisserman. Repeated structures: Image correspondence constraints and 3D structure recovery. In J. L. Mundy, A. Zisserman, and D. Forsyth, editors, *Applications of Invariance in Computer Vision, Second Joint European - US Workshop*, volume 825 of *Lecture Notes in Computer Science*, pages 89–106, Ponta Delgada, Azores, Portugal, October 1993. Springer–Verlag.

[102] V. S. Nalwa. Line-drawing interpretation: Bilateral symmetry. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 11(10):1117–1120, October 1989.

[103] H. Noborio, S. Fukuda, and S. Arimoto. Construction of the octree approximating three-dimensional objects by using multiple views. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 10(6):769–782, November 1988.

[104] M. Pollefeys, R. Koch, and L. J. van Gool. Self-calibration and metric reconstruction inspite of varying and unknown intrinsic camera parameters. *Int. Journal of Computer Vision*, 32(1):7–25, August 1999.

[105] F. E. Pollick. Perceiving shape from profiles. *Perception and Psychophysics*, 55(2):152–161, 1994.

[106] J. Ponce, D. M. Chelberg, and W. B. Mann. Invariant properties of straight homogeneous generalized cylinders and their contours. *IEEE Trans. on*

Pattern Analysis and Machine Intelligence, 11(9):951–966, September 1989.

[107] J. Porrill and S. B. Pollard. Curve matching and stereo calibration. *Image and Vision Computing*, 9(1):45–50, February 1991.

[108] M. Potmesil. Generating octree models of 3D objects from their silhouettes in a sequence of images. *Computer Vision, Graphics and Image Processing*, 40(1):1–29, October 1987.

[109] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C : The Art of Scientific Computing*. Cambridge University Press, Cambridge, UK, 2 edition, January 1993.

[110] L. Quan. Self-calibration of an affine camera from multiple views. *Int. Journal of Computer Vision*, 19(1):93–105, July 1996.

[111] J. H. Rieger. Three dimensional motion from fixed points of a deforming profile curve. *Optics Letters*, 11(3):123–125, March 1986.

[112] L. G. Roberts. Machine perception of three-dimensional solids. In J. T. Tippett, D. A. Berkowitz, L. C. Clapp, C. J. Koester, and A. Vanderburgh, Jr., editors, *Optical and Electro-Optical Information Processing*, pages 159–197. MIT Press, 1965.

[113] H. Samet. *Design and Analysis of Spatial Data Structures*. Addison-Wesley, Reading, MA, 1990.

[114] H. Sato and T. O. Binford. Finding and recovering SHGC objects in an edge image. *Computer Vision, Graphics and Image Processing*, 57(3):346–358, May 1993.

[115] J. Sato and R. Cipolla. Affine integral invariants for extracting symmetry axes. *Image and Vision Computing*, 15(8):627–635, August 1997.

[116] J. Sato and R. Cipolla. Affine reconstruction of curved surfaces from uncalibrated views of apparent contours. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 21(11):1188–1198, November 1999.

[117] J. G. Semple and G. T. Kneebone. *Algebraic Projective Geometry*. Oxford Classic Texts in the Physical Sciences. Clarendon Press, Oxford, UK, 1998. Originally published in 1952.

[118] L. S. Shapiro, A. Zisserman, and M. Brady. 3D motion recovery via affine epipolar geometry. *Int. Journal of Computer Vision*, 16(2):147–182, October 1995.

[119] I. Sobel. On calibrating computer controlled cameras for perceiving 3-D scenes. *Artificial Intelligence*, 5(2):185–198, June 1974.

[120] S. K. Srivastava and N. Ahuja. Octree generation from object silhouettes in perspective views. *Computer Vision, Graphics and Image Processing*, 49(1):68–84, January 1990.

[121] C. Stauffer and W. E. L. Grimson. Adaptive background mixture models for real-time tracking. In *Proc. Conf. Computer Vision and Pattern Recognition*, volume II, pages 246–252, Fort Collins, CO, June 1999.

[122] K. Sugihara. An algebraic approach to shape-from-image problems. *Artificial Intelligence*, 23(1):59–95, 1984.

[123] S. Sullivan and J. Ponce. Automatic model construction and pose estimation from photographs using triangular splines. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 20(10):1091–1096, October 1998.

[124] R. Szeliski. Rapid octree construction from image sequences. *Computer Vision, Graphics and Image Processing*, 58(1):23–32, July 1993.

[125] R. Szeliski and R. Weiss. Robust shape recovery from occluding contours using a linear smoother. *Int. Journal of Computer Vision*, 28(1):27–44, June 1998.

[126] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: A factorization method. *Int. Journal of Computer Vision*, 9(2):137–154, November 1992.

[127] K. Toyama, J. Krumm, B. Brumitt, and B. Meyers. Wallflower: Principles and practice of background maintenance. In *Proc. 7th Int. Conf. on Computer Vision*, pages 255–261, Corfu, Greece, September 1999.

[128] B. Triggs. Autocalibration and the absolute quadric. In *Proc. Conf. Computer Vision and Pattern Recognition*, pages 609–614, San Jaun, PR, June 1997.

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

[130] R. Y. Tsai and T. S. Huang. Uniqueness and estimation of three-dimensional motion parameters of rigid objects with curved surfaces. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 6(1):13–27, January 1984.

[131] K. J. Turner. *Computer Perception of Curved Objects Using a Television Camera*. PhD thesis, University of Edinburgh, 1974.

[132] S. Ullman. *The Interpretation of Visual Motion*. MIT Press, Cambridge, MA, 1979.

[133] F. Ulupinar and R. Nevatia. Shape from contour: Straight homogeneous generalized cylinders and constant cross-section generalized cylinders. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 17(2):120–135, February 1995.

[134] R. Vaillant and O. D. Faugeras. Using extremal boundaries for 3D object modeling. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 14(2):157–173, February 1992.

[135] T. Vieville and D. Lingrand. Using singular displacements for uncalibrated monocular visual systems. In B. Buxton and R. Cipolla, editors, *Proc. 4th European Conf. on Computer Vision*, volume 1065 of *Lecture Notes in Computer Science*, pages 207–216, Cambridge, UK, April 1996. Springer–Verlag.

[136] D. Waltz. Understanding line drawings of scenes with shadows. *Artificial Intelligence*, 2:79–116, 1971.

[137] A. P. Witkin. Recovering surface shape and orientation from texture. *Artificial Intelligence*, 17:17–45, 1981.

[138] K.-Y. K. Wong and R. Cipolla. Structure and motion from silhouettes. In *Proc. 8th Int. Conf. on Computer Vision*, volume II, pages 217–222, Vancouver, BC, Canada, July 2001.

[139] K.-Y. K. Wong, P. R. S. Mendonça, , and R. Cipolla. Head model acquisition from silhouettes. In C. Arcelli, L. P. Cordella, and G. Sanniti di Baja, editors, *4th International Workshop on Visual Form*, volume 2059 of *Lecture Notes in Computer Science*, pages 797–796, Capri, Italy, May 2001. Springer–Verlag.

[140] K.-Y. K. Wong, P. R. S. Mendonça, and R. Cipolla. Reconstruction and motion estimation from apparent contours under circular motion. In T. Pridmore and D. Elliman, editors, *Proc. British Machine Vision Conference*, volume 1, pages 83–92, Nottingham, UK, September 1999.

[141] K.-Y. K. Wong, P. R. S. Mendonça, and R. Cipolla. Camera calibration from symmetry. In R. Cipolla and R. Martin, editors, *The Mathematics of Surfaces IX*, pages 214–226, Cambridge, UK, September 2000. Springer–Verlag.

[142] R. J. Woodham. Analysing images of curved surfaces. *Artificial Intelligence*, 17:117–140, 1981.

[143] M. Zerroug and R. Nevatia. Three-dimensional descriptions based on the analysis of the invariant and quasi-invariant properties of some curved-axis

generalized cylinders. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 18(3):237–253, March 1996.

[144] M. Zerroug and R. Nevatia. Volumetric descriptions from a single intensity image. *Int. Journal of Computer Vision*, 20(1/2):11–42, 1996.

[145] R. Zhang, P. S. Tsai, J. E. Cryer, and M. Shah. Shape from shading: A survey. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 21(8):690–706, August 1999.

[146] Z. Zhang. Determining the epipolar geometry and its uncertainty: A review. *Int. Journal of Computer Vision*, 27(2):161–195, March 1998.

[147] A. Zisserman, D. Forsyth, J. L. Mundy, and C. A. Rothwell. Recognizing general curved objects efficiently. In J. L. Mundy and A. Zisserman, editors, *Geometric Invariance in Computer Vision*, Artificial Intelligence Series, chapter 11, pages 228–251. MIT Press, Cambridge, MA, 1992.

[148] A. Zisserman, D. Liebowitz, and M. Armstrong. Resolving ambiguities in auto-calibration. *Phil. Trans. Royal Soc. London A*, 356(1740):1193–1211, 1998.

[149] A. Zisserman, J. L. Mundy, D. A. Forsyth, J. Liu, N. Pillow, C. Rothwell, and S. Utcke. Class-based grouping in perspective images. In *Proc. 5th Int. Conf. on Computer Vision*, pages 183–188, Cambridge, MA, USA, June 1995.