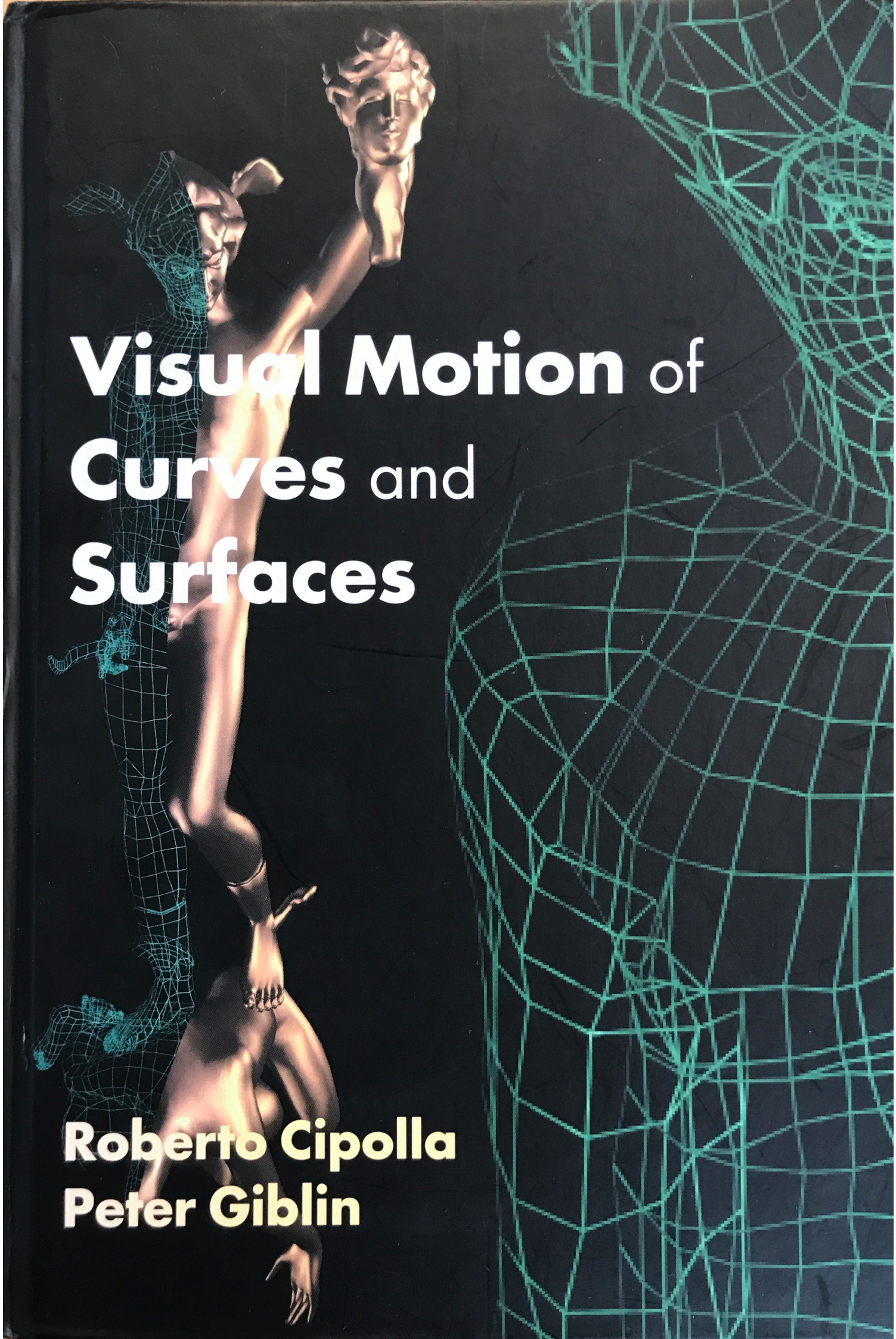


Visual Motion of Curves and Surfaces

Roberto Cipolla
Peter Giblin

Visual Motion of Curves and Surfaces

Roberto Cipolla

Peter Giblin

Contents

Preface		*page* vi
1	Introduction	1
2	Differential geometry of curves and surfaces	5
2.1	Curves and their tangents	5
2.2	Surfaces: the parametric form	7
2.3	Monge form	13
2.4	Implicit form	15
2.5	First fundamental form for surfaces	18
2.6	Curvature of curves	21
2.7	Three surface types	26
2.8	Second fundamental form and curvatures: parametrized surfaces	28
2.9	Second fundamental form and curvatures: Monge form of surface	41
2.10	Special Monge form	43
2.11	Second fundamental form: implicit form of surface	47
2.12	Special curves on a surface	47
2.13	Contact	50
3	Views of curves and surfaces	57
3.1	Camera models: parallel (orthographic) projection	57
3.2	Perspective projection	58
3.3	Opaque vs. semi-transparent surfaces	62
3.4	Static properties of contour generators and apparent contours	63
3.5	Properties: orthogonal projection	64
3.6	Properties: perspective projection	69
3.7	Methods of proof: Monge–Taylor proofs	76
3.8	Monge–Taylor proofs: orthogonal projection	76
3.9	Monge–Taylor proofs: perspective projection	78
3.10	Vector proofs: orthogonal projection	79
3.11	Vector proofs: perspective projection	80

3.12 Methods of proof: pure geometric proofs 81
4 Dynamic analysis of apparent contours 83
4.1 Orthogonal projection 84
4.2 Epipolar parametrization: orthogonal case 88
4.3 Perspective projection 90
4.4 Epipolar parametrization: perspective case 94
4.5 Surface curvatures using the epipolar parametrization 99
4.6 Degeneracies of the epipolar parametrization 100
4.7 Visual events: swallowtail, lips and beaks 101
4.8 Frontiers (epipolar tangencies) 101
4.9 Following cusps 110
4.10 Formulae for K and H by following cusps 110
4.11 Image velocity of a cusp point 113
4.12 Envelopes of surfaces and apparent contours 114
5 Reconstruction of surfaces from profiles 119
5.1 Localization and tracking of apparent contours 119
5.2 Camera model for perspective projection onto image plane 124
5.3 Camera model for weak perspective and orthographic projection 128
5.4 Camera calibration 129
5.5 Epipolar geometry 131
5.6 Epipolar geometry from projection matrices 134
5.7 Reconstruction of surfaces 136
6 Recovery of viewer motion from profiles 145
6.1 The fundamental matrix from point correspondences 145
6.2 Recovery of the projection matrices and viewer motion 148
6.3 Recovery of the projection matrices for uncalibrated cameras 150
6.4 Frontier points and epipolar tangencies 153
6.5 Recovery of motion under pure translation 155
6.6 General motion 157
6.7 Weak perspective 161
6.8 Circular motion 164
6.9 Envelope of apparent contours under circular motion 171
Afterword 179
Bibliography 181

Preface

Computer Vision is the automatic analysis of sequences of images for the purpose of recovering three-dimensional surface shape. In recent years, several branches of mathematics, both ancient and modern, have been applied to computer vision. Projective geometry, which in its mathematical form dates back at least two centuries, is used to describe the relationship between points and lines in different images of the same object. Differential geometry, which is even older, though it received its definitive modern look in the first half of the nineteenth century, is used to describe the shape of curves and surfaces. More recently developments in singularity theory have enriched the field of geometry by making possible a wealth of detail only dreamed of fifty years ago. Likewise, developments in the speed and power of computers over the last decade have turned other dreams into reality, and made possible real-world applications of mathematical theory.

The goal of this book is to reconstruct surfaces from their 'apparent contours', that is the outlines which they present to us when we view them from a distance. It is not obvious that these apparent contours contain enough information to reconstruct an unmarked smooth surface at all. It is even less obvious that without accurate knowledge of the observer's motion they contain this information; in fact, at the time of writing, we do not know in generality whether this is true. We have, however, successfully implemented the reconstruction when the observer's motion is only partly known – that is, when it is constrained to be of a special kind called circular motion. Other work on more general motion is in progress.

Chapter 1 is introductory, and Chapters 2 to 4 introduce the mathematical ideas and techniques necessary to the study of surfaces and their apparent contours under viewer motion. In Chapters 5 and 6 we bring the mathematics to life with the latest techniques in photogrammetry and computer vision. We describe the real-time implementation of the theory with real

image sequences. We show that in practice apparent contours can be used effectively to reconstruct both motion and surface shape.

Acknowledgements

Chapters 2 and 3 owe a great deal to Andrew Zisserman, who collaborated on an earlier version. We are particularly grateful to him for help in proof-reading and many valuable comments.

Several students have helped in producing figures. Gordon Fletcher was instrumental in creating many of the figures from Chapters 2, 3 and 4, using the 'Liverpool Surfaces Modelling Package' written by Richard Morris. Paulo Mendonca and Kenneth Wong have helped in the real-time implementation of the theory and in obtaining the results described in Chapter 5 and 6.

The first author acknowledges the support of the Engineering and Physical Science Research Council. The second author acknowledges the hospitality of the Mathematics Department of Brown University, where some of this book was written, and also the Fulbright Commission for a Senior Fulbright Scholarship 1997/8. He also acknowledges the European grants VIVA and VANGUARD.

The front cover is based on a reconstruction of Benvenuto Cellini's bronze sculpture, Perseus, which is undergoing restoration at the Uffizi Gallery in Florence. We are very grateful to Dr Francesca d'Uva and Dr Tina Guiducci of Burson Marsteller and Cassa di Risparmio di Firenze and to Francesco Porta of Datanord Multimedia for supplying the images.

1

Introduction

We have tried to make our book self-contained. The underlying differential geometry and singularity theory is explained with minimal prerequisites in Chapter 2. We hope that this chapter will prove of value to anyone who wishes to apply differential geometry to vision problems. To follow the main thrust of this chapter the reader only needs a working knowledge of calculus and linear algebra. Most of the material is quite well known, but the slant given here is towards applications, and we have tried to illustrate the material with many examples and figures. In particular, we study curvature of surfaces and special curves on surfaces such as parabolic and flecnodal curves. We make much use of the idea of contact, between surfaces and lines or planes. It is this idea which links classical differential geometry with modern singularity theory, where geometrical properties are studied by means of functions or mappings which in turn measure contact.

In Chapter 3 we introduce the main character in our story, the apparent contour. Apparent contours are the outlines or profiles of curved surfaces. An example is shown in Figure 1.1. We describe apparent contours under orthographic projection and for perspective projection, and we describe in detail the singularities which a single apparent contour can be expected to have. We also obtain geometrical information about surfaces from a single apparent contour, though this is necessarily limited. Initially results are stated, but we give three different approaches to proofs, 'Monge-Taylor proofs', which rely on special coordinate systems which are very powerful for proving results about surfaces; 'vector proofs', which are coordinate-free but require more experience to use effectively; and 'pure geometric proofs', which are more like thought experiments but can sometimes yield the greatest intuition.

An excellent modern reference for applications of projective geometry to computer vision is O.D. Faugeras' *Three-Dimensional Computer Vision*. On

the differential geometry side, another book from which we, and others, have drawn inspiration is J.J. Koenderink's now classic *Solid Shape*, published in 1990. Koenderink's book is replete with geometric proofs and statements, but sometimes lacks mathematical detail. We have tried to supply some of this detail in Chapters 2 and 3 of our book.

In Chapter 4 we introduce dynamic contours. That is, we progress from a single view of a surface to multiple views, from which we can expect to derive much more information. In fact, in principle, a complete reconstruction of a surface is possible from a family of apparent contours obtained by circumnavigating a surface. (Unfortunately in practice some parts of a surface may be occluded by other parts, and in addition apparent contours may be hard to track.) The 'in principle' reconstruction was first established for orthographic projection in Giblin and Weiss (1987), and this was generalized and placed in a better mathematical framework by Cipolla and Blake (1990 and 1992).

We describe the dynamic analysis for orthographic and perspective projection, and introduce the important idea of an epipolar parametrization. We also give a brief introduction to circular motion, which will play a major role in the last chapters of the book. The epipolar parametrization breaks down in certain circumstances, one of which is the 'epipolar tangency' situation. This is bad news for reconstruction but, surprisingly, very good news for determining motion. In fact the so-called frontier points which arise from epipolar tangency are instrumental in giving us information about the motion of the observer, something we exploit in Chapter 6. Other breakdowns of the epipolar parametrization are caused by degeneracies of the apparent contour – the 'visual events' which we observe when moving our viewpoint – and we list the possible cases and explain their geometrical significance.

In Chapter 5 we bring the mathematical techniques to life and describe the implementation of algorithms to reconstruct a surface from the image sequence of outlines. Details of every stage in the reconstruction, from raw pixel intensities to a stable description of the three-dimensional surface, are given. These include the calibration of cameras, localization and tracking of outlines, epipolar geometry and stereo reconstruction.

In Chapter 6 we address the more difficult problem of recovering the observer's motion from the apparent contours in different views. The recovery of the three-dimensional configuration of points and the motion compatible with their views (known as structure from motion) has been an active area of research in computer vision over the last two decades and a large number of algorithms and working systems already exist. We review the key results in the literature, in many cases providing simple geometric and

algebraic proofs. Finally we show how the motion of the viewer can be computed from apparent contours instead of points, by using properties of the frontier. The significance of frontier points seems to have been noticed first by J. Rieger (1986), and they were then applied to circular motion and orthographic projection in (Giblin et al. 1994), where it is proved that recovery of motion is essentially unique in this simple case. The extension to general motion and perspective projection was presented by (Cipolla et al. 1995) and (Aström et al. 1996 and 1999), where an iterative algorithm gives good results in many cases. We present the latest techniques for estimating the camera motion. A particularly simple and reliable method is presented for recovering the motion of objects on turntables, known as circular motion. This exploits symmetry of the envelope of apparent contours. This has been used to acquire three-dimensional models of arbitrary objects from an uncalibrated camera.

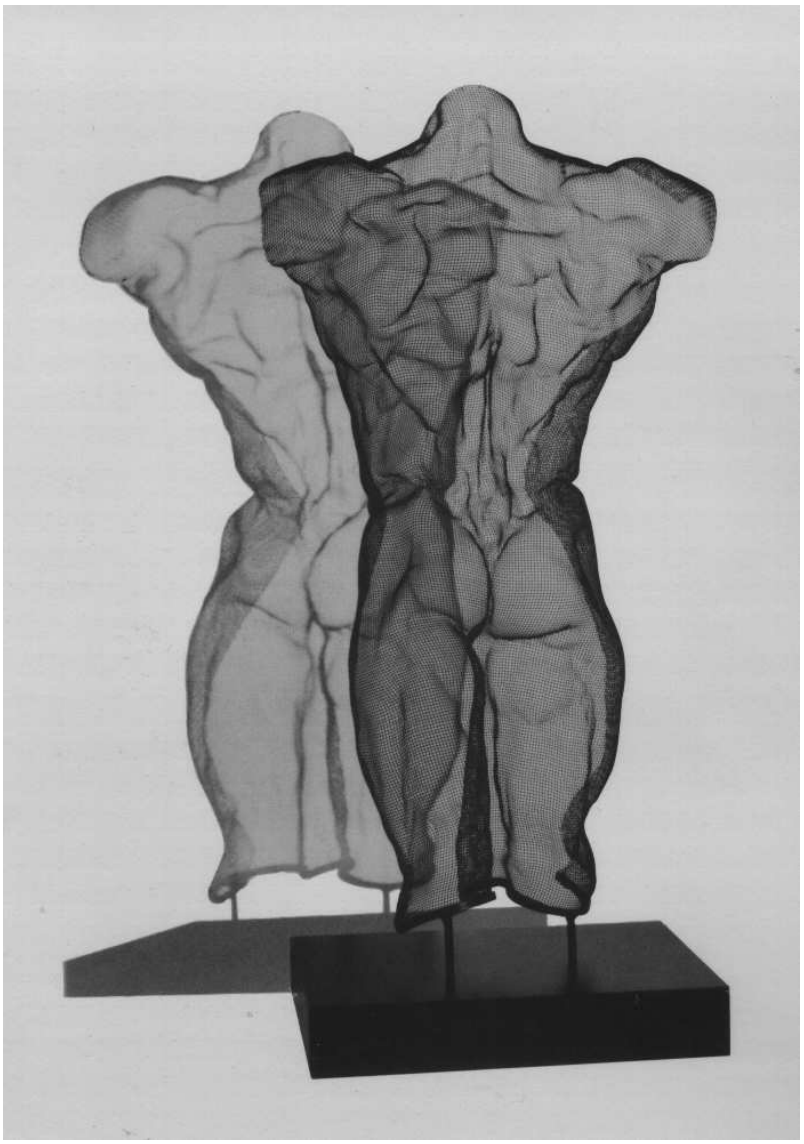


Fig. 1.1. Two views of a semi-transparent surface (sculpture) by David Begbie. For curved surfaces, the dominant image feature is the apparent contour or outline. This is the projection of the locus of points on the surface which separate visible and occluded parts. The apparent contours are rich sources of geometric information. In particular, their deformation under viewer motion can be used to recover the geometry of the visible surface. The geometry of the viewpoints can also be inferred.

5

Reconstruction of surfaces from profiles

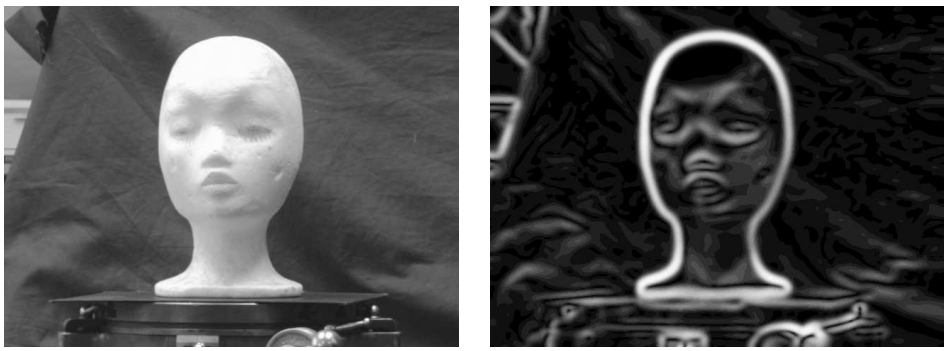
In this chapter we describe the implementation of the theory of Chapter 4 and show how to recover the geometry of a surface from an image sequence of apparent contours (profiles) from different viewpoints.

The algorithms are described in sufficient practical detail to allow the reader to implement the theory. Details are given on the localization and tracking of apparent contours; the recovery of the viewpoint geometry and camera calibration; the epipolar parametrization of the spatio-temporal family of apparent contours; and the reconstruction of surfaces. These algorithms have been used in a real-time system to recover the geometry of visible surfaces from apparent contours under known viewer motion. Examples are given.

5.1 Localization and tracking of apparent contours

A monochrome video image can be digitized into a pixel array of quantized (discrete) intensity values which can be represented by a matrix $I(u, v)$ where u, v here refer to the column and row position of the pixel respectively. A typical image is shown in Figure 5.1(a). Its size is 512×512 and the intensity value of each pixel is sampled to an accuracy of 8 bits giving 256 brightness values varying from 0 (black) to 255 (white).

The projections of surface markings, surface edges and contour generators appear as fragments of image curves across which there is an abrupt change in intensity. These image curves or contours can be extracted by first detecting the position of intensity discontinuities (*edge detection*) and then representing aggregates of edges analytically with B-splines.



(a) (b)

Fig. 5.1. Edge detection.

Edge detection

Edges can be detected by localizing the maxima of intensity gradients after convolution with a smoothing filter. The smoothing filter helps reduce the effect of image noise on the derivative operation used to find the intensity gradient. A typical edge detection algorithm (Canny 1986) involves the following the steps to smooth the image and to localize the maxima.

Algorithm 5.1.1 Edge detection.

(i) The image is smoothed by convolution with a smoothing filter $G_\sigma(i,j)$:

$$S(u,v) = \sum_{i=-n}^{n} \sum_{j=-n}^{n} G_\sigma(i,j) I(u-i, v-j)$$

where the filter kernel is of size $(2n+1)$ and is a discrete approximation for the Gaussian function:

$$G_\sigma(i,j) = \frac{1}{2\pi\sigma^2} \exp\left[-\left(\frac{i^2+j^2}{2\sigma^2} \right) \right].$$

(ii) The gradient, ∇S, of the smoothed image $S(u,v)$ is then computed at every pixel. Differentiation is performed using a finite-difference approximation.

$$\nabla S(u,v) = \begin{bmatrix} S(u+1,v) - S(u,v) \\ S(u,v+1) - S(u,v) \end{bmatrix}$$

Figure 5.1(b) shows $|\nabla S|$ for the image of Figure 5.1(a).

(iii) Edge elements, or *edgels*, are placed at locations where $|\nabla S|$ is a local maxima in the directions $\pm\nabla S$ and is above a threshold.

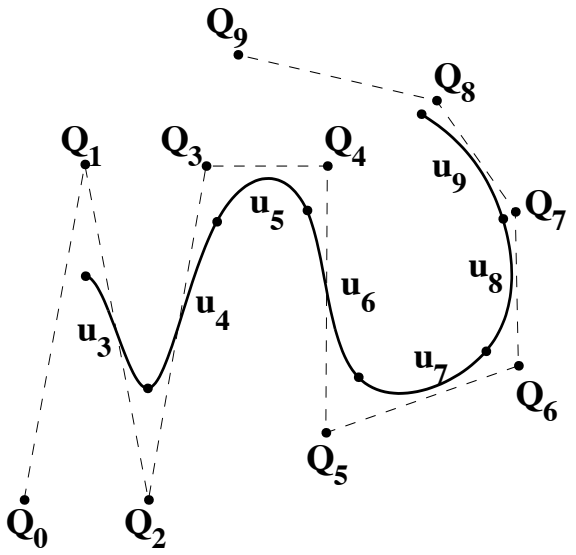


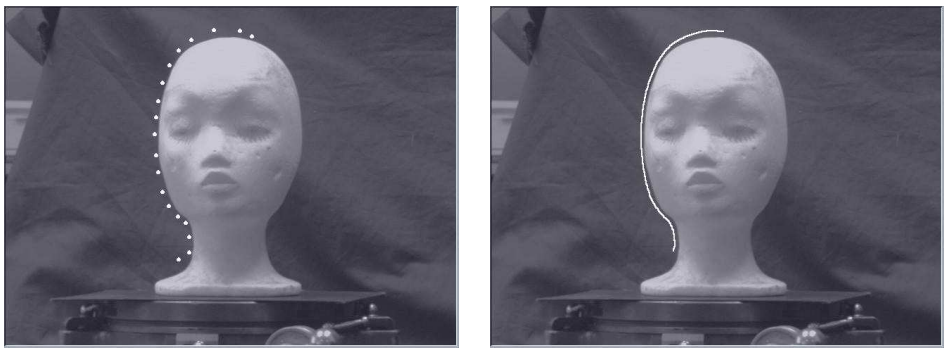
Fig. 5.2. The cubic B-spline.

The representation of the image contours as a linked chain of edge elements or edgels is not very compact and does not lend itself readily to the sub-pixel localization of the image curve and the computation of image curvature. An alternative is to automatically fit an algebraic, parametrized curve to the chain of edgels of interest. The representation is now extremely compact (only the coefficients of the curve's equation need be stored) and the smoothness and continuity of the curve are implicit.

B-spline curve representation

A natural choice for the curve parametrization is the *B-spline*, which is widely used in computer graphics (Bartels et al. 1987). A cubic B-spline is specified by $m+1$ *control points* $\mathbf{Q}_0, \mathbf{Q}_1, \ldots, \mathbf{Q}_m$ and comprises $m-2$ cubic polynomial curve segments $\mathbf{u}_3, \mathbf{u}_4, \ldots, \mathbf{u}_m$. The joining points between each curve segment are known as *knots*. The equation of each curve segment is

$$\mathbf{u}_i(s) = \frac{1}{6}\begin{bmatrix} s^3 & s^2 & s & 1 \end{bmatrix} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{Q}_{i-3} \\ \mathbf{Q}_{i-2} \\ \mathbf{Q}_{i-1} \\ \mathbf{Q}_i \end{bmatrix} \tag{5.1}$$



(a) (b)

Fig. 5.3. Initialization of the B-spline snake.

for $0 \leq s < 1$ and $3 \leq i \leq m$. See Figure 5.2. B-splines are ideal for representing curves and fitting to image edges. They may be open or closed as required, and are defined with continuity properties at each point and knot. The flexibility of the curve increases as more control points are added: each additional control point allows one more inflexion. It is also possible to use multiple knots to reduce the continuity at knots. They also exhibit *local control*: modifying the position of one control point causes only a small part of the curve to change.

A number of methods exist to fit B-splines to image edges. An automatic scheme that selects the number of control points and their image positions is described in (Cham and Cipolla 1999). In the following we describe a simpler algorithm to localize and track image contours using a variant of *snakes* or active contours (Kass et al. 1988) which use B-splines (Cipolla and Blake 1990, Blake and Isard 1998). The snake is a computational construct, a dynamic curve, which is able to track moving and deforming image contours.

Tracking image curves with B-spline snakes

B-splines can be fitted to image edges by the following iterative algorithm:

Algorithm 5.1.2 B-spline snake.

(i) Initialize a B-spline by placing control points $\mathbf{Q}_0, \mathbf{Q}_1, \ldots, \mathbf{Q}_m$ near the image edge. An example is given in Figure 5.3.

(ii) Select a number of evenly spaced sample points, N, along each segment of the B-spline, $\mathbf{u}_i(s)$. The sample points are given by $\mathbf{u}_i(s_j)$

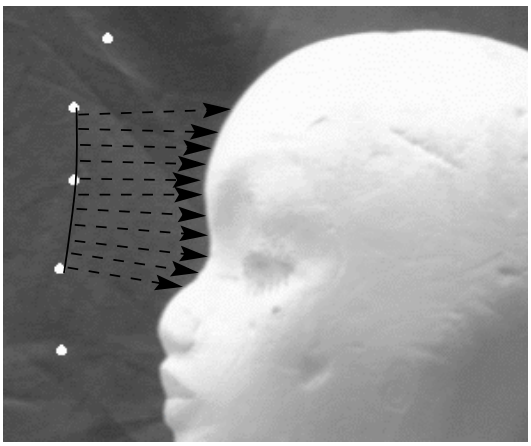


Fig. 5.4. The B-spline snake. B-spline snakes can be used to localize image contours. The control points, $\mathbf{Q}_0, \mathbf{Q}_1, \ldots, \mathbf{Q}_m$, are positioned iteratively to minimize the normal displacements (shown as arrowed vectors) between the spline segments and nearby edge features.

where $i = 3, \ldots, m$ and $j = 0, \ldots, N$. Typically $N > 10$ for each segment, $\Delta s < 0.1$ between samples.

(iii) From each sample point search along the normal to the spline for edges in the image and calculate the distance to the nearest edge. See example in Figure 5.4.

(iv) Move the control points to minimize the sum of squares of the distances between the discrete data points of the image feature and the B-spline approximation. This is a standard least squares problem and we can compute the new control point positions (Cipolla and Blake 1992).

(v) Repeat steps (iii) and (iv) until the algorithm has converged to produce a spline which localizes the image contour.

As the B-spline snake approaches the image contour the scale at which the edge search takes place can be reduced to enable accurate contour localization. After localization the same algorithm is used to track the image contour over the image sequence, provided the inter-frame image motion is less than the search (capture) window of the snake.

Since accurate measurements are required to compute surface geometry, care has been taken over sub-pixel resolution. At earlier stages of tracking, when coarse blurring (large scale) is used, the capture range of the snake is large but localization is poor – the snake may lag behind the contour. Once the snake has converged on to the contour, standard edge-detection

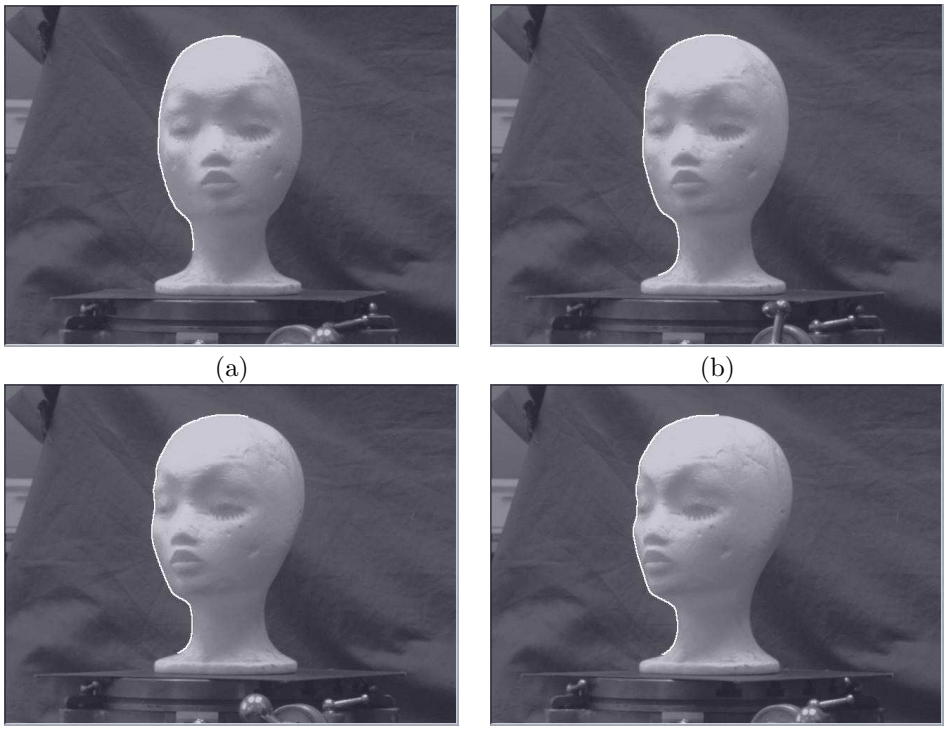


Fig. 5.5. A B-spline snake can localize and track the image contours over an image sequence.

techniques such as smoothing for sub-pixel resolution (Canny 1986) can be used to obtain accurate localization.

Figure 5.5 shows a sequence of images and a B-spline snake which has been used to localize and track the apparent contour of a surface. The output of the tracking algorithm is a family of image contours, $\mathbf{u}(s,t)$, which are parametrized by the spline parameter s and indexed by the time, t, when the image was taken. In the next section we show how to normalize and parametrize this family of image contours to recover the spatio-temporal family of apparent contours in the reference coordinate system, $\mathbf{p}(s,t)$. The latter will be used to recover the geometry of the surface.

5.2 Camera model for perspective projection onto image plane

To reconstruct the surface we require the mapping from image plane pixel coordinates, $\mathbf{u}$, to visual rays in the fixed *world* coordinate system, $\mathbf{p}$. This

is determined by transformations describing the position and orientation of the coordinate system attached to the camera relative to the world coordinate system; perspective (central) projection onto the image plane and the geometry of the CCD array. These three transformations are derived below and can be conveniently written as a 3×4 projection matrix (Roberts 1965).

Property 5.2.1 The projection matrix. *Under perspective projection the map between the three-dimensional world coordinates of a point* (X, Y, Z) *and its two-dimensional image plane pixel coordinates* (u, v) *can be written as a linear mapping in homogeneous coordinates and represented by a* 3×4 *projection matrix:*

$$\begin{bmatrix} \zeta u \\ \zeta v \\ \zeta \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}. \tag{5.2}$$

Rigid-body transformation

Consider a coordinate system $\mathbf{X} = (X, Y, Z)$ attached to the world reference frame, and another coordinate system $\mathbf{X}_c = (X_c, Y_c, Z_c)$ attached to the camera at position $\mathbf{c}(t)$, where the optical axis is aligned with Z_c. See Figure 5.6. In terms of the notation of Chapter 4 and §4.3, the camera centre $\mathbf{c}$ is the origin of the $\mathbf{X}_c$ coordinate system and we write R for the geometrical rotation which has 3×3 matrix $\mathbf{R}$. The measurements in the different coordinate systems are given by:

$$\mathbf{r} = \mathbf{X} = \mathbf{c} + \lambda \mathbf{p}, \quad \mathbf{X}_c = \lambda \mathbf{q}, \quad \mathbf{p} = R\mathbf{q}.$$

For apparent contours, $\mathbf{r}, \mathbf{X}, \mathbf{X}_c, \lambda, \mathbf{p}$ and $\mathbf{q}$ are functions of two variables s and t, while R and $\mathbf{c}$ are functions of time t alone. Recall that λ is the depth measured along the ray from $\mathbf{c}$ and the unit vector $\mathbf{p}$ is the direction (in world coordinates) of a surface point $\mathbf{r}$ from $\mathbf{c}$ while the unit vector $\mathbf{q}$ is the direction in camera coordinates of the same point.

The camera and reference coordinate systems are related by a rigid body transformation:

$$\mathbf{X} = \mathbf{R}\mathbf{X}_c + \mathbf{c} \tag{5.3}$$

$$\mathbf{X}_c = \mathbf{R}^\top(\mathbf{X} - \mathbf{c}) \tag{5.4}$$

which are conveniently represented with a rotation matrix and a translation

vector, $\mathbf{t}$, by:

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \begin{bmatrix} t_X \\ t_Y \\ t_Z \end{bmatrix} \tag{5.5}$$

where the translation vector is related to the position of the camera centre by

$$\mathbf{t} = -\mathbf{R}^\top \mathbf{c}.$$

Perspective projection onto the CCD image plane

Perspective projection onto the imaging plane followed by the conversion of image plane coordinates into CCD pixel coordinates, (u, v), can be modelled by

$$u = u_0 + \alpha_u \frac{X_c}{Z_c} \tag{5.6}$$

$$v = v_0 + \alpha_v \frac{Y_c}{Z_c} \tag{5.7}$$

where the CCD array axes are assumed aligned with the X_c and Y_c axes; (u_0, v_0) is the *principal point* (the point of intersection of the optical axis and the image plane); α_u and α_v are image scaling factors. These four parameters are known as the *internal camera parameters*. The ratio α_v/α_u is known as the aspect ratio.

Projection matrix

The relationship between image pixel coordinates and rays in Euclidean 3-space can now be expressed succinctly by introducing *homogeneous coordinates* to represent image points with 3-vectors and points in 3-space by 4-vectors, defined up to arbitrary scales (e.g. ζ). Homogeneous (projective) coordinates are often used in projective geometry and allow us to represent projective transformations as a matrix multiplications. By concatenating the matrices for the transformations described above the relationship becomes:

$$\begin{bmatrix} \zeta u \\ \zeta v \\ \zeta \end{bmatrix} = \begin{bmatrix} \alpha_u & 0 & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_X \\ r_{21} & r_{22} & r_{23} & t_Y \\ r_{31} & r_{32} & r_{33} & t_Z \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

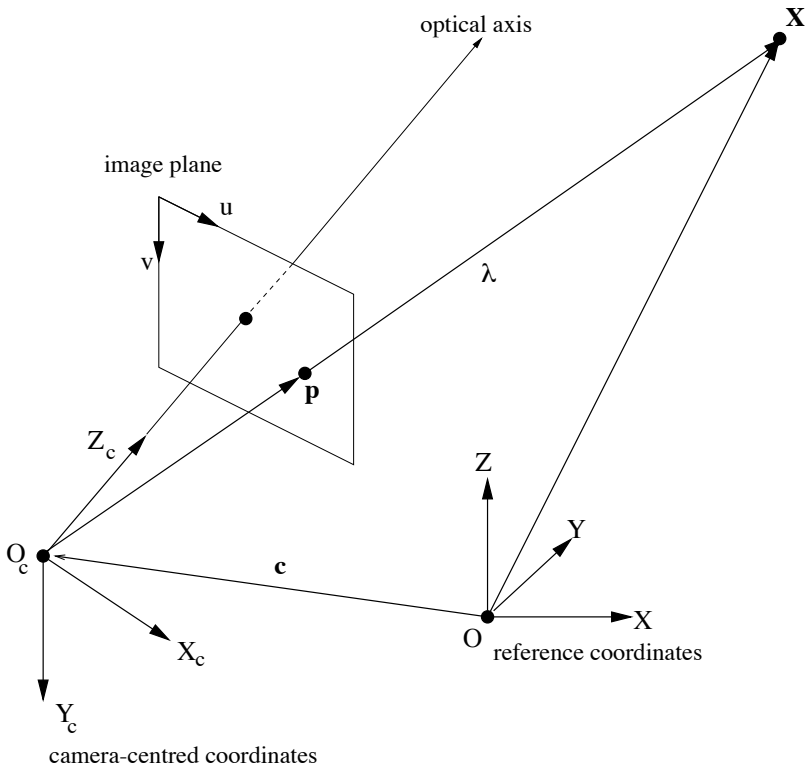
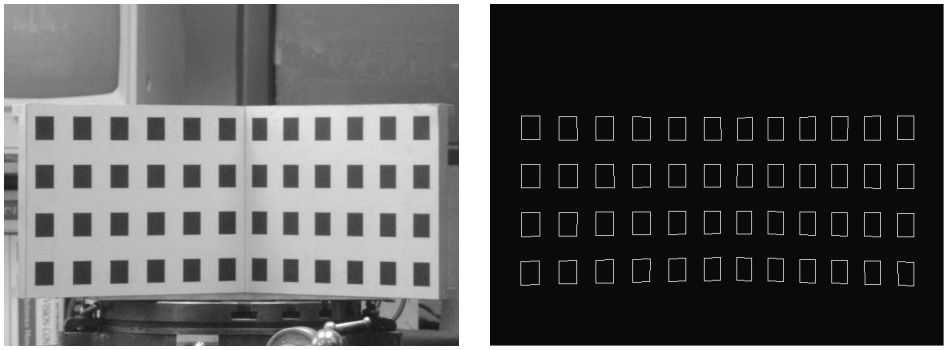


Fig. 5.6. Camera model and camera and reference coordinate systems. $\mathbf{X} = \mathbf{c} + \lambda\mathbf{p}$ and $\mathbf{X} = \mathbf{c} + \mathbf{R}\mathbf{X}_c$.



(a) (b)

Fig. 5.7. Camera calibration. A camera is calibrated by processing an image of a calibration grid (a). The image positions of known 3D points on the grid are extracted automatically. Edge detection is followed by fitting lines to the image segments. Intersections of lines are used to localize the image features to sub-pixel accuracy (b).

or more simply as the 3×4 *projection matrix* representing the perspective projection of a point in space onto a digitized image given in Property 5.2.1 where equality is defined up to an arbitrary scale.

$$\mathbf{u} = \mathbf{PX} \tag{5.8}$$

The projection matrix, $\mathbf{P}$, is not a general 3×4 matrix. It has 11 parameters (since the overall scale does not matter) and it can be decomposed into a 3×3 upper triangular matrix of camera internal parameters called the *camera calibration matrix*, $\mathbf{K}$, and a matrix representing the rigid-body motion.

The mapping from an image point to a visual ray in 3-space is expressed in homogeneous coordinates and up to an arbitrary scale by:

$$\mathbf{u} = \mathbf{Kq} \tag{5.9}$$

$$\mathbf{u} = \mathbf{KR}^\top\mathbf{p} \tag{5.10}$$

5.3 Camera model for weak perspective and orthographic projection

A useful approximation to perspective projection occurs when the field of view is narrow or the depth variation along the line of sight is small compared with the distance from the camera to the scene. The camera model can then be simplified to *weak perspective* and (5.6) and (5.7) can be re-written as:

$$u = u_0 + \alpha_u \frac{X_c}{Z_o} \tag{5.11}$$

$$v = v_0 + \alpha_v \frac{Y_c}{Z_o}. \tag{5.12}$$

The difference with perspective projection is that all image points are scaled uniformly by Z_o, the mean distance of the features of the scene to the camera centre. Weak perspective can in fact be considered as the orthographic (parallel) projection of all points onto the plane $Z_c = Z_0$ followed by a perspective projection to give a uniform inverse-depth scaling. It can be represented by the transformation:

$$\begin{bmatrix} u_a \\ v_a \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha_u & 0 & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_X \\ r_{21} & r_{22} & r_{23} & t_Y \\ 0 & 0 & 0 & Z_o \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

If the principal point of a camera is (u_0, v_0), the variation of depth in the

scene is ΔZ along the optical axis and the mean distance of the features of the scene to the camera is Z_0, the difference of the image of a point taken from a perspective camera (u, v) and its image with the weak perspective camera, (u_a, v_a) is given by

$$u - u_a = (u - u_0)\Delta Z/Z_o \tag{5.13}$$

$$v - v_a = (v - v_0)\Delta Z/Z_o. \tag{5.14}$$

When the field of view is narrow, the terms $u - u_0$ and $v - v_0$ will be small. In this case, or when the depth variation of the scene is much smaller than its mean depth, e. g. $\Delta Z/Z_0 < 0.1$, the error due to the weak perspective appproximation is negligible.

Orthographic projection can be modelled in exactly the same way but with no scaling due to depth by setting $Z_c = f$.

5.4 Camera calibration

For reconstruction we require the camera centres, $\mathbf{c}(t)$, and the rays $\mathbf{p}(s, t)$. These can be obtained from the projection matrix for each viewpoint. *Camera calibration* is the name given to the process of recovering the projection matrix from an image of a controlled scene. For example, we might set up the camera to view the calibrated grid shown in Figure 5.7(a) and automatically extract the image positions of known 3D points (Figure 5.7(b)). Each image point, (u_i, v_i), of a known calibration point, X_i, Y_i, and Z_i, generates two equation which the elements of the projection matrix must satisfy:

$$u_i = \frac{\zeta u_i}{\zeta} = \frac{p_{11}X_i + p_{12}Y_i + p_{13}Z_i + p_{14}}{p_{31}X_i + p_{32}Y_i + p_{33}Z_i + p_{34}}$$

$$v_i = \frac{\zeta v_i}{\zeta} = \frac{p_{21}X_i + p_{22}Y_i + p_{23}Z_i + p_{24}}{p_{31}X_i + p_{32}Y_i + p_{33}Z_i + p_{34}}.$$

These equations can be rearranged to give two linear equations in the 12 unknown elements of the projection matrix. For n calibration points and their corresponding image projections we have $2n$ equations:

$$\begin{bmatrix} X_1 & Y_1 & Z_1 & 1 & 0 & 0 & 0 & 0 & -u_1X_1 & -u_1Y_1 & -u_1Z_1 & -u_1 \\ 0 & 0 & 0 & 0 & X_1 & Y_1 & Z_1 & 1 & -v_1X_1 & -v_1Y_1 & -v_1Z_1 & -v_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ X_n & Y_n & Z_n & 1 & 0 & 0 & 0 & 0 & -u_nX_n & -u_nY_n & -u_nZ_n & -u_n \\ 0 & 0 & 0 & 0 & X_n & Y_n & Z_n & 1 & -v_nX_n & -v_nY_n & -v_nZ_n & -v_n \end{bmatrix} \begin{bmatrix} p_{11} \\ p_{12} \\ p_{13} \\ p_{14} \\ p_{21} \\ p_{22} \\ p_{23} \\ p_{24} \\ p_{31} \\ p_{32} \\ p_{33} \\ p_{34} \end{bmatrix} = \mathbf{0}.$$

Since there are 11 unknowns (scale is arbitrary), we need to observe at least 6 reference points to recover the projection matrix and calibrate the camera.

Numerical considerations

The equations can be solved using orthogonal least squares. First, we write the equations in matrix form:

$$\mathbf{Ax} = \mathbf{0} \tag{5.15}$$

where $\mathbf{x}$ is the 12×1 vector of unknowns (the 12 elements of the projection matrix, p_{ij}), $\mathbf{A}$ is the $2n \times 12$ matrix of measurements and n is the number of observed calibration points. A linear solution (least squares) which minimizes $||\mathbf{Ax}||$ subject to $||\mathbf{x}|| = 1$ is obtained as the unit eigenvector corresponding to the smallest eigenvalue of $\mathbf{A}^\top\mathbf{A}$. Numerically this computation is performed via the *singular value decomposition* of the matrix (Strang 1988)

$$\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^\top$$

where $\mathbf{\Lambda} = \mathrm{diag}(\sigma_1, \sigma_2, \ldots, \sigma_{12})$ is the diagonal matrix of singular values and the matrices $\mathbf{U}$ and $\mathbf{V}$ are orthonormal. The columns of $\mathbf{V}$ are the eigenvectors of $\mathbf{A}^\top\mathbf{A}$ and the least squares solution is given by the last column of $\mathbf{V}$ which is the singular vector with the smallest singular value σ_{12}. The least squares solution is, however, only approximate and should be used as the starting point for non-linear optimization: i.e. finding the parameters of the projection matrix, $\mathbf{P}$, that minimize the errors between measured image points, (u_i, v_i) and the projections onto the image plane of the reference points:

$$\min_{\mathbf{P}} \sum_i ||(u_i, v_i) - \mathbf{P}(X_i, Y_i, Z_i, 1)||^2$$

Once the projection matrix has been estimated the first 3×3 submatrix, $\mathbf{KR}^\top$, can be easily decomposed by standard matrix algorithms into an upper triangular matrix, $\mathbf{K}$, and a rotation (orthonormal) matrix (known as QR decompositon) or used directly to determine the ray in space $\mathbf{p}$ and the position of the camera centre:

$$\mathbf{p} = \mathbf{R}\mathbf{K}^{-1}\mathbf{u} \tag{5.16}$$

$$\mathbf{c} = -\mathbf{R}\mathbf{K}^{-1}(p_{14}, p_{24}, p_{34})^\top. \tag{5.17}$$

5.5 Epipolar geometry

Epipolar geometry plays a key part in the algorithms to recover the geometry of surfaces from apparent contours. The *epipolar parametrization* of the spatio-temporal family of apparent contours, $\mathbf{p}(s,t)$, introduced in Chapter 4, requires the *epipolar geometry* between successive snapshots of an apparent contour.

We briefly review the geometry of two views and describe how to compute the epipolar geometry when the cameras are calibrated. The use of uncalibrated cameras and the recovery of the epipolar geometry from apparent contours is described in Chapter 6.

The epipolar constraint

In stereo vision the projection of a world point in two calibrated viewpoints can be used to recover the three-dimensional position by triangulation. The geometry of the two views, as shown in Figure 5.8, plays a key part in helping to find correspondences by constraining the search for correspondence from a region to a line. This matching constraint is known as the *epipolar constraint*.

The epipolar constraint arises from the fact that the two rays, $\mathbf{p}$ and $\mathbf{p}'$, to a common scene point, $\mathbf{X}$, and the optical centres of the two camera (the stereo baseline, $\mathbf{t} = \Delta\mathbf{c}$) lie in a plane called the *epipolar plane*. The intersection of the epipolar plane with each image plane defines a line called an *epipolar line*. The correspondence of an image point in the first view, $\mathbf{u}$, must line on the epipolar line, $\mathbf{l}'$, in the other view shown in Figure 5.8. Using homogeneous coordinates to represent the coefficients of a line in the image as a 3-vector, the epipolar constraints in each view can be written as:

$$\mathbf{u} \cdot \mathbf{l} = 0 \tag{5.18}$$

$$\mathbf{u}' \cdot \mathbf{l}' = 0. \tag{5.19}$$

Each world point, $\mathbf{X}$, has its own epipolar plane. The family of epipolar planes define a *pencil* of epipolar lines which pass through a common point called the *epipole*, illustrated in Figure 5.9. The epipoles and pencil of epipolar lines in each view are known as the *epipolar geometry*. The epipolar geometry is completely determined by the relative position, $\mathbf{t}$, and relative orientation, $\mathbf{R}$, of the two views and the camera parameters of each camera, $\mathbf{K}$ and $\mathbf{K}'$ respectively. It does not depend on the 3D structure of the scene being viewed.

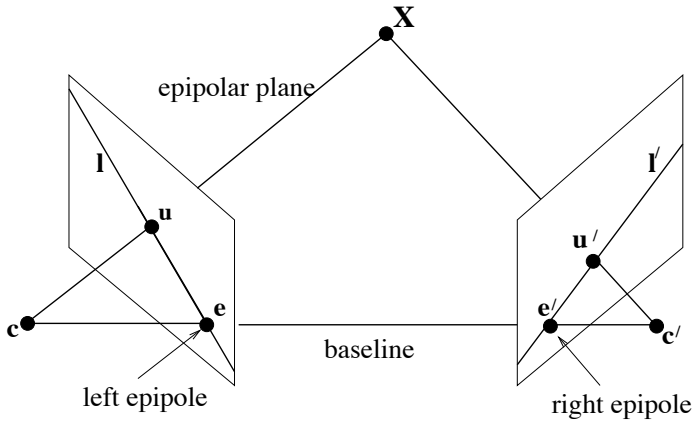


Fig. 5.8. The geometry of two views. In stereo vision an *epipolar plane* is the plane defined by a 3D point $\mathbf{X}$ and the optical centres of the two cameras. The *baseline* is the line joining the optical centres. An *epipole* is the point of intersection of the baseline with the image plane. An *epipolar line*, $\mathbf{l}$ and $\mathbf{l}'$, is a line of intersection of the epipolar plane with an image plane. It is the image in one camera of the ray from the other camera's optical centre to the point $\mathbf{X}$.

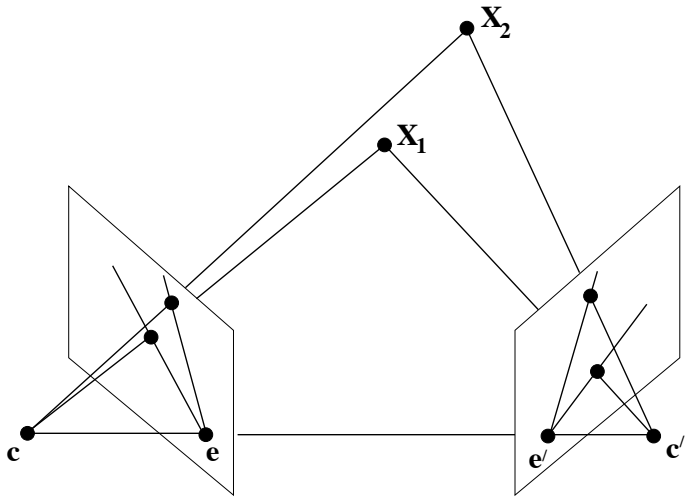


Fig. 5.9. Epipolar geometry. Each world point $\mathbf{X}$ has its own epipolar plane which rotates about the baseline. All epipolar lines intersect at the epipole.

The essential matrix

The epipolar constraint is a co-planarity constraint and can be expressed algebraically as a scalar triple product:

$$\mathbf{p}' \cdot (\mathbf{t} \wedge \mathbf{p}) = 0. \tag{5.20}$$

With out loss of generality, we can align the reference coordinate system with the second camera so that the epipolar constraint can be rewritten in terms of image positions (3-vectors in homogeneous coordinates), $\mathbf{u}$ and $\mathbf{u}'$, using (5.16):

$$\mathbf{u}'^{\top}\mathbf{K}'^{-\top}\mathbf{E}\mathbf{K}^{-1}\mathbf{u} = 0 \tag{5.21}$$

where $\mathbf{E}$ is a 3×3 matrix known as the *essential matrix* (Longuet-Higgins 1981) and is the product of a skew-symmetric or antisymmetric matrix (representing the vector product with the translation vector) and an orthonormal matrix representing the rotation between the two views:

$$\mathbf{E} = \mathbf{t} \wedge \mathbf{R} = [\mathbf{t}]_{\times}\mathbf{R}$$

where

$$[\mathbf{t}]_{\times} = \begin{bmatrix} 0 & -t_3 & t_2 \\ t_3 & 0 & -t_1 \\ -t_2 & t_1 & 0 \end{bmatrix}$$

and $\mathbf{R}$ now specifies the relative orientation between the views.

The essential matrix is of maximum rank 2. Its factorization into the product of a non-zero skew-symmetric matrix and a rotation matrix is only possible if it has two equal non-zero singular values. The other is of course equal to zero (Tsai and Huang 1984, Faugeras and Maybank 1990).

The fundamental matrix

From (5.21) we see that the epipolar geometry can be conveniently specified by introducing a matrix, $\mathbf{F}$ (Faugeras 1992)

$$\mathbf{F} = \mathbf{K}'^{-\top}\mathbf{E}\mathbf{K}^{-1}. \tag{5.22}$$

Property 5.5.1 The epipolar constraint and the fundamental matrix. *The image coordinates (projective representation using homogeneous coordinates) of all pairs of corresponding points,* $\mathbf{u}_i = (u_i, v_i, 1)^{\top}$ *and* $\mathbf{u}'_i = (u'_i, v'_i, 1)^{\top}$*, must satisfy the epipolar constraint:*

$$\mathbf{u}'^{\top}_i \mathbf{F}\mathbf{u}_i = 0 \tag{5.23}$$

or

$$\begin{bmatrix} u'_i & v'_i & 1 \end{bmatrix} \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{bmatrix} \begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} = 0 \tag{5.24}$$

where $\mathbf{F}$ *is a* 3×3 *real matrix of rank 2 which is defined up to an arbitrary scale and is known as the fundamental matrix.*

Epipolar lines and epipoles

The epipolar geometry (see Figure 5.9) is completely determined by the fundamental matrix.

Property 5.5.2 Epipolar geometry from fundamental matrix.

(i) **Epipolar lines.**
The epipolar line (represented by a homogeneous 3-vector), $\mathbf{l}'$*, corresponding to a point* $\mathbf{u}$ *in the other view is given by*

$$\mathbf{l}' = \mathbf{F}\mathbf{u} \tag{5.25}$$

and $\mathbf{u}'$ *must lie on this line to satisfy the epipolar constraint:*

$$\mathbf{u}' \cdot \mathbf{l}' = 0.$$

The epipolar line corresponding to $\mathbf{u}'$ *is given by* $\mathbf{l} = \mathbf{F}^\top \mathbf{u}'$.

(ii) **Epipoles.**
The epipole is defined as the point in each image which is common to all the epipolar lines. The left and right epipoles ($\mathbf{e}$ *and* $\mathbf{e}'$ *in homogeneous coordinates) are therefore given by the null spaces of* $\mathbf{F}$ *and* $\mathbf{F}^\top$ *respectively*

$$\mathbf{F}\mathbf{e} = \mathbf{0} \tag{5.26}$$

$$\mathbf{F}^\top \mathbf{e}' = \mathbf{0}. \tag{5.27}$$

5.6 Epipolar geometry from projection matrices

For calibrated cameras with known projection matrices it is trivial to compute the fundamental matrix and hence obtain the epipolar geometry (epipoles and epipolar lines for each image feature). We here outline a simple method by exploiting the following result.

Property 5.6.1 Projective ambiguity. *The pair of cameras and projection matrices* $\mathbf{P}$ *and* $\mathbf{P}'$ *give rise to the same fundamental matrix as the pair of cameras and projection matrices* $\mathbf{PH}$ *and* $\mathbf{P}'\mathbf{H}$ *where* $\mathbf{H}$ *is a* 4×4 *non-singular matrix.*

A simple proof can be found in (Hartley 1992 and 1994) but follows trivially from the fact that the simultaneous transformation of the projection matrices, $\mathbf{P}$ by $\mathbf{H}$ and the 3D point coordinates, $\mathbf{X}$, by $\mathbf{H}^{-1}$ leaves the image coordinates $\mathbf{u} = \mathbf{PX}$, unchanged.

Assume we are given the projection matrices for two viewpoints, $\mathbf{P}$ and $\mathbf{P}'$. The position of the optical centre of the first camera, $\mathbf{c}$, can be computed directly from the projection matrix $\mathbf{P}$ from (5.17). In homogeneous coordinates we can represent it by a 4-vector $\mathbf{C} = (\mathbf{c}^\top 1)$ so that

$$\mathbf{PC} = \mathbf{0}$$

and its projection into the second image plane defines the epipole, $\mathbf{e}'$,

$$\mathbf{e}' = \mathbf{P}'\mathbf{C}. \tag{5.28}$$

We can also compute the pseudo-inverse, $\mathbf{P}^+$, of the projection matrix $\mathbf{P}$,

$$\mathbf{P}^+ = \mathbf{P}^\top(\mathbf{P}\mathbf{P}^\top)^{-1}, \tag{5.29}$$

such that multiplication with the first projection matrix gives the identity matrix, $\mathbf{I}$, and multiplication with the second projection matrix gives a 3×3 matrix (a two-dimensional projective transformation), $\mathbf{M}$:

$$\mathbf{I} = \mathbf{P}\,\mathbf{P}^+ \tag{5.30}$$

$$\mathbf{M} = \mathbf{P}'\mathbf{P}^+ \tag{5.31}$$

The two projection matrices have in this way been normalized to have the special forms:

$$\mathbf{PH} = [\mathbf{I} \mid \mathbf{0}] \tag{5.32}$$

$$\mathbf{P}'\mathbf{H} = [\mathbf{M} \mid \mathbf{e}'] \tag{5.33}$$

These normalized projection matrices are known as *canonical cameras*. From Property 5.6.1 both the original projection matrices and these normalized forms have the same fundamental matrix $\mathbf{F}$ given by (see Property 6.3.1):

$$\mathbf{F} = [\mathbf{e}']_\times \mathbf{M}. \tag{5.34}$$

An example of the epipolar geometry of two discrete views computed from calibrated projection matrices using (5.28) - (5.31) and (5.34) is shown in Figure 5.10.

For uncalibrated cameras we do not have the projection matrices and hence $\mathbf{E}$, $\mathbf{K}$ and $\mathbf{K}'$ are unknown *a priori*. The fundamental matrix and epipolar geometry, however, can still be estimated from *point* and curve correspondences between the two views (see Chapter 6).

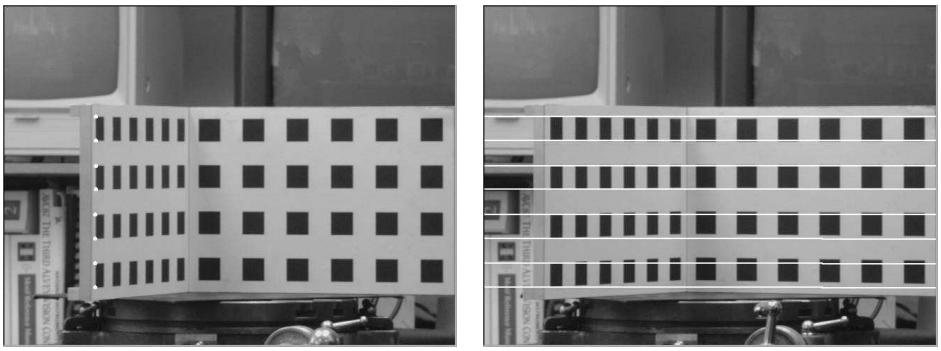


Fig. 5.10. Epipolar geometry computed from known projection matrices. Selected image points are shown in the left view with corresponding epipolar lines shown in the right view. The corresponding image feature satisfies the epipolar constraint.

5.7 Reconstruction of surfaces

Implementation of the epipolar parametrization

In the *epipolar* parametrization of the spatio-temporal image and surface, a point on an apparent contour in the first image is *matched* to a point in successive images (in an infinitesimal sense) by searching along the corresponding epipolar lines. This allows us to extract t-parameter curves ($\mathbf{u}(s_0, t)$ or $\mathbf{p}(s_0, t)$) from the spatio-temporal image. As shown in Chapter 4, depth and surface curvature are then computed from first and second-order temporal derivatives of this t-parameter spatio-temporal curve by equations (4.5) and (4.11). This requires a dense (continuous) image sequence.

In practice the epipolar parametrization and the reconstruction can be implemented in a variety of ways. In (Cipolla and Blake 1992) and (Cipolla 1995) two simple methods are described for special cases. For *pure translation* perpendicular to the optical axis and with a dense (continuous) image sequence it is possible to recover the depth and surface curvature from the first and second-order derivatives of the spatio-temporal image trajectories directly (see Figure 5.11). For *linear* motion and a minimum of three discrete views a simple numerical method was proposed to estimate the depth of the contour generators and the *osculating circle* in each epipolar plane by assuming that the curvature of the epipolar curve, $\mathbf{r}(s_0, t)$, is locally constant. See Figure 5.12. This approximation was also exploited by (Vallaint and Faugeras 1992) and (Szeliski and Weiss 1998).

We choose to implement the theory presented in Chapter 4 directly by estimating temporal derivatives from measurements in the discrete views of the apparent contours, $\mathbf{u}(s, t_0), \mathbf{u}(s, t_1), \ldots, \mathbf{u}(s, t_n)$, which are indexed by

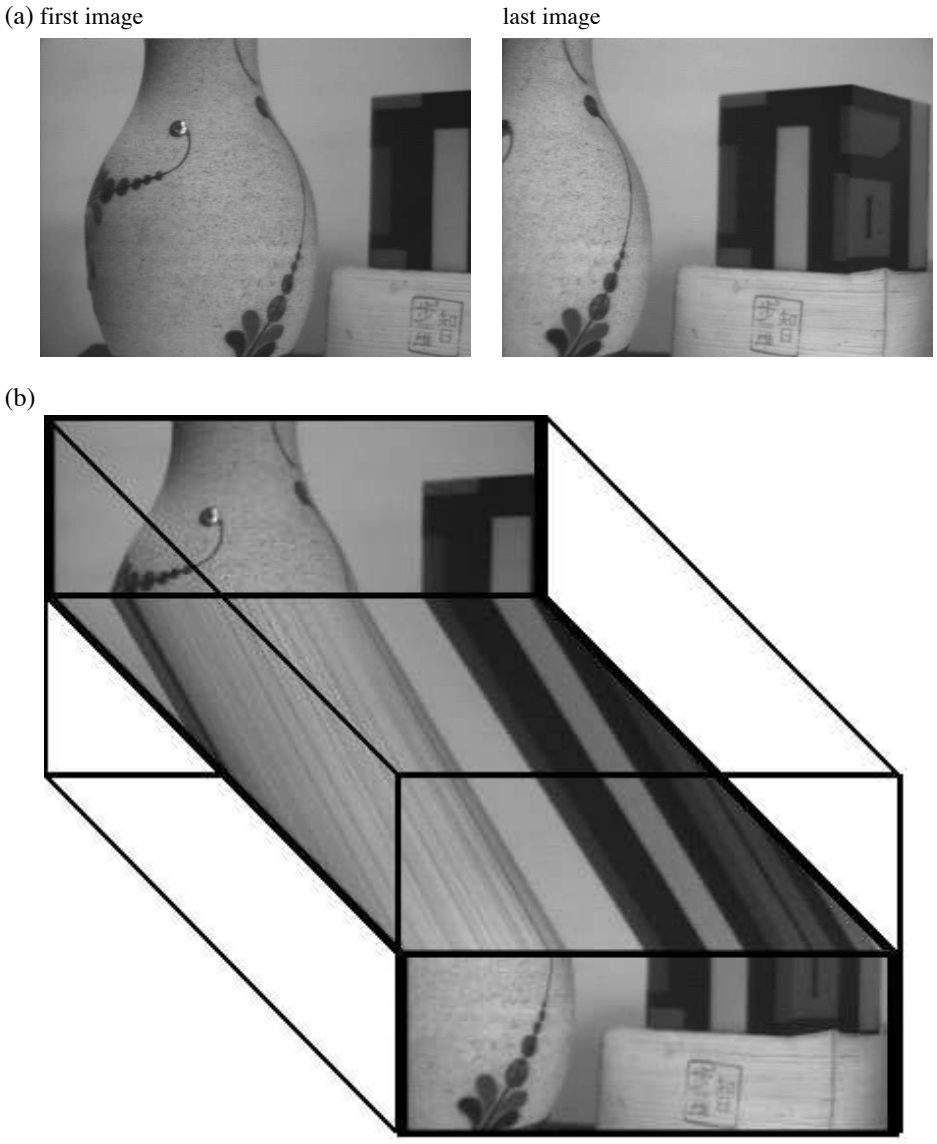


Fig. 5.11. 3D spatio-temporal image. (a) The first and last image from an image sequence taken from a camera mounted on a robot arm and moving horizontally from left to right without rotation. (b) The 3D spatio-temporal image formed from the image sequence piled up sequentially with time. The top of the first image and the bottom of the last image are shown along with the spatio-temporal cross-section corresponding to the same epipolar plane. For simple viewer motions consisting of camera translations perpendicular to the optical axis the spatio-temporal cross-section image is formed by storing the scan-lines (epipolar lines) for a given epipolar plane sequentially in order of time. The t-parameter curves $\mathbf{p}(s_0, t)$ are easily extracted from this spatio-temporal image and its first and second derivatives can be used to recover depth and surface curvature respectively (Cipolla and Blake 1992).

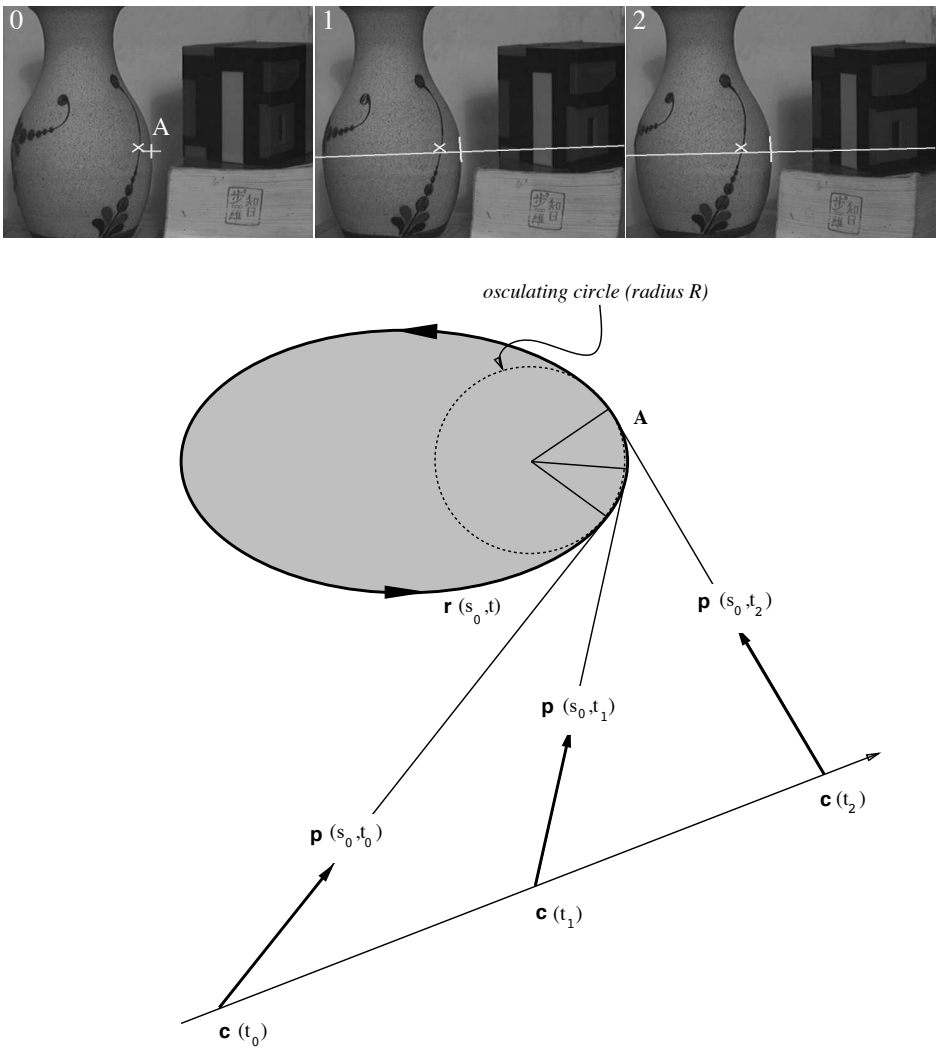


Fig. 5.12. The epipolar plane. Each view defines a tangent to the epipolar curve $\mathbf{r}(s_0, t)$. For linear camera motion and epipolar parametrization the rays and $\mathbf{r}(s_0, t)$ lie in a plane. If $\mathbf{r}(s_0, t)$ can be approximated locally by its osculating circle, it can be uniquely determined from measurements in three views, $\mathbf{p}(s_0, t_0)$, $\mathbf{p}(s_0, t_1)$ and $\mathbf{p}(s_0, t_2)$. For curvilinear motion the epipolar geometry is continuously changing and the epipolar curve is no longer planar (Cipolla and Blake 1992).

time, t_i and the corresponding camera position, $\mathbf{c}(t_i)$. Figure 5.5 shows 4 of 36 images taken by rotating an object in front of a fixed camera. This is equivalent to considering the image sequence to have been obtained by the camera rotating about the same axis of rotation, i.e. circular motion.

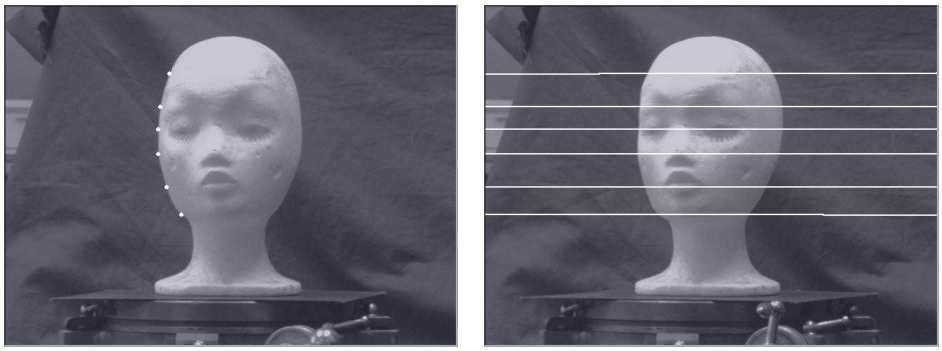


Fig. 5.13. In the epipolar parametrisation of apparent contours, *correspondence* between points on successive snapshots of an apparent contours is set by matching along epipolar lines.

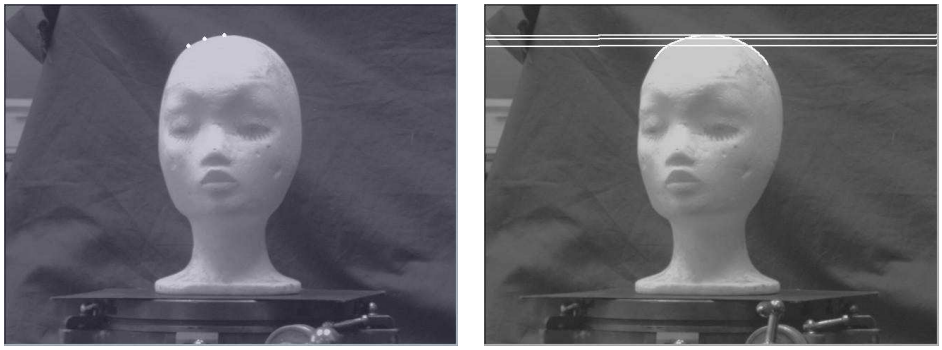


Fig. 5.14. The epipolar parametrisation is degenerate at frontier points where the epipolar plane is tangent to the surface. Frontier points appear as epipolar tangencies – the epipolar lines are tangent to the apparent contour. In the vicinity of an epipolar tangency the parametrization is sensitive to errors in the epipolar geometry and the localization of the image contour. Reconstruction is therefore poor near a frontier point.

The apparent contours are extracted and tracked over the sequence using B-splines snakes.

The family of image contours is then parametrized using the known epipolar geometry obtained by first calibrating the sequence of discrete viewpoints by viewing a calibration grid as described in the previous section and illustrated in Figure 5.10. The projection matrices for each view can be used to obtain camera positions, $\mathbf{c}(t_i)$, and the epipolar geometry as described in §5.6. The epipolar geometry is then used to parametrize the image contours as follows.

Points are selected on the apparent contours in the first view, e.g. $\mathbf{u}(s_i, t_0)$,

shown in Figure 5.13(a), and the calibrated epipolar geometry is used to induce a parametrization of the apparent contour in the other views, e.g. $\mathbf{u}(s_i, t_1), \ldots, \mathbf{u}(s_i, t_n)$. An example is shown in Figure 5.13(b). This requires finding the intersection of the cubic B-spline representing the apparent contour and the epipolar line. This is done analytically and multiple solutions are disambiguated by using ordering and disparity gradient constraints as in stereo vision. Particular care must be taken at an epipolar tangency (frontier point). See Figure 5.14. The known (calibrated) projection matrices for each viewpoint can then be used to convert image plane pixel coordinates, $\mathbf{u}(s_i, t_1), \ldots, \mathbf{u}(s_i, t_n)$, to rays in 3-space, $\mathbf{p}(s_i, t_1), \ldots, \mathbf{p}(s_i, t_n)$, as described in §5.2.

Recovery of depth

The depth is then estimated by a finite-difference (discrete) approximation to the infinitesimal analysis presented in (4.5). For each apparent contour point (indexed by spline parameter, s_0) in each view (indexed by time, t_i) an approximation to the depth to the contour generator can be computed from:

$$\lambda(s_0, t_i) \approx -\frac{\Delta\mathbf{c} \cdot \mathbf{n}}{\Delta\mathbf{p} \cdot \mathbf{n}} \tag{5.35}$$

where $\Delta\mathbf{c} = \mathbf{c}(t_{i+1}) - \mathbf{c}(t_i)$ and $\Delta\mathbf{p} = \mathbf{p}(s_0, t_{i+1}) - \mathbf{p}(s_0, t_i)$. The surface normal, $\mathbf{n} = \mathbf{n}(s, t_{i+1})$, is estimated directly from the apparent contour in a single view as described by Property 3.6.1.

The use of finite-differences introduces an error. Equation (5.35) is in fact equivalent to estimating the distance to the surface point by *triangulation* as the intersection of two rays. This is, of course, exact in the stereo reconstruction of a fixed point from its correspondences in two discrete views but is an approximation when the views are of contour generators except in the infinitesimal limit as $\Delta\mathbf{c} \to 0$. See Figure 4.9.

The error introduced by using a finite-difference approximation can be easily quantified in the case of circular motion. See Figure 5.15. By assuming that the radius of curvature R along the epipolar curve is locally constant, the error in the recovered distance, $\Delta\lambda$, is given by:

$$\Delta\lambda = R \tan(\frac{\varphi}{2}). \tag{5.36}$$

The error ξ, which is the distance between the reconstructed point and the

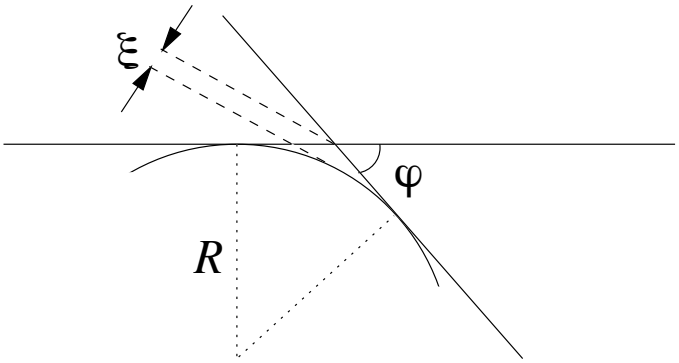


Fig. 5.15. The error ξ between the reconstructed point and the surface is related to the radius of curvature R and the angle φ between the viewing rays.

surface, is however given by

$$\xi = \left(\sec(\frac{\varphi}{2}) - 1\right) R \approx \frac{\varphi^2}{8} R \tag{5.37}$$

where φ is the angle between the viewing directions (see Figure 5.15). If the camera is far from a rotating object, φ can be approximated by the angle of rotation ω. For ω equals 10°, the error will be 0.38% of the radius R, which will be negligible for small values of R.

Except for the degenerate cases listed in §4.6, we can use the epipolar parametrization of the family of apparent contours to recover the two families of parametric curves. The s-parameter curves are the contour generators from the different viewpoints and the t-parameter curves are the epipolar curves formed by the intersection of a pencil of epipolar planes defined by the camera centres in adjacent views. These surface curves form a *conjugate* grid, shown in Figure 5.16 and Figure 5.17.

The reconstructed surface from the image sequence of Figure 5.5 is shown in Figure 5.17 as a grid of 36 contour generators and 100 epipolar curves. The reconstruction is surprisingly accurate given that the views are discrete and not continuous. Figure 5.18 shows another example of a reconstruction using the epipolar parametrisation but in which greater care is taken in computing the spatio-temporal derivatives (Boyer and Berger 1997).

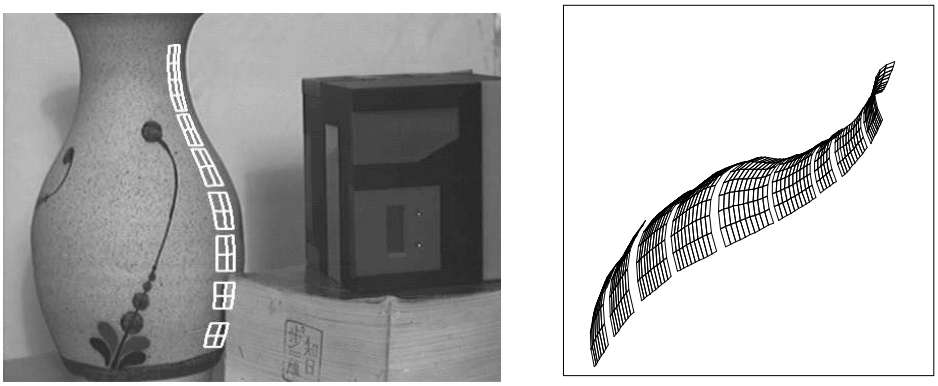


Fig. 5.16. Recovery of surface strip in vicinity of apparent contour. The surface is recovered as a family of s-parameter curves, $\mathbf{r}(s, t_i)$ – the contour generators – and t-parameter curves, $\mathbf{r}(s_0, t)$ – portions of the *osculating* circles measured in each epipolar plane. The strip is shown projected into the image of the scene from a different viewpoint and after extrapolation (Cipolla and Blake 1992).

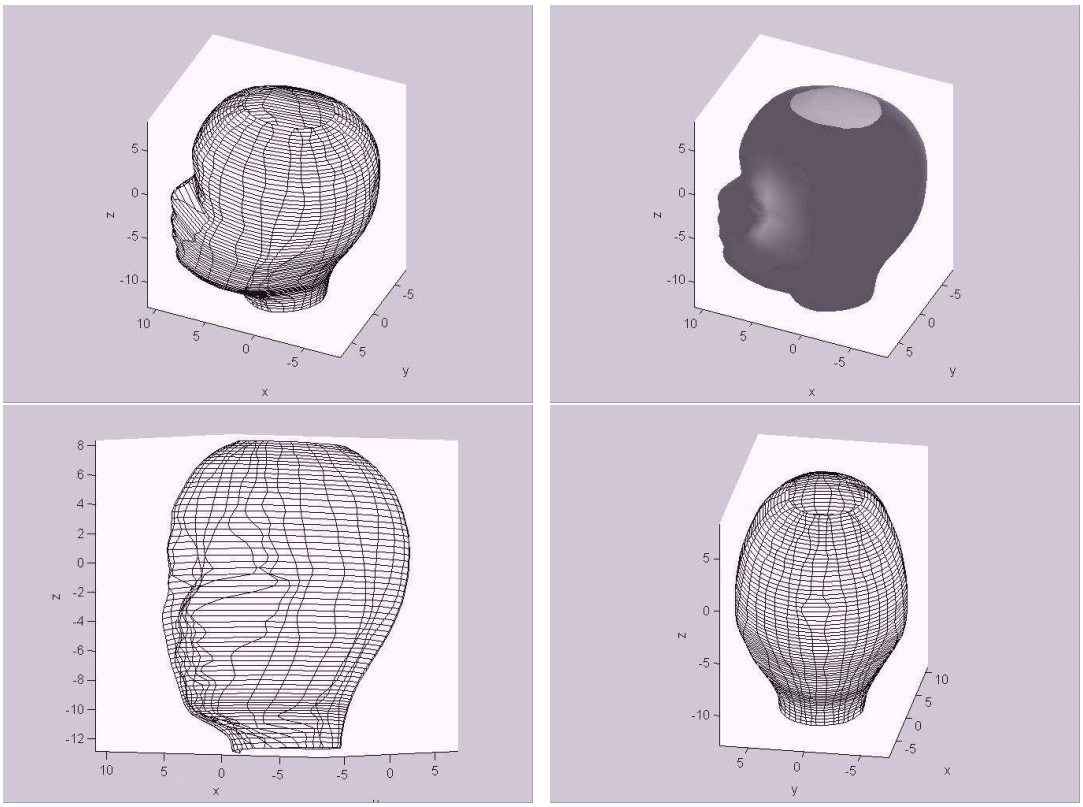


Fig. 5.17. The reconstruction of the head from 36 views of the apparent contour under circular motion. Three discrete views of the 36 reconstructed contour generators and epipolar curves are shown. The reconstructed surface is also shown shaded.

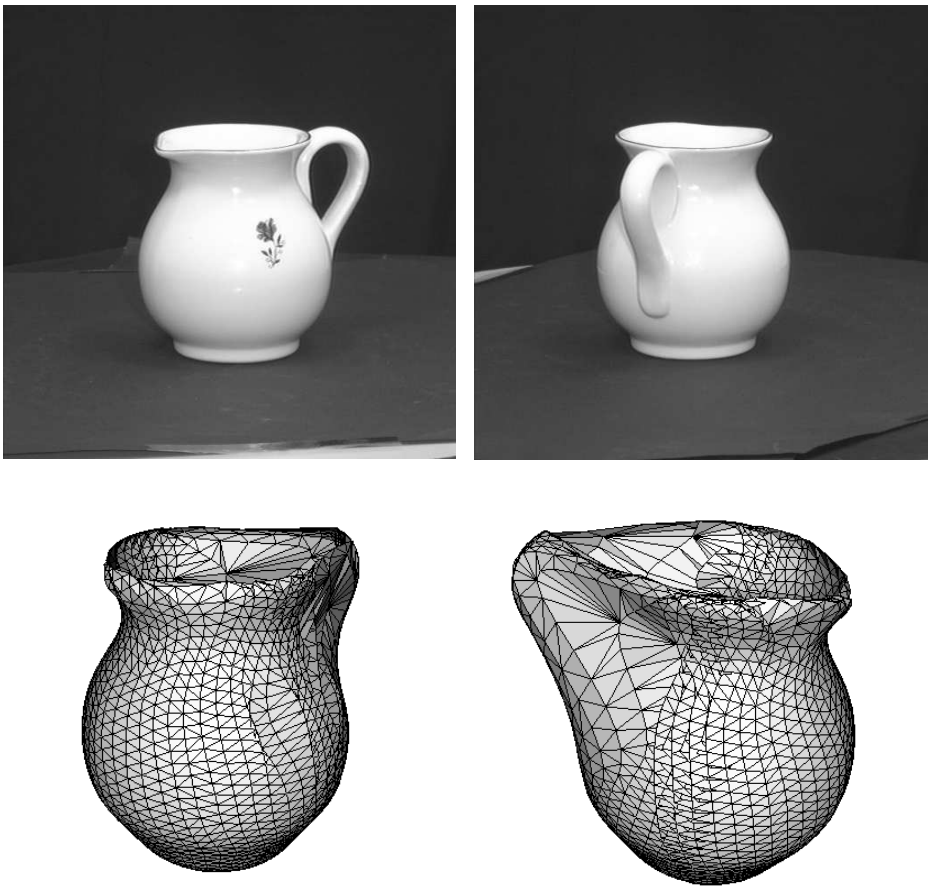


Fig. 5.18. Reconstruction of a vase using the epipolar parametrization and the method of triangulation described in (Boyer and Berger 1997).

6

Recovery of viewer motion from profiles

In the previous chapter we described the algorithms and their practical implementation to recover the geometry of visible surfaces from the deformation of apparent contours (profiles) under *known* viewer motion. The epipolar geometry between the distinct viewpoints played an important part in the parametrization and recovery of the surface. In this chapter we describe how the epipolar geometry (and hence the camera motion) can be recovered from the deformation of apparent contours when the viewer motion is not known a priori.

The recovery of the structure and motion from point correspondences has attracted considerable attention and many practical algorithms exist to recover both the spatial configuration of the points and the viewer motion compatible with the views. These are briefly reviewed in §6.1 to §6.3 before showing how the deformation of apparent contours, and in particular frontier points, can be used to recover viewer motion. A set of working algorithms are then presented which are able to recover viewer motion and which exploit this to reconstruct surfaces. The special case of circular motion, which is often used in 3D model acquisition and hence has important practical applications, is described in greater detail.

6.1 The fundamental matrix from point correspondences

For uncalibrated cameras we do not have the projection matrices and hence the essential matrix parameters, $\mathbf{t}$ and $\mathbf{R}$, and the camera calibration parameters, $\mathbf{K}$ and $\mathbf{K}'$, are unknown a priori. The fundamental matrix, however, can be estimated from *point* correspondences between the two views. We briefly describe the algorithms used for computing the epipolar geometry from point correspondences (also reviewed in Zhang 1998) before describing the extension to curves and apparent contours.

From the epipolar constraint (5.24) we see that each point correspondence, $\mathbf{u}_i = (u_i, v_i, 1)^\top$ and $\mathbf{u'}_i = (u'_i, v'_i, 1)^\top$, generates one constraint on the epipolar geometry which can be expressed in terms of the elements of the fundamental matrix $\mathbf{F}$:

$$\begin{bmatrix} u'_i & v'_i & 1 \end{bmatrix} \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{bmatrix} \begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} = 0.$$

For n pairs of correspondences, the constraints can be rearranged as linear equations in the 9 unknown elements of the fundamental matrix:

$$\begin{bmatrix} u'_1u_1 & u'_1v_1 & u'_1 & v'_1u_1 & v'_1v_1 & v'_1 & u_1 & v_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ u'_nu_n & u'_nv_n & u'_1 & v'_nu_n & v'_nv_n & v'_n & u_n & v_n & 1 \end{bmatrix} \begin{bmatrix} f_{11} \\ f_{12} \\ f_{13} \\ f_{21} \\ f_{22} \\ f_{23} \\ f_{31} \\ f_{32} \\ f_{33} \end{bmatrix} = \mathbf{0}$$

or in matrix form:

$$\mathbf{Af} = \mathbf{0}$$

where $\mathbf{A}$ is an $n \times 9$ measurement matrix, and $\mathbf{f}$ represents the elements of the fundamental matrix as a 9-vector. Given 8 or more correspondences a solution† can be found by least squares as the unit eigenvector ($\mathbf{f}$ is defined up to an arbitrary scale) corresponding to the minimum eigenvalue of $\mathbf{A}^\top\mathbf{A}$. A unique solution is obtained unless the points and the camera centres lie on a ruled quadric or all the points lie on a plane (Faugeras and Maybank 1990).

The computation can be poorly conditioned and it is important to precondition the image points by normalizing them to improve the condition number of $\mathbf{A}^\top\mathbf{A}$ before estimating the elements of the fundamental matrix by singular value decomposition (Hartley 1998).

† Note that the fundamental matrix has only 7 degrees of freedom since its determinant must be zero. A non-unique solution can be obtained from only 7 point correspondences and is described in (Huang and Netravali 1994).

Parametrization of the fundamental matrix

Two steps can be taken to improve the solution. The most important requires enforcing the rank 2 property of the fundamental matrix. This can be achieved by a suitable parametrization of $\mathbf{F}$.

The epipolar geometry between two uncalibrated views is completely determined by 7 independent parameters: the position of the epipoles in the two views, $\mathbf{e} = (u_e, v_e, 1)^\top$ and $\mathbf{e}' = (u'_e, v'_e, 1)^\top$, and the 3 parameters of the one-dimensional projective transformation† relating the pencil of epipolar lines in view 1 to those in view 2 (Luong and Faugeras 1996),

$$\tau'_i = -\frac{h_2\tau_i + h_1}{h_4\tau_i + h_3} \tag{6.1}$$

where τ_i and τ'_i represent the directions (as the gradient of a line) of a pair of corresponding epipolar lines, $\mathbf{l}_i$ and $\mathbf{l}'_i$, in the first and second images respectively. Namely:

$$\tau_i = \frac{v_i - v_e}{u_i - u_e} \tag{6.2}$$

$$\tau'_i = \frac{v'_i - v'_e}{u'_i - u'_e}. \tag{6.3}$$

The transformation of epipolar lines between views is sometimes known as the *epipolar transformation* and is fixed by 3 pairs of epipolar line correspondences. The correspondence of any additional epipolar line is completely determined since it must preserve the cross-ratio of the 4 epipolar planes and corresponding epipolar lines. See (Luong and Faugeras 1996) and Figure 6.1.

Substituting (6.2) and (6.3) into (6.1) for the image coordinates of a pair of corresponding points results in the epipolar constraint and leads to the following minimal parametrization of the fundamental matrix:

$$\mathbf{F} = \begin{bmatrix} h_1 & h_2 & -u_e h_1 - v_e h_2 \\ h_3 & h_4 & -u_e h_3 - v_e h_4 \\ -u'_e h_1 - v'_e h_3 & -u'_e h_2 - v'_e h_4 & u_e u'_e h_1 + v_e u'_e h_2 + u_e v'_e h_3 + v_e v'_e h_4 \end{bmatrix} \tag{6.4}$$

This parametrization will be exploited later when apparent contours are used instead of point correspondences to estimate the epipolar geometry.

† This one-dimensional projective transformation (also known as a collineation or homography) can be represented by a 2×2 matrix in homogeneous coordinates.

Optimization

Another improvement requires finding the 7 independent parameters of the fundamental matrix which minimize the distances between the image points and their epipolar lines.

Property 6.1.1 Geometric error using epipolar distances. *The geometric distance between an image point* $\mathbf{u}'$ *and the epipolar line,* $\mathbf{l}' = \mathbf{F}\mathbf{u}$ *is given by:*

$$\frac{(\mathbf{u}'^{\top}_i \mathbf{F}\mathbf{u}_i)^2}{(\mathbf{F}\mathbf{u}_i)_1^2 + (\mathbf{F}\mathbf{u}_i)_2^2} \tag{6.5}$$

A suitable cost function, C, consisting of the sum of the squared geometric distances (defined above) between image points and their epipolar lines in both images (Luong and Faugeras 1996),

$$C = \sum_i \left(\frac{1}{(\mathbf{F}\mathbf{u}_i)_1^2 + (\mathbf{F}\mathbf{u}_i)_2^2} + \frac{1}{(\mathbf{F}^{\top}\mathbf{u}'_i)_1^2 + (\mathbf{F}^{\top}\mathbf{u}'_i)_2^2} \right) (\mathbf{u}'^{\top}_i \mathbf{F}\mathbf{u}_i)^2$$

can be minimized by non-linear optimization techniques (Press et al. 1988).

6.2 Recovery of the projection matrices and viewer motion

As shown above it is possible to recover the epipolar geometry (via the fundamental matrix) from point correspondences in the case of uncalibrated cameras. Nevertheless we must recover the projection matrices corresponding to each viewpoint if we are to attempt reconstruction.

Factorization of the essential matrix

If the camera internal parameters, $\mathbf{K}$ and $\mathbf{K}'$, are known the viewer motion and the projection matrices are determined by the epipolar geometry. We can transform the recovered fundamental matrix into an essential matrix (5.22):

$$\mathbf{E} = \mathbf{K}'^{\top}\mathbf{F}\mathbf{K} \tag{6.6}$$

and decompose this matrix into a skew-symmetric matrix corresponding to translation and an orthonormal matrix corresponding to the rotation between the views:

$$\mathbf{E} = [\mathbf{t}]_{\times}\mathbf{R}. \tag{6.7}$$

The latter is in fact only possible if the the essential matrix has rank 2 and two equal singular values (Tsai and Huang 1984). This property turns out

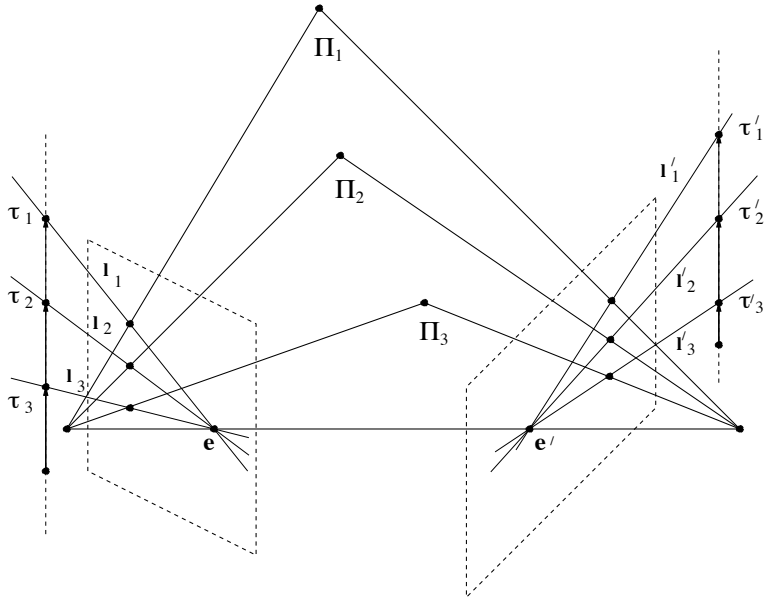


Fig. 6.1. The epipolar geometry of an uncalibrated stereo pair of images is completely specified by the image positions of the epipoles and 3 pairs of corresponding epipolar lines. The projective parameters τ and τ' represent the intersection of the epipolar line and the line at infinity. The directions in the 2 views are related by a one-dimensional projective transformation (homography).

to be very important in recovering constraints on the internal parameters of the cameras when they are uncalibrated. The difference in the two singular values can be used to refine the camera parameters. In fact, each fundamental matrix places two quadratic constraints on the internal calibration parameters (the Kruppa equations) which can be used to estimate, for example, the scale factors of the two cameras. This is known as self-calibration (Maybank and Faugeras 1992 and Hartley 1992).

The translation vector, $\mathbf{t}$, which can only be recovered up to an unknown magnitude, can be found as the unit eigenvector corresponding to the smallest eigenvalue of $\mathbf{E}\mathbf{E}^\top$ since it must satisfy

$$\mathbf{E}^\top \mathbf{t} = 0.$$

The rotation can then be obtained as the orthonormal matrix which minimizes the matrix Frobenius norm

$$||\mathbf{E} - [\mathbf{t}_\times]\mathbf{R}||^2$$

which can be solved linearly if we represent the rotation with a quaternion (Horn 1987).

Numerical considerations

An alternative numerical approach is to perform the singular value decomposition (Strang 1988) of the essential matrix (Hartley 1992):

$$\mathbf{E} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^\top \tag{6.8}$$

where $\mathbf{\Lambda} = \mathrm{diag}(\sigma_1, \sigma_2, \sigma_3)$ and the matrices $\mathbf{U}$ and $\mathbf{V}$ are orthogonal. The decomposition into a translation vector and the rotation between the two views requires that $\sigma_1 = \sigma_2 \neq 0$ and $\sigma_3 = 0$. The nearest (in the sense of minimizing the Frobenius norm between the two matrices) essential matrix with the correct properties can be obtained by setting the two largest singular values to be equal to their average and the smallest one to zero (Hartley 1992). The translation and axis and angle of rotation can then be obtained directly up to arbitrary signs and unknown scale for the translation:

$$[\mathbf{t}]_\times = \mathbf{U} \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \mathbf{U}^\top \tag{6.9}$$

$$\mathbf{R} = \mathbf{U} \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \mathbf{V}^\top \tag{6.10}$$

The projection matrices follow directly from the recovered translation and rotation by aligning the reference coordinate system with the first camera to give:

$$\begin{aligned} \mathbf{P} &= \mathbf{K}[\mathbf{I} \mid \mathbf{0}] \\ \mathbf{P}' &= \mathbf{K}'[\mathbf{R} \mid \mathbf{t}]. \end{aligned}$$

Four solutions are possible due to the arbitrary choice of signs for translation, $\pm\mathbf{t}$, and rotation, $\mathbf{R}$ or $\mathbf{R}^\top$. The correct solution is easily disambiguated by ensuring that reconstructed points lie in front of the cameras.

6.3 Recovery of the projection matrices for uncalibrated cameras

If the camera calibration matrices are unknown the projection matrices can not be uniquely recovered from the epipolar geometry of two views alone. In fact we will see that they can only be recovered up to an arbitrary 3D projective transformation, known as a projective ambiguity.

From (5.22) it follows that the fundamental matrix can, like the essential

matrix, be factorized into a skew-symmetric matrix corresponding to translation and a 3×3 non-singular matrix (ignoring arbitrary scalings of the elements of $\mathbf{F}$):

$$\begin{aligned} \mathbf{F} &= \mathbf{K'}^{-\top}[\mathbf{t}]_\times \mathbf{R}\mathbf{K}^{-1} \\ &= [\mathbf{K'}\mathbf{t}]_\times \mathbf{K'}\mathbf{R}\mathbf{K}^{-1} \\ &= [\mathbf{e'}]_\times \mathbf{M}_\infty \end{aligned} \tag{6.11}$$

where

$$\mathbf{M}_\infty = \mathbf{K'}\mathbf{R}\mathbf{K}^{-1} \tag{6.12}$$

is a 2D projective transformation (homography) which maps points on the plane at infinity in one image to the other (Luong and Vieville 1996).

The factorization of the fundamental matrix as the product of a skew-symmetric matrix and a non-singular matrix $\mathbf{M}$ is not unique since there is a 3 parameter family of matrices $\mathbf{M}$ (which represents the 2D projective transformation between views induced by different planes) such that:

$$\mathbf{M} = \mathbf{M}_\infty + \mathbf{e'}\mathbf{v}^\top$$

where $\mathbf{v}$ can be arbitrarily chosen to give a different projective transformation but the same fundamental matrix (Hartley 1994 and Luong and Vieville 1996)†.

Property 6.3.1 Factorization of the fundamental matrix. *The fundamental matrix can be factorized into a skew-symmetric matrix and a* 3×3 *non-singular matrix,* $\mathbf{M}$*:*

$$\mathbf{F} = [\mathbf{e'}]_\times \mathbf{M}$$

where $\mathbf{e'}$ *is equivalent to the epipole in the second view:*

$$\mathbf{F}^\top \mathbf{e'} = \mathbf{0}$$

and $\mathbf{M}$ *can be chosen from the 3-parameter family (defined by the arbitrarily choice of* $\mathbf{v}$*) of homographies given by:*

$$\mathbf{M} = [\mathbf{e'}]_\times \mathbf{F} + \mathbf{e'}\mathbf{v}^\top.$$

† Note the relationship with the minimal parametrization introduced in (6.4). A 3-parameter family of two-dimensional projective transformations, $\mathbf{M}$, representing the transformation between views induced by points on a plane, can be recovered from the pair of epipoles and 3 pairs of corresponding epipolar lines. The epipoles must satisfy $\mathbf{e'} = \mathbf{M}\mathbf{e}$ while the epipolar lines must pass through the epipoles and satisfy $\mathbf{l}'_i = \mathbf{M}^{-\top}\mathbf{l}_i$.

Singular value decomposition of the fundamental matrix

As with the essential matrix, we can factorize the fundamental matrix into a skew-symmetric component and a non-singular matrix by analysing its singular value decomposition:

$$\mathbf{F} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^\top$$

where $\mathbf{\Lambda} = \mathrm{diag}(r, s, 0)$. The skew-symmetric component can be recovered from:

$$[\mathbf{e}']_\times = \mathbf{U} \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \mathbf{U}^\top \tag{6.13}$$

in exactly the same way as with calibrated cameras. The non-singular matrix $\mathbf{M}$ is no longer an orthogonal transformation and is not uniquely defined. As shown by Property 6.3.1, the homography (two-dimensional projective transformation) is defined up to an arbitrary choice of parameters, here described by $\{\alpha, \beta, \gamma\}$:

$$\mathbf{M} = \mathbf{U} \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r & 0 & 0 \\ 0 & s & 0 \\ \alpha & \beta & \gamma \end{bmatrix} \mathbf{V}^\top. \tag{6.14}$$

Canonical cameras and projective ambiguity

The factorization of the fundamental matrix can be used to compute the canonical cameras – the normalized projection matrices – given by (5.32) and (5.33)

$$\begin{aligned} \mathbf{P}\mathbf{H} &= [\mathbf{I} \mid \mathbf{0}] \\ \mathbf{P}'\mathbf{H} &= [\mathbf{M} \mid \mathbf{e}'] \end{aligned}$$

The real projection matrices, $\mathbf{P}$ and $\mathbf{P}'$, have only been recovered up to an arbitrary 3D projective transformation represented algebraically by a 4×4 matrix $\mathbf{H}$, and known as a projective ambiguity.

Property 6.3.2 Projective ambiguity. *A general 3D projective transformation can be represented by a non-singular* 4×4 *matrix,* $\mathbf{H}$*, of the form*

$$\mathbf{H} = \begin{bmatrix} s\mathbf{R}_\mathbf{w} & \mathbf{t}_\mathbf{w} \\ \mathbf{0}^\top & 1 \end{bmatrix} \begin{bmatrix} \mathbf{K}^{-1} & \mathbf{0} \\ \mathbf{0}^\top & 1 \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{v}^\top & 1 \end{bmatrix}. \tag{6.15}$$

The projective ambiguity is composed of the following effects. A metric transformation resulting from the rigid body motion between the coordinate system of the first camera and the reference frame and an arbitrary scaling. This can be ignored if we align the reference coordinate system with the first camera and accept that shape can only be recovered up to an arbitrary scale, s, if the distance between the two camera centres is unknown. The second component of the ambiguity results from an 3D affine transformation due to the unknown parameters of the first camera. Finally we are left with a projective transformation which transforms points on the plane $(\mathbf{v}^\top 1)\mathbf{X} = 0$ to points on the plane at infinity and results from the ambiguity of Property 6.3.2.

The ambiguity in the projection matrices is of the form above and will result in a projective ambiguity in the recovered geometry, i.e. the 3D coordinates of visible points, $\mathbf{X}$, can only be recovered up to a 3D projective transformation, $\mathbf{H}^{-1}\mathbf{X}$. This ambiguity can only be removed with additional information derived from scene constraints or knowledge of the camera parameters, $\mathbf{K}$ and $\mathbf{K}'$. In particular the ambiguity is completely removed by using the 3D position of 5 known scene points to determine the transformation $\mathbf{H}$ or $\mathbf{H}^{-1}$. Alternatively we require the internal camera parameters of the first camera and must then find the equation of the plane at infinity represented by $\mathbf{v}$ where

$$\mathbf{M} = \mathbf{K}'\mathbf{R}\mathbf{K}^{-1} + \mathbf{e}'\mathbf{v}^\top.$$

The ambiguity is removed by using knowledge of the camera parameters to fix $\mathbf{v}$ to make the homography, $\mathbf{M}$, be that induced by the plane at infinity (Hartley 1994). The rotation matrix follows.

6.4 Frontier points and epipolar tangencies

As described in Chapter 4 an important degeneracy of the epipolar parametrization occurs when an epipolar plane (spanned by the direction of translation and the visual ray) coincides with the tangent plane to the surface. This will occur at a finite set of points on the surface where the surface normal $\mathbf{n}$ is perpendicular to the direction of translation:

$$\mathbf{c}_t \cdot \mathbf{n} = 0. \tag{6.16}$$

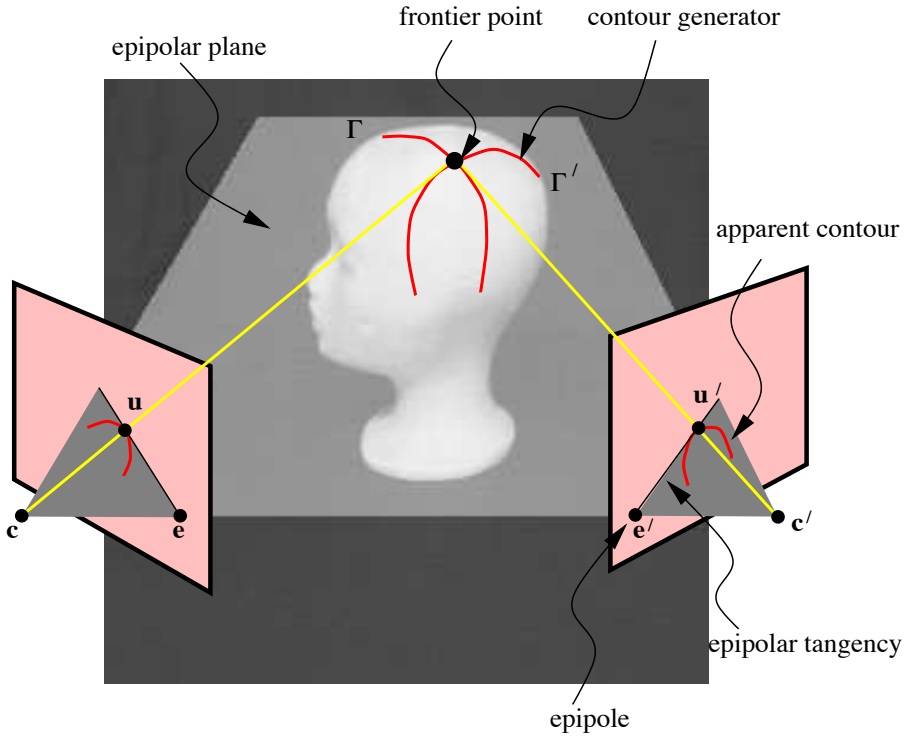


Fig. 6.2. The frontier point and epipolar tangencies in two distinct views.

The contour generator is locally stationary. In fact (See Figure 4.13) consecutive contour generators will intersect at a point on the surface called a *frontier point* (Rieger 1986 and Giblin et al. 1994)†.

For larger discrete motions (see Figure 6.2) the contour generators defined by the discrete viewpoints also intersect at points on the surface where the epipolar plane is tangent to the surface. This is easily seen if we consider the motion to be linear. $\mathbf{c}_t$ is then a *constant* vector, and the frontier point on the surface at time t satisfies the frontier condition at subsequent times. The frontier degenerates to a point on the surface. In the discrete case the frontier points are defined by the condition

$$\Delta\mathbf{c} \cdot \mathbf{n} = 0 \tag{6.17}$$

† In the image sphere model of Figure 4.13 the epipole $\mathbf{e}$ is the intersection of the epipolar great circle with the baseline $\mathbf{c}_1 - \mathbf{c}_2$, or with the velocity vector $\mathbf{c}_t$ in the limit as the two camera centres tend to coincidence. Thus the epipolar great circle passes through the epipole in exactly the same way as, in the image plane model adopted here, the epipolar line passes through the epipole.

where $\Delta\mathbf{c} = \mathbf{c}(t_2) - \mathbf{c}(t_1)$ and $\mathbf{n}$ is the surface normal at the point of intersection of the two contour generators.

Property 6.4.1 Epipolar tangency. *The frontier point projects to a point on the apparent contour which is an epipolar tangency, i.e. the epipolar line is tangent to the apparent contour.*

The property follows directly from the fact that the epipolar plane is also the tangent plane at a frontier point. Its projection in the two views, $\mathbf{u}(s)$ and $\mathbf{u}'(s)$, must be such that its tangent ($\mathbf{u}_s(s)$ and $\mathbf{u}'_s(s)$) passes through the respective epipole ($\mathbf{e}$ and $\mathbf{e}'$):

$$\det[\mathbf{e}\ \mathbf{u}(s_i)\ \mathbf{u}_s(s_i)] = 0 \tag{6.18}$$

where $\mathbf{u}_s(s_i)$ represents the image contour tangent.

The surface curvature can not be recovered at these points since the epipolar parametrization fails (§4.8). However frontier points correspond to real, fixed feature points on the surface which are visible in both views. Once detected they can be used to provide a constraint on the epipolar geometry (Figure 6.3) and hence the viewer motion. In fact they can be used in the same way as points in the recovery of the epipolar geometry via the epipolar constraint:

$$\mathbf{u}'(s_i)^\top \mathbf{F}\mathbf{u}(s_i) = 0. \tag{6.19}$$

The epipoles and epipolar tangencies in each view completely determine the epipolar geometry ((Cipolla et al. 1995) and (Aström et al. 1996)).

Property 6.4.2 Epipolar tangency constraint. *Given two images, and the epipoles* $\mathbf{e}$ *and* $\mathbf{e}'$*, the set of lines through* $\mathbf{e}$ *which are tangent to the apparent contour, and the corresponding set of epipolar tangency lines through the epipole in the other image, are related by a one-dimensional projective transformation (since they arise from the same pencil of planes through the two camera centres).*

6.5 Recovery of motion under pure translation

Under pure translation (i.e. the rotation R of §4.3 is *constant*, or that Ω in §4.3 is zero) the epipolar geometry is completely determined by the position of the epipole in a single view. The fundamental matrix has the simple

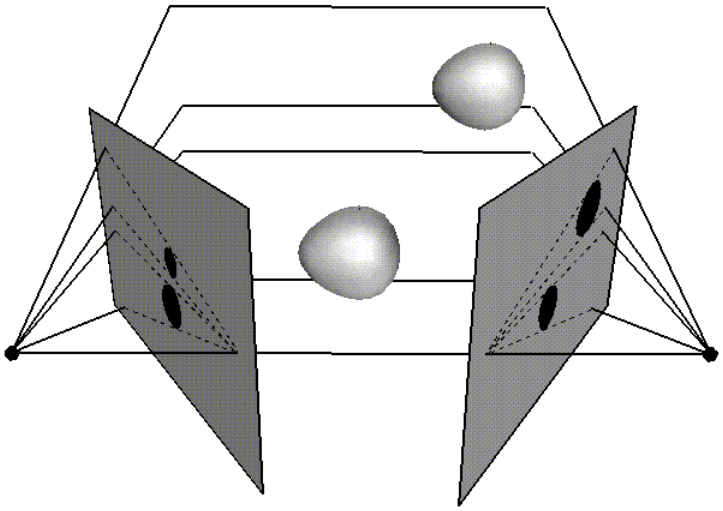


Fig. 6.3. Epipolar geometry and epipolar tangencies under arbitrary motion. The epipolar geometry is completely determined by the epipoles and 3 epipolar tangencies in each view.

skew-symmetric form:

$$\mathbf{F} = [\mathbf{e}]_\times. \tag{6.20}$$

If the camera parameters do not change the position of the epipole is the same in both views. The epipolar lines have the same directions with corresponding image points lying on the same epipolar line (i.e. are *auto-epipolar*):

$$\begin{aligned} \mathbf{e} &= \mathbf{e}' \\ \mathbf{l} &= \mathbf{l}'. \end{aligned}$$

The key fact here is that, when there is no rotation, the projection of the frontier is simply the envelope of apparent contours in *rotated* coordinates; generally it is only the envelope when the coordinates are *unrotated*. See Property 4.8.1, and (4.8) which, when $R(t)$ is constant, states the (obvious) fact that $\mathbf{p}_t = R\mathbf{q}_t$. Note that $\mathbf{p}_s = R\mathbf{q}_s$ always, since this refers to a particular time for which R will have some particular value.

For a discrete motion we superimpose the two views and find common tangents to the two consecutive apparent contours instead of the envelope. See Figure 6.4. We refer to these informally as *bitangents* (though they are only tangent to each apparent contour once). The bitangents are in

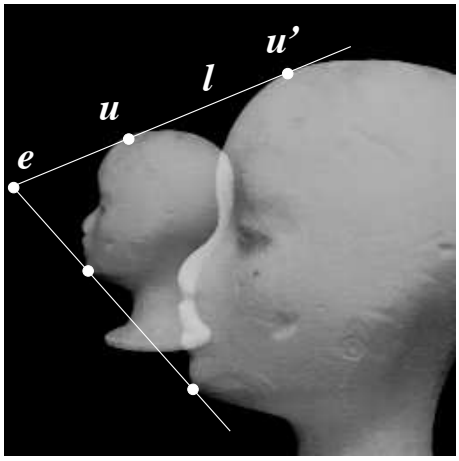


Fig. 6.4. Under pure translation the epipolar tangency point moves along the epipolar line since the position of the epipole and the direction of the epipolar lines do not change. From a minimum of two *bitangents* of the apparent contour in two views it is possible to recover the epipole, **e**.

fact epipolar tangencies and hence the projection of frontier points. The intersection of at least two distinct tangencies (epipolar lines) is therefore sufficient to determine the position of the epipole and hence the epipolar geometry. A simple procedure to find the epipole, and hence the direction of translation, is described in (Sato and Cipolla 1998).

6.6 General motion

The solution is no longer trivial in the case of arbitrary motion with rotation. There is in fact no closed form solution since the epipoles are needed to define the epipolar tangency points and these are needed to determine the epipolar geometry.

The solution proceeds as a search and optimization problem to find the position of the epipoles in both views such that the epipolar tangencies in the first view are related to the set of epipolar tangencies in the second view by a one-dimensional projective transformation or homography (see Property 6.4.2 and (6.1)).

A suitable cost function is needed. Geometric distances are used in the estimation of the fundamental matrix from point correspondences and can also be used in the case of curves. The geometric distance is computed as the sum over all tangency points of the square of the distance between the

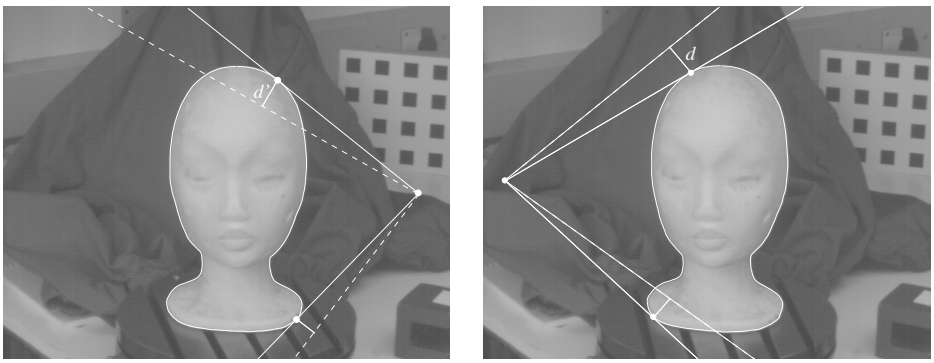


Fig. 6.5. Illustration of the cost function to be minimized in the motion estimation algorithm. From the initial guess of the epipoles the 1D projective transformation which maps epipolar lines can be determined from a minimum of 3 tangencies. Epipolar tangencies are then transfered from one image to the other. The length d is the distance from a tangency point in the first image and an epipolar line obtained by the transfer of an epipolar tangency from the second image. The distance d' is found in the same way, interchanging the roles of the images. The cost function is then the sum $\sum_i (d_i^2 + d_i'^2)$ for each matching pair i of putative epipolar tangencies.

image point and the corresponding epipolar line from the tangency point in the other view, as shown in Figure 6.5.

The key to a successful implementation is to ensure that the search space is reduced and that the optimization begins from a good starting point using approximate knowledge of the camera motion or point correspondences. A minimum of 7 epipolar tangencies of features which are visible in each image are required. The solution proceeds as follows.

Algorithm 6.6.1 Motion recovery under general motion and perspective projection.

(i) Start with an initial guess or estimate of the epipoles in both views, $\mathbf{e}$ and $\mathbf{e}'$.

(ii) Compute the epipolar tangencies, $\mathbf{u}_i(\mathbf{e})$ and $\mathbf{u}'_i(\mathbf{e}')$, in both views respectively. These are points on the apparent contours with tangents passing through the epipole as defined by (6.18).

(iii) Estimate the elements of the homography (one dimensional projective transformation) between the pencil of tangencies in both views (6.1). This can be done linearly by minimizing

$$\sum_i (h_4 \tau_i \tau'_i + h_3 \tau'_i + h_2 \tau_i + h_1)^2 \tag{6.21}$$

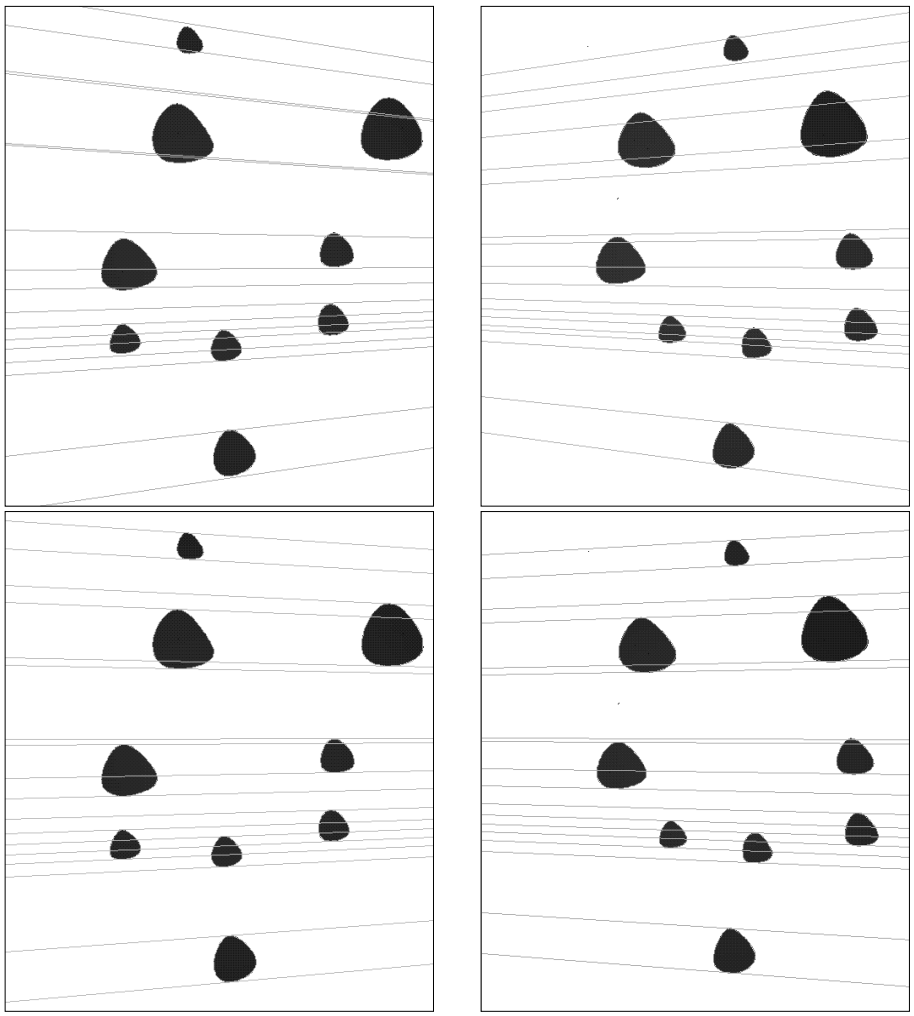


Fig. 6.6. Starting point for optimization (above). An initial guess of the position of the epipoles is used to determine epipolar tangencies in both views and the homography relating the epipolar lines. For each tangency point the corresponding epipolar line is drawn in the other view. The distances between epipolar lines and tangency points are used to search for the correct positions of the epipoles. Convergence to local minimum after 5 iterations (below). The epipolar lines are tangent to apparent contours in both views.

by least squares over all pairs ($n \geq 3$) of correspondences, τ_i and τ_i'.

(iv) The fundamental matrix is now given by the parametrization of §6.1 and the cost function, i.e. sum of squared geometric distances between tangency point and corresponding epipolar line, can be computed as

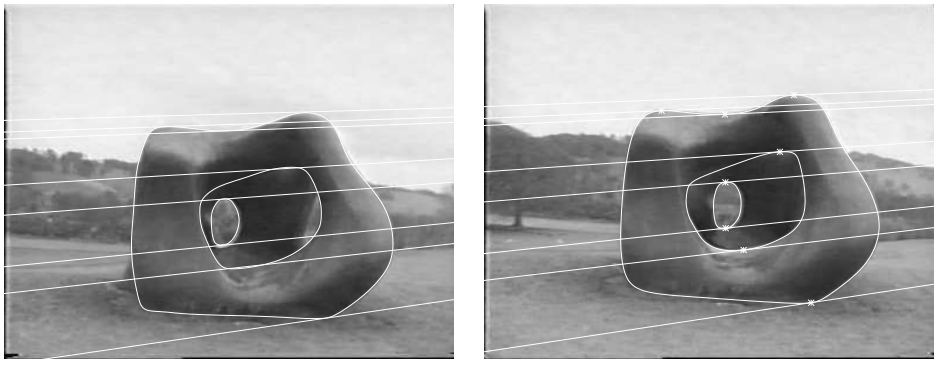


Fig. 6.7. Local minimum obtained by iterative scheme to estimate the epipolar geometry from 8 epipolar tangencies in a stereo pair of a Henry Moore sculpture.

below:

$$C = \sum_i \left(\frac{1}{(\mathbf{F}\mathbf{u}_i)_1^2 + (\mathbf{F}\mathbf{u}_i)_2^2} + \frac{1}{(\mathbf{F}^T\mathbf{u'}_i)_1^2 + (\mathbf{F}^T\mathbf{u'}_i)_2^2} \right) (\mathbf{u'}_i^T \mathbf{F}\mathbf{u}_i)^2. \tag{6.22}$$

(v) Minimize the cost function by using the Levenberg-Marquardt algorithm or the conjugate gradient method (Press et al. 1992). The search space is restricted to the four coordinates of the epipoles only. This requires the first-order partial derivatives of the cost function (6.22) with respect to the coordinates of the epipoles which can be computed analytically but are more conveniently estimated by numerical techniques.

At each iteration of the algorithm, steps (ii) to (v) are repeated, and the positions of the epipoles are iteratively refined. The search is stopped when the root-mean-square distance converges to a minimum (usually less than 0.1 pixels). It is of course not guaranteed to find a unique solution and we do not know whether there is a unique solution. The only case where we know of a uniqueness theorem is that of circular motion and parallel projection (Giblin et al. 1994).

Experimental results

A number of experiments were carried out with simulated data (with noise) and known motion (Figure 6.6). The apparent contours were automatically extracted from the sequence by fitting B-splines. 5-10 iterations each for 4 different initial guesses for the position of the epipole were sufficient to

find the correct solution to within an root-mean-square error of 0.1 pixel per tangency point.

Figure 6.7 shows an example with real data whose apparent contours are detected and automatically tracked using B-splines snakes. A solution is found very quickly which minimizes the geometric distances. The solution is however incorrect and corresponds to a local minimum. As with all structure from motion algorithms, a limited field of view and small variation in depths result in a solution which is sensitive to image localisation errors. A more stable solution can be obtained by considering a simpler projection model (see below).

Other more practical alternatives include using a planar contour (if present in the scene) to estimate the 2D projective transformation component, $\mathbf{M}$, directly. Recall that the fundamental matrix can be factored into:

$$\mathbf{F} = [\mathbf{e}']_\times \mathbf{M}.$$

We can apply this planar projective transformation, $\mathbf{M}$, to an apparent contour in the first view. The epipolar constraint can be rewritten as:

$$\mathbf{u}'(s_i)^\top [\mathbf{e}']_\times (\mathbf{M}\mathbf{u}(s_i)) = 0$$

and the epipolar geometry between points in this rectified view, $\mathbf{M}\mathbf{u}(s)$, and the second view, $\mathbf{u}'(s)$, has now been simplified. This has, in fact, reduced the general problem to the case of pure translation and is known as *projective reduction* (Aström et al. 1999). The pure translation algorithm can now be applied to find common tangents to the pairs of apparent contours (analogous to bitangents) to determine the epipole and hence the full epipolar geometry.

Finally, analysing extended image sequences, instead of only two views, avoids convergence to a local minima and results in a more accurate and better conditioned solution for the epipolar geometry and motion (Aström and Kahl 1999).

6.7 Weak perspective

When the field of view is narrow or the depth variation is small compared with the distance from the camera to the scene, the epipoles will be far from the image centre, and the epipolar lines will be approximately parallel. This viewing geometry suggests the use of a *weak perspective* camera model. The epipolar geometry under weak perspective is known as *affine epipolar geometry* (Shapiro et al. 1995), and assumes that the epipoles will be at

infinity. This reduces the degrees of freedom of the fundamental matrix which will then take the form:

$$\mathbf{F} = \begin{bmatrix} 0 & 0 & c \\ 0 & 0 & d \\ a & b & e \end{bmatrix}. \tag{6.23}$$

There are two circumstances when the *affine fundamental matrix* may be used. The first is when the weak perspective camera model can be used to describe the cameras, as described in §5.3. Another favourable situation for the use of the affine fundamental matrix is when the motion is restricted to translation orthogonal to the optical axis and rotation about the optical axis. In this case the affine fundamental matrix can be used even though the weak perspective camera model is inappropriate. It is important to notice that a rotation by a small angle around a distant axis is a good approximation for such motion.

As scale factors are not important, the affine fundamental matrix has only four degrees of freedom, and can be linearly computed from 4 point correspondences. Each epipole, being at infinity, is described by a single parameter, corresponding to its direction in the image plane. This observation suggests another parametrization for the fundamental matrix, where the directions of the epipoles are made explicit. If ϕ and ϕ' are the directions of the epipolar lines in the first and second images, the affine fundamental matrix can be expressed as

$$\mathbf{F} = \begin{bmatrix} 0 & 0 & \alpha' \sin\phi' \\ 0 & 0 & -\alpha' \cos\phi' \\ -\alpha \sin\phi & \alpha \cos\phi & \sqrt{1-\alpha^2-\alpha'^2} \end{bmatrix} \tag{6.24}$$

where the parameters α and α' are related to the distances between epipolar lines in each image. The geometric interpretation of the parameters α and α' can be seen in Figure 6.8. It is easy to show that they are proportional to the distance between epipolar lines, or, in the notation of Figure 6.8,

$$\begin{bmatrix} \alpha \\ \alpha' \end{bmatrix} = \frac{1}{\sqrt{(d_2 d_1' + d_1 d_2')^2 + (d_1' - d_2')^2 + (d_1 - d_2)^2}} \begin{bmatrix} d_1' - d_2' \\ d_2 - d_1 \end{bmatrix}. \tag{6.25}$$

In the affine case the epipolar tangencies will be parallel lines, with directions given by the corresponding epipole, and, as in the perspective, the epipolar tangencies will touch the apparent contours at corresponding points. Since the number of degrees of freedom of the affine fundamental matrix is 4, this will also be the minimum number of epipolar tangencies necessary for its computation.

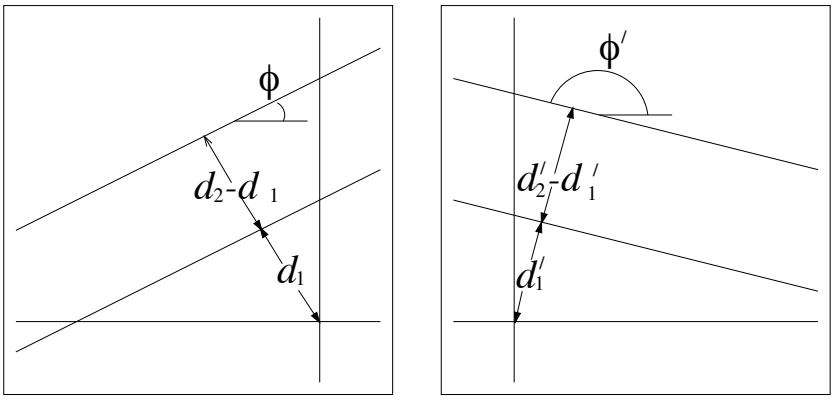


Fig. 6.8. Geometric interpretation of the parametrization of the affine fundamental matrix presented in (6.24). The orientation of the epipolar lines in each image is given by the angles ϕ and ϕ', and the parameters α and α' are proportional to the differences of distances $d_2' - d_1'$ and $d_2 - d_1$, respectively (see (6.25)).

The algorithm for computing the epipolar geometry from apparent contours under weak perspective is described as follows:

Algorithm 6.7.1 Motion recovery under weak perspective.

(i) Start with initial estimates for the directions of both epipoles, ϕ and ϕ'.

(ii) Determine tangency point correspondences $\mathbf{u}_i(\phi)$ and $\mathbf{u}_i'(\phi')$ from epipolar tangencies consistent with the directions of the epipoles.

(iii) Compute the affine fundamental matrix from the epipoles and the correspondences. This must be done by using the parametrization given in (6.24).

(iv) Minimize the sum of geometric distances from the tangency points on the contours to their corresponding epipolar lines. The search is restricted to the two directions of the epipoles, and the cost function is the same as given by (6.22).

Experimental Results

The algorithm was tested on the images shown in Figure 6.7, with the directions of the epipoles initialized at 0°. The recovered epipolar lines are shown in Figure 6.9. There is some discrepancy between the result obtained with the general motion algorithm and that obtained by the algorithm for the affine case. The epipolar geometry found by the algorithm assuming

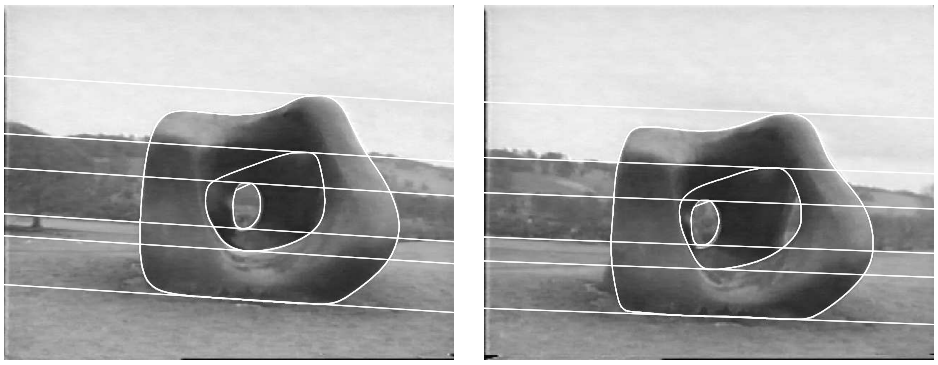


Fig. 6.9. Estimated affine epipolar geometry from the apparent contours of the Moore sequence. The result differs with the one found by the general motion algorithm shown in Figure 6.7.

weak perspective is consistent with planar motion along the ground plane and includes an epipolar line parallel with the horizon. This is qualitatively correct for the motion used in acquiring the images.

6.8 Circular motion

Circular motion is commonly used in model acquisition. An object is placed on a turntable and a sequence of snapshots are taken from a fixed camera as the object rotates about a fixed axis. This is equivalent to a stationary object and a camera undergoing motion with its centre moving on a circle and its image plane rotating rigidly with it (compare Example 4.1.1). Figure 6.10 shows a sequence obtained by a single camera under circular motion.

The estimation of epipolar geometry under circular motion is considerably simpler than more general motions. It is possible to exploit features which remain fixed in the image over the complete sequence. In fact the epipolar geometry of two views is completely specified with 6 parameters (compared with 7 in the general motion case). The epipolar geometry for a sequence of n images can be parametrized with only $5+n$ parameters: each additional view adding only one additional degree of freedom and with 5 of the parameters corresponding to features which remain fixed over the whole image sequence, irrespective of the viewpoints (Fitzgibbon et al. 1998).

Consider circular motion with the camera internal parameters remaining fixed during the object rotation. The following relationships between corresponding features lead to a simple parametrization of the epipolar geometry and fundamental matrices (Figures 6.10 and 6.11):

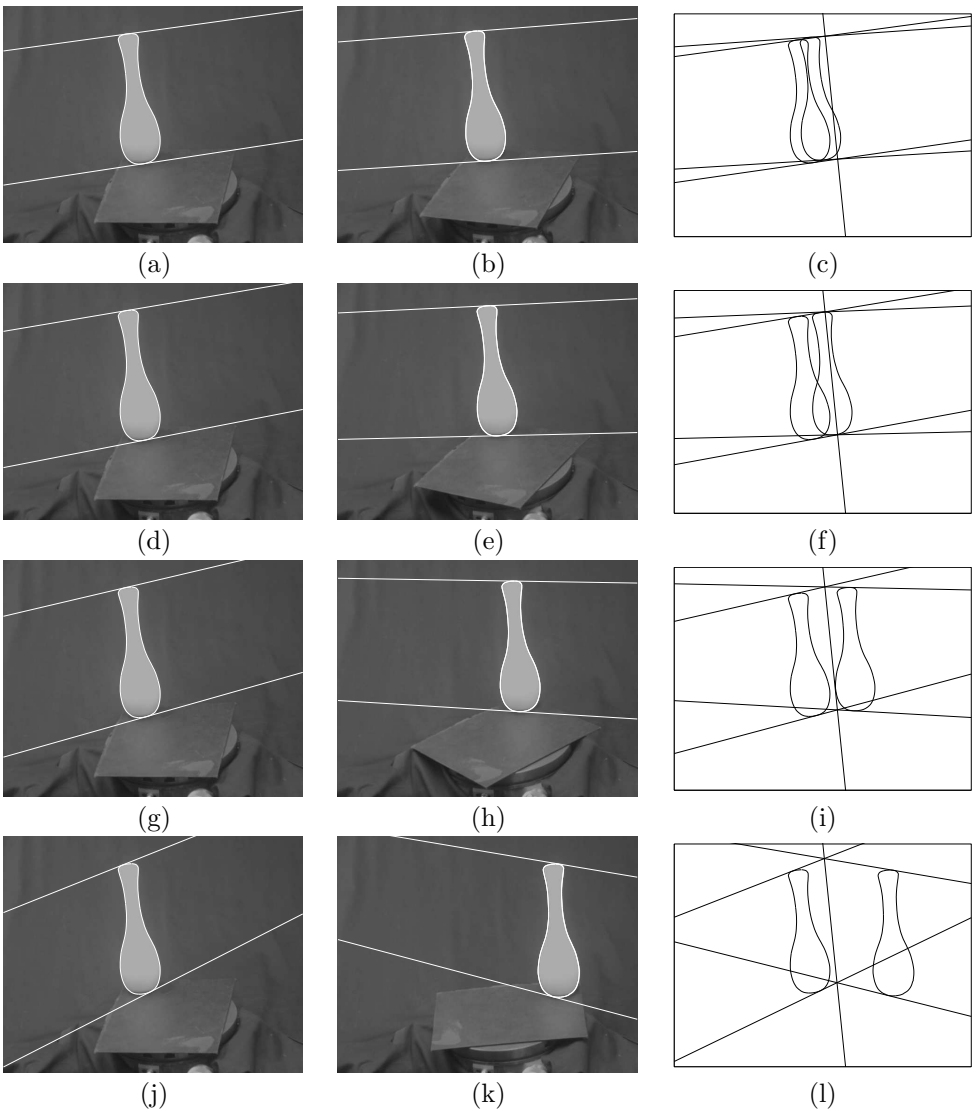


Fig. 6.10. Five images from a single camera and circular motion after a rotation of $10°, 20°, 40°$ and $80°$ are shown in (b), (e), (h) and (k). The epipolar geometry between pairs of images is shown. Epipolar tangencies are found that intersect at a common axis; lie on a common horizon (above and not shown) and which are projectively symmetric about this axis shown superimposed onto a single image in (c), (f), (i) and (l).

(i) There is no relative motion between the axis of rotation and the camera. The projection of the axis of rotation in the image, $\mathbf{l}_s$, is

therefore a fixed image line. Points on the projection of the axis, $\mathbf{l}_s \cdot \mathbf{u} = 0$, have correspondences with the same image position:

$$\mathbf{u} = \mathbf{u}'.$$

(ii) Points on the projection of the axis of rotation must therefore lie on the corresponding epipolar line drawn in the same image (i.e. are auto-epipolar). If we superimpose the two pencils of epipolar lines onto a common image, corresponding epipolar lines must intersect on the projection of the axis.

(iii) The image of the plane of motion containing the camera centres, $\mathbf{l}_h$, remains fixed in the image sequence. We refer to it as the horizon. All image points on the horizon, $\mathbf{l}_h \cdot \mathbf{u} = 0$, have corresponding image points which lie on the same epipolar line, i.e. are also auto-epipolar:

$$\mathbf{l} = \mathbf{l}'.$$

The epipoles are constrained to lie on the horizon. In homogeneous coordinates a line (the horizon) is defined by two points (the epipoles)

$$\mathbf{l}_h = \mathbf{e} \wedge \mathbf{e}'.$$

(iv) The rays corresponding to pairs of epipoles superimposed onto a single image are bisected by the plane containing the camera centre and the axis of rotation (i.e. exhibit bilateral symmetry).

(v) There is a fixed point on the horizon, $\mathbf{u}_s$, which is the same for all views and depends on the orientation of the camera relative to the axis of rotation. As a consequence, the positions of the epipoles on the horizon are constrained. The cross-ratio of the two epipoles, the intersection of the axis of rotation and the horizon and the fixed point on the horizon, is fixed and must correspond to bilateral symmetry in space (explained below).

All points $\mathbf{u}$ which lie on the projection of the axis or the horizon are auto-epipolar and must therefore satisfy the quadratic form:

$$\mathbf{u}^\top \mathbf{F} \mathbf{u} = 0 \tag{6.26}$$

when $\mathbf{l}_h^\top \mathbf{u} = 0$ or $\mathbf{l}_s^\top \mathbf{u} = 0$. The condition is satisfied by any skew-symmetric matrix. The symmetric part of the fundamental matrix (up to an arbitrary scale factor) is constrained to be of the form

$$\mathbf{F} + \mathbf{F}^\top = \mathbf{l}_s \mathbf{l}_h^\top + \mathbf{l}_h \mathbf{l}_s^\top.$$

Property 6.8.1 Epipolar geometry under circular motion. *Under circular motion the fundamental matrices, $\mathbf{F}_{ij}$, relating correspondences in two views in the sequence, i and j, have a special form which can be parametrized into an anti-symmetric component which depends on a fixed vanishing point on the horizon, $\mathbf{u}_s$, and a symmetric component which depends only on the image of the axis of rotation, $\mathbf{l}_s$,and the horizon, $\mathbf{l}_h$:*

$$\mathbf{F}_{ij} = [\mathbf{u}_s]_\times + k_{ij}\left[\mathbf{l}_s\mathbf{l}_h^\top + \mathbf{l}_h\mathbf{l}_s^\top\right] \tag{6.27}$$

The scaling factor, k_{ij}, depends on the angle between the two views, ϕ_{ij}, and uniquely determines the position of both epipoles once the other 5 parameters are known.

A derivation of (6.27) follows directly by considering two projection matrices parametrized for circular motion (see (6.34)). The scaling factor, k_{ij} can be shown to be equal to $\tan(\phi_{ij}/2)$. The anti-symmetric component results from symmetry properties of circular motion which will be considered below. The latter can be used to derive a simpler parametrization of the fundamental matrix.

Bilateral symmetry under perspective projection

Symmetry properties play a useful role in the recovery of epipolar geometry under circular motion. These are briefly derived below. Consider the projection of pairs of points in space which are bilaterally symmetric about the plane containing the camera centre and the axis of rotation. If the optical axis intersects the axis of rotation, the projection of point correspondences will also display bilateral symmetry about the projection of the axis of rotation, $\mathbf{l}_s$.

The symmetry transformation, $\mathbf{T}$, between corresponding points in the image, $\mathbf{u}$ and $\mathbf{u}'$:

$$\mathbf{u}' = \mathbf{T}\mathbf{u}$$

has a very special structure since it must satisfy $\mathbf{T}^2 = \mathbf{I}$ and must map points on the axis to themselves. In fact the transformation has eigenvalues $\{-1, 1, 1\}$ with the eigenvectors with the same eigenvalue defining the image of the axis of symmetry and the other corresponding to the vanishing point of the lines of symmetry (i.e. in a direction perpendicular to the axis and at infinity). See Figure 6.12(a).

The bilateral symmetry is projectively distorted if the optical axis is rotated away from the axis of rotation. Note that the components of rotation about the optical axis and perpendicular to the axis of rotation leave this

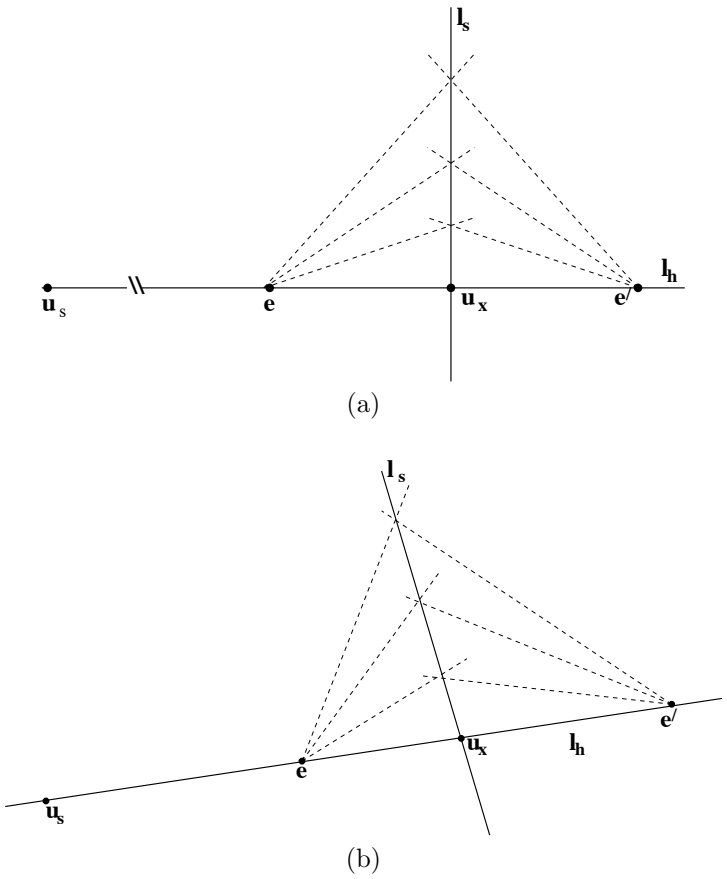


Fig. 6.11. Epipolar geometry under circular motion. The epipolar geometry of two views is completely determined by the projection of the axis of rotation, $\mathbf{l}_s$, and the position of the two epipoles. Corresponding epipolar lines intersect on the axis. Only two epipolar tangencies are required to fix the one-dimensional projective transformation between pencils of epipolar lines. The epipoles are also bilaterally symmetric about the axis when the optical axis intersects the axis of rotation (a). Rotation of the optical axis away from the axis of rotation introduces a projective distortion with lines of symmetry meeting at a vanishing point $\mathbf{u}_s$, (b). This point is a infinity in (a).

symmetry unchanged. Only the component of rotation about an axis parallel to the rotation axis will result in projective distortion. See Figure 6.12.

Under rotation about the optical centre, image points are mapped by a 2D projective transformation given by (6.12). The new transformation relating points in the image which are bilaterally symmetric in space is given by:

$$\mathbf{T}_s = \mathbf{M}_\infty \mathbf{T} \mathbf{M}_\infty^{-1}$$

where $\mathbf{M}_\infty$ is the 2D projective transformation which maps image points before and after the rotation. The eigenvalues are preserved under this transformation, leading to the following property (Springer 1964).

Property 6.8.2 Planar projective symmetry transformation. *Under perspective projection, pairs of points in space which are bilaterally symmetric about the plane containing the camera centre and the axis of rotation, are related in the image by a projective symmetry transformation,* $\mathbf{T}_s$*, which can be represented by a* 3×3 *matrix with 4 degrees of freedom and eigenvalues* $\{-1, 1, 1\}$ *given by:*

$$\mathbf{T}_s = \mathbf{I} - 2\frac{\mathbf{u}_s \mathbf{l}_s^\top}{\mathbf{l}_s^\top \mathbf{u}_s}. \tag{6.28}$$

The transformation is known as a *planar harmonic homology* and is fixed by the image of the axis of rotation, $\mathbf{l}_s$, and the vanishing point, $\mathbf{u}_s$. The latter corresponds to the image of parallel lines of symmetry which map corresponding points and is known as the centre of the homology. $\mathbf{u}_s$ is at infinity in the special case of bilateral symmetry in the image.

The transformation must satisfy $\mathbf{T}_s^2 = \mathbf{I}$ and maps the point $\mathbf{u}_s$ and points on the axis, $\mathbf{l}_s^\top \mathbf{u} = 0$, to themselves:

$$\mathbf{T}_s \mathbf{u}_s = -\mathbf{u}_s \tag{6.29}$$

$$\mathbf{T}_s \mathbf{u} = \mathbf{u}. \tag{6.30}$$

The transformation can therefore be parametrized by (6.28).

Fundamental matrix under circular motion

The camera centres, epipoles and pencils of epipolar lines are also symmetric about the axis of rotation. See Figure 6.11. The epipoles are therefore mapped by the projective symmetry transformation:

$$\mathbf{e}' = \mathbf{T}_s \mathbf{e} \tag{6.31}$$

while corresponding epipolar lines are related by

$$\mathbf{l}' = \mathbf{T}_s^{-\top} \mathbf{l} = \mathbf{T}_s^\top \mathbf{l}. \tag{6.32}$$

The projective transformation, $\mathbf{T}_s$, is in fact a special case of the 2D projective transformation (homography) between views induced by an arbitrary plane (i.e. $\mathbf{M}$ in Property 6.3.1) and can be used to define a minimal parametrization of the fundamental matrix.

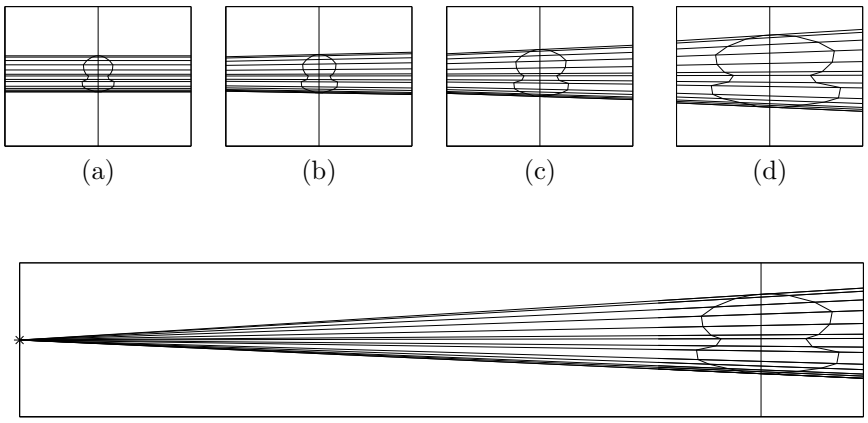


Fig. 6.12. Under perspective projection objects with a plane of symmetry which passes through the camera centre exhibit a projective symmetry. The axis and lines of symmetry are shown for different orientations of the camera relative to the object's rotation axis. If the optical axis intersects the axis of rotation the symmetry is bilateral (a). The amount of projective distortion increases with camera rotation about an axis parallel to the rotation axis of the object. The projective transformation for rotations of 20°, 30° and 40° is shown in (b), (c) and (d) respectively. The transformation is completely determined by the axis, $\mathbf{l}_s$ and the vanishing point, $\mathbf{u}_s$, shown in (e).

Property 6.8.3 Parametrization of fundamental matrix under circular motion. *The fundamental matrix under circular motion can be parametrized by a single epipole and the planar projective symmetry transformation:*

$$\mathbf{F} = [\mathbf{e}']_\times \mathbf{T}_s \tag{6.33}$$

where

$$\mathbf{T}_s = \mathbf{I} - 2\frac{\mathbf{u}_s \mathbf{l}_s^\top}{\mathbf{l}_s^\top \mathbf{u}_s}.$$

The epipolar geometry of two views is completely determined by a single epipole, the fixed point $\mathbf{u}_s$ and the projection of the axis of rotation, $\mathbf{l}_s$. Alternatively the two epipoles, $\mathbf{e}$ and $\mathbf{e}'$, and the projection of the axis of rotation, $\mathbf{l}_s$ can be used. It has only 6 degrees of freedom.

Computation of epipolar geometry

Consider a pair of views obtained under circular motion. The key observation with apparent contours is that every epipolar tangency must be transferred to an epipolar line which remains tangent to the apparent contour in the other image. The two epipolar lines must also be related by the projective symmetry transformation and hence intersect at the projection of the axis of rotation. See Figures 6.10 and 6.11. This provides a very simple constraint for finding the epipolar tangencies in both views.

If we assume that the positions of the epipoles are known, two epipolar tangencies are sufficient to fix $\mathbf{l}_s$ and thus the epipolar geometry. The position of the epipoles is controlled by four degrees of freedom (the coordinates of the epipoles), and at least four more epipolar tangencies are required to determine the epipolar geometry of the pair of cameras. The general motion algorithm can be applied to the two images, exploiting the simplified parametrization for the fundamental matrix.

With more images better results can be obtained by exploiting the fact that the horizon, $\mathbf{l}_h$; the projection of the axis of rotation, $\mathbf{l}_s$; and the vanishing point, $\mathbf{u}_s$, remain fixed in all of the images. Each view, in fact, adds only 1 additional unknown (parameter k_{ij} in (6.27)) corresponding to the position of one of the epipoles on the horizon. Two epipolar tangencies in at least four images is sufficient to completely determine the epipolar geometry and fix the angles of rotation between the views. Results on real images are as shown in Figure 6.10.

6.9 Envelope of apparent contours under circular motion

Simpler methods exist when viewing an object that undergoes a full rotation around a fixed axis. The object sweeps out a surface of revolution. In the image the envelope of the apparent contours is in fact the image of the envelope of surfaces (a surface of revolution in this case) as described in §4.12) and (Giblin et al. 1994). If viewed by a camera pointing towards the axis of rotation, the two contour generators are a bilaterally symmetric pair with the plane of symmetry passing through the camera centre. In the image the envelope will also be symmetric about the image of the axis of rotation. See Figure 6.13.

The symmetry in the image is projectively distorted if the optical axis is rotated away from the axis of rotation (see Figure 6.12) with the two sides of the envelope being mapped by the projective symmetry transformation $\mathbf{T}_s$. The envelope can thus be used to recover the transformation $\mathbf{T}_s$. Its eigen-

Fig. 6.13. Envelope of apparent contours under circular motion. If the optical axis intersects the axis of rotation the two sides of the envelope are bilaterally symmetric. Rotation of the optical axis away from the axis of rotation introduces a projective distortion.

vectors will determine the image of the axis of rotation, $\mathbf{l}_s$, and a vanishing point on the horizon, $\mathbf{u}_s$.

Hence the epipolar geometry is determined by only 2 parameters once this transformation is known, corresponding the position of one of the epipoles. Alternatively if the horizon is also known, then only one parameter is required to fix the epipolar geometry. This parameter corresponds to the angle between the two views or the position of one of the epipoles on the horizon.

Again the key observation with apparent contours is that every epipolar tangency must be transferred to an epipolar line which remains tangent to the apparent contour in the other image. See Figure 6.15. By exploiting symmetry properties, this provides a very simple constraint for finding the epipolar tangencies in both views. In fact only a one-parameter search is required to fully compute the epipolar geometry once the transformation $\mathbf{T}_s$ has been determined as described in the algorithm below.

Implementation

The implementation proceeds in two stages. First the projective symmetry transformation, $\mathbf{T}_s$, is estimated from the envelope of the apparent contours. Many methods exist to do this, for example, by using invariant descriptions of planar curves (Sato and Cipolla 1998). Here we choose to find the trans-

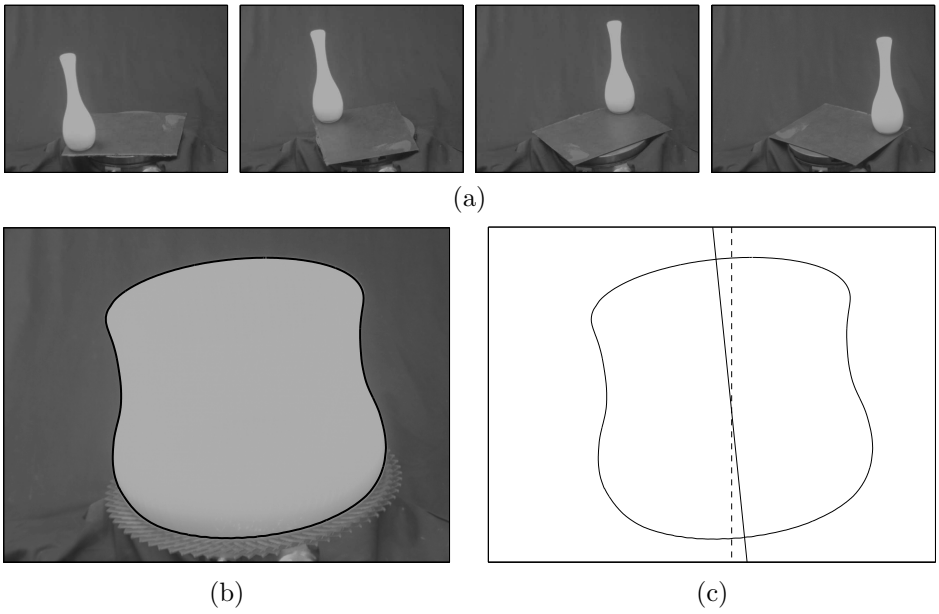


Fig. 6.14. (a) Image 1, 8, 15 and 22 in a sequence of 36 images of a rotating vase. (b) Envelope of apparent contours produced by overlapping all images in the sequence. (c) Initial guess (dashed line) and final estimation (solid line) of the image of the rotation axis.

formation by search (Cham and Cipolla 1996). The second stage involves a one-parameter search for the epipolar tangencies.

Algorithm 6.9.1 Motion recovery under circular motion.

Estimate the projective symmetry transformation, $\mathbf{T}_s$ from the envelope of apparent contours:

(i) Extract the envelope of apparent contours, $\mathbf{E}$. This can be obtained from the family of B-spline snakes used to track the apparent contour. See Figure 6.14.

(ii) Estimate the 4 parameters of the homography $\mathbf{T}_s(\mathbf{l}_s, \mathbf{u}_s)$ by sampling the envelope at N image points, $\mathbf{u}$, and finding the transformation which minimizes the sum of the squared distances between the envelope and mapped points. Initialization is performed by assuming bilateral symmetry (i.e. optical axis pointing at axis of rotation).

Search for epipolar tangencies between pairs of images with the following algorithm:

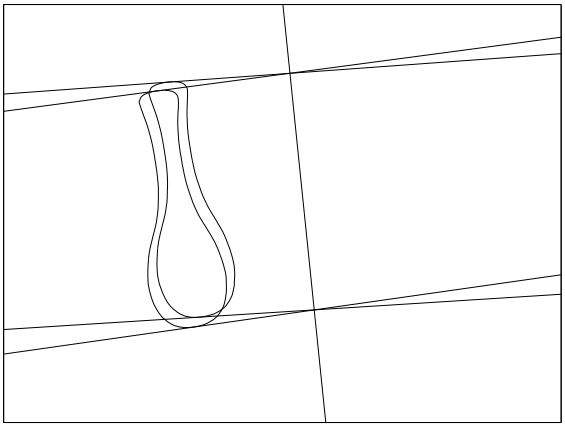


Fig. 6.15. Corresponding pair of epipolar tangents near the top and bottom of an apparent contour in two images.

(i) For each epipolar tangency, assume the orientation of the epipolar line in the first view is α such that the epipolar line is $\mathbf{l}(\alpha)$.
(ii) Transfer this tangency to the other apparent contour by computing

$$\mathbf{l}'(\alpha) = \mathbf{T}_s^{\top}\mathbf{l}(\alpha)$$

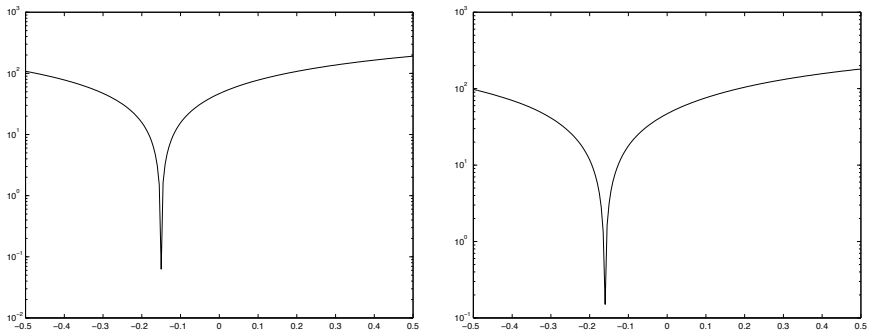
and compute the geometric distance to the apparent contour.
(iii) Update α (one-parameter optimization problem) which minimizes the geometric distance.

Figure 6.16 shows the cost function for the top and bottom epipolar tangencies of the apparent contours of Figure 6.15. A minimum of two epipolar tangencies uniquely define the epipoles and the epipolar geometry.

An alternative to performing the one-parameter search is to map the apparent contour of one view into the other view by using the projective symmetry transformation. The two epipoles have been mapped to a single point and common tangents to the pair of apparent contours (referred to as bitangents for convenience) define the epipoles uniquely. This is exactly the same method exploited under pure translation and projective reduction. Again the epipolar geometry between the apparent contour in the rectified image, $\mathbf{T}_s\mathbf{u}(s_i)$, and the second image has been reduced to the case of pure translation:

$$\mathbf{u}'(s_i)^{\top}[\mathbf{e}']_{\times}(\mathbf{T}_s\mathbf{u}(s_i)) = 0.$$

See Figure 6.17(a) and (b). Note that this computation is ill-conditioned when the apparent contours display symmetry about the projection of the



(a) (b)

Fig. 6.16. Plot of the cost function for the pair of apparent contours shown in Figure 6.15. The cost function related to the orientation (in radians) of the epipolar lines for an epipolar tangency near the top of the apparent contour (a) and bottom (b).

axis. This is well-known to pyschophysicists who have reported a similar result in experiments on the human perception of shape from profiles (Pollick 1994).

Recovery of projection matrices

After computing the fundamental matrices, the projection matrices can be recovered for each viewpoint. If the camera internal parameters are known this is straightforward and follows the decomposition of §6.2. See Figure 6.18.

In the following we provide an alternative. Without loss of generality, we can fix the reference coordinate system to be centred at the axis of rotation, with the Z axis aligned with the axis of rotation. The X axis can be aligned with the ray defined by the intersection of the horizon and the projection of the axis of rotation. The projection matrices are then given by (Fitzgibbon et al. 1998)

$$\mathbf{P}(\phi_i) = \mathbf{K}\mathbf{R}_\mathbf{0} \begin{bmatrix} \cos\phi_i & \sin\phi_i & 0 & -\rho \\ -\sin\phi_i & \cos\phi_i & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{6.34}$$

and defined up to a 2D projective transformation $\mathbf{K}\mathbf{R}_\mathbf{0}$ where $\mathbf{R}_\mathbf{0}$ is the orientation of the camera relative to the reference coordinate system attached to the turntable. This transformation is completely fixed by the 3 vanishing points in the image plane corresponding to the directions of the reference

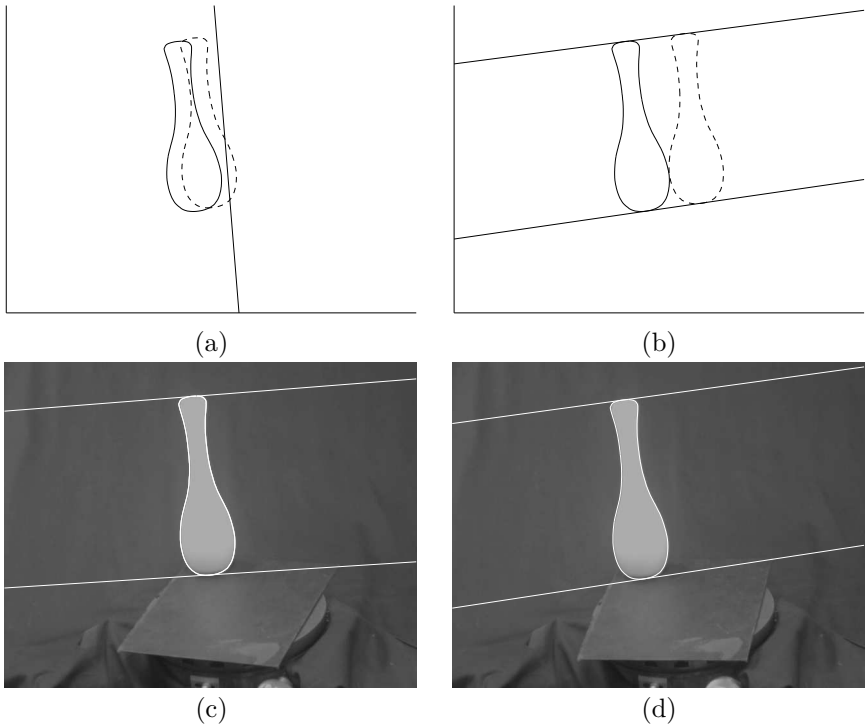


Fig. 6.17. Exploiting symmetry. The one-dimensional parameter search to find the tangencies can by avoided by exploiting the projective symmetry transformation. The original contours with symmetry axis (a). Common tangents (bitangents) to the apparent contour and transformed apparent contour from other view uniquely determine the epipolar geometry (b). The epipolar geometry for the pair of images is shown in (c) and (d).

coordinate system,

$$\mathbf{K}\mathbf{R}_0 = [\mathbf{u}_X \ \mathbf{u}_Y \ \mathbf{u}_Z] \, .$$

Two of the vanishing points have already been determined directly from image measurements alone. The intersection of the horizon and the projection of the axis of rotation rotation, $\mathbf{u}_X = \mathbf{l}_h \wedge \mathbf{l}_s$, is in fact the projection of the X axis. The vanishing point corresponding to the Y axis is the fixed point, $\mathbf{u}_Y = \mathbf{u}_s$, i.e. the vanishing point of the lines of symmetry in (6.28). Note that both of these can be obtained without knowledge of internal camera parameters.

The last component of orientation requires the position of the vanishing point of lines parallel to the axis of rotation. This must lie on the projection of the axis of rotation. The direction in space is of course perpendicular

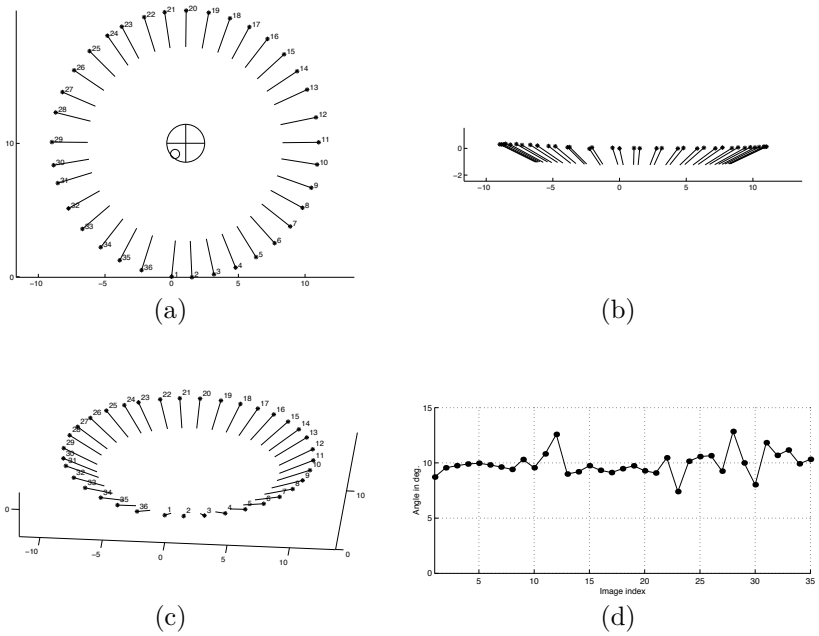


Fig. 6.18. (a-c) Final configuration of the estimated motion of the cameras. (d) Estimated angles of rotation.

to the others and can be computed up to an arbitrary sign from the other vanishing points if the camera calibration parameters are known. For an uncalibrated camera with unknown internal parameters an ambiguity remains in the orientation of the camera and leads to a 3D projective ambiguity in the reconstructed surface (Fitzgibbon et al. 1998).

The angles of rotation, ϕ_i, can be recovered from the position of the epipole in each image. Projecting the camera centre of the first viewpoint ($\phi = 0$), $\mathbf{c} = (\rho, 0, 0)$, into view i gives the epipole position:

$$\mathbf{e}'_i = \mathbf{K}\mathbf{R}_0 \begin{bmatrix} \cos\phi_i - 1 \\ \sin\phi_i \\ 0 \end{bmatrix}$$

or alternatively (up to an arbitrary scale):

$$\mathbf{e}'_i = \mathbf{u}_s - \tan(\frac{\phi_i}{2})\mathbf{u}_X. \tag{6.35}$$

As noted before the epipoles are constrained to lie on the horizon. The arbitrary scale and angles can be recovered from a minimum of three views, even when the cameras are uncalibrated.

The epipolar geometry and the recovered projection matrices under circular motion have proven to be extremely reliable. In combination with the methods of reconstruction presented in Chapter 5, it is now possible to acquire a three-dimensional model of an arbitrary object placed on a turntable in front of a fixed camera. Both the estimation of camera motion and the reconstruction can be made to be fully automatic. Extensions to more general motions are in progress.

Afterword

We have tried to show in this book that the visual geometry of curves and surfaces is a rich field for mathematical and experimental investigation. (It has kept us busy for the last decade or so.) On the mathematical side, we have developed enough of the geometry of surfaces to show how apparent contours can, in principle, be used to reconstruct the surfaces of objects in the environment, using a calibrated camera with known motion. We hope that, in fact, the discussion of surface geometry will serve as an introduction and reference for anyone wishing to apply this powerful theory to problems in computer vision. On the experimental side, we have shown how the theory can be put into practice, even in the very difficult case of uncalibrated cameras with restricted but not fully known motion. It is, to our minds, very surprising that any useful information can be obtained from apparent contours in this situation. This brings us back to mathematical questions of uniqueness of reconstruction: could the data be consistent with two different restricted motions of a camera viewing different objects? The method we presented is iterative, and it is possible that iteration does not lead to a unique solution. Even harder is the question of whether motion and object can be recovered from apparent contours when the camera motion is not restricted, and, correspondingly, whether the answer is unique. These are problems for future investigation by ourselves or others. We hope that we have shown that there are highly practical problems to be solved here, and also that there is elegant and powerful mathematics at hand with which to attack them.

Bibliography

Aström, K., and Kahl, F. (1999). Motion estimation in image sequences using the deformation of apparent contours. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 21:(2), 114–127.

Aström, K., Cipolla, R. and Giblin, P.J. (1996,1999). Generalised epipolar constraints. *Int. Journal of Computer Vision*, 33(1):51–72. Also in *Proc. 4th European Conf. on Computer Vision*, (Cambridge). Edited by B.F. Buxton and R. Cipolla, pages 97–108, Lecture Notes in Computer Science 1065, Springer–Verlag.

Banchoff, T., Gaffney, T. and McCrory, C. (1982). *Cusps of Gauss Mappings*, Pitman Research Notes in Mathematics, volume 55.

Berger, M. and Gostiaux, B. (1988). *Differential Geometry: Manifolds, Curves and Surfaces*, Springer-Verlag Graduate Texts in Mathematics No. 115.

Bartels, R., Beatty, J., and Barsky, B. (1987). *An introduction to splines for use in computer graphics and geometric modeling*, Morgan Kaufmann.

Blake, A. and Cipolla, R. (1990). Robust estimation of surface curvature from deformation of apparent contours. In *Proc. 1st European Conference on Computer Vision*, (Antibes). Edited by O. Faugeras, pages 465–474, Lecture Notes in Computer Science 427, Springer–Verlag.

Blake, A. and Isard, M. (1998). *Active Contours: the application of techniques from graphics, vision, control theory and statistics to visual tracking of shapes in motion*, Springer-Verlag, London.

Bruce, J.W. and Fidal, D. (1989). On binary differential equations and umbilics. *Proc. Royal Soc. Edinburgh*, A111:147–168.

Bruce, J.W. and Giblin, P.J. (1984,1992). *Curves and Singularities*, Cambridge University Press, 2nd edition. (1st edition 1984.)

Bruce, J.W. and Tari, F. (1995). On binary differential equations, *Nonlinearity* 8:255–271.

Boyer, E. and Berger, M.O. (1997). 3D Surface reconstruction using occluding contours. *Int. Journal of Computer Vision*, 22(3):219–233.

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

Cham, T.J. and Cipolla, R. (1996). Geometric saliency of curve correspondences and grouping of symmetric contours. In *Proc. 4th European Conf. on Computer Vision*, (Cambridge). Edited by B.F. Buxton and R. Cipolla, Lecture Notes in Computer Science 1064, Springer–Verlag, pages 385–398,

Cham, T.J. and Cipolla, R. (1999). Automated B-spline curve representation incorporating MDL and error-minimizing control point insertion strategies. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 21:(1):49–53.

Cipolla, R. (1991, 1995). *Active Visual Inference of Surface Shape*. D. Phil. thesis, University of Oxford. Also as Lecture Notes in Computer Science 1016, Springer-Verlag, Heidelberg, 1995.

Cipolla, R., Aström, K. and Giblin, P.J. (1995). Motion from the frontier of curved surfaces. In *Proc. IEEE 5th Int. Conf. on Computer Vision*, (Cambridge, Mass.), pages 269–275.

Cipolla, R. and Blake, A. (1990). The dynamic analysis of apparent contours. In *Proc. IEEE 3rd Int. Conf. on Computer Vision*, (Osaka), pages 616–623.

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

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

Cipolla, R. and Zisserman, A. (1992). Qualitative surface shape from deformation of image curves. *Int. Journal of Computer Vision*, 8(1):53–69.

Faugeras, O.D. (1992). What can be seen in three dimensions with an uncalibrated stereo rig? In *Proc. 2nd European Conference on Computer Vision*, (Santa Margherita Ligure). Edited by G. Sandini, pages 563–578, Lecture Notes in Computer Science 588, Springer–Verlag.

Faugeras, O.D. (1993). *Three-Dimensional Computer Vision: A Geometric Viewpoint*, MIT Press, Cambridge (Mass.).

Faugeras, O.D. (1995). Stratification of three-dimensional vision: projective, affine and metric representation. *J. Opt. Soc. America* A12:465–485.

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

Fitzgibbon, A., Cross, G. and Zisserman, A. (1998). Automatic 3D model construction for turn-table sequences. In *3D structure from multiple images of large-scale environments*, pages 155–170, Lecture Notes in Computer Science 1506, Springer–Verlag.

Fletcher, G. (1996). *Geometrical Problems in Computer Vision*, Ph.D. thesis, University of Liverpool.

Fletcher, G. and Giblin, P.J. (1996). Class based reconstruction techniques using singular apparent contours. In *Proc. 4th European Conf. on Computer Vision*, (Cambridge). Edited by B.F. Buxton and R. Cipolla, pages 107–116, Lecture Notes in Computer Science 1064, Springer–Verlag.

Giblin, P.J., Pollick, F.E. and Rycroft, J.E. (1994). Recovery of an unknown axis of rotation from the profiles of a rotating surface. *J. Opt. Soc. America* A11(7):1976–1984.

Giblin, P.J., Pollick, F.E. and Rycroft, J.E. (1994). Moving surfaces. *Design and Applications of Curves and Surfaces: Mathematics of Surfaces V*, Edited by R.B. Fisher, pages 433–453, Clarendon Press, Oxford,

Giblin, P.J. and Soares, M.G. (1988). On the geometry of a surface and its singular profiles. *Image and Vision Computing* 6:225–234.

Giblin, P.J. and Weiss, R. (1987). Reconstruction of surfaces from profiles. In *Proc. IEEE 1st Int. Conf. on Computer Vision* (London), pages 136–144.

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

Gutierrez, C. and Sotomayor, J. (1998). Lines of curvature, umbilic points and

Carathéodory Conjecture, *Resenhas IME-Univ. São Paulo, Brazil* 3:291–322.
Hallinan, P., Gordon, G., Yuille, A., Giblin, P. and Mumford, D. (1999). *Two and three dimensional patterns of the face*, A.K.Peters, Natick, Massachusetts.
Hartley, R.I. (1992). Estimation of relative camera positions for uncalibrated cameras. In *Proc. 2nd European Conference on Computer Vision*, (Santa Margherita Ligure). Edited by G. Sandini, pages 579–587, Lecture Notes in Computer Science 588, Springer–Verlag.
Hartley, R.I., Gupta, R. and Chang, T. (1992). Stereo from uncalibrated cameras. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pages 761–764.
Hartley, R.I. (1994). Euclidean reconstruction from uncalibrated views. In *Applications of invariance in computer vision*. Edited by J. Mundy, A. Zisserman and D.A. Forsyth, pages 237–256, Lecture Notes in Computer Science 825, Springer-Verlag.
Hartley, R.I. (1997). Lines and points in three views and the trifocal tensor. *Int. Journal of Computer Vision*, 22(2):125–140.
Hartley, R.I. (1998). Minimizing algebraic error. *New Geometric Techniques in Computer Vision*. Phil. Trans. R. Soc. Lond., A356(1740):1175–1189.
Hartley, R.I. and Zisserman, A. *Multiple View Geometry in Computer Vision*. (In press), Cambridge University Press.
Horn, B.K.P. (1987). Closed-form solution of absolute orientation using unit quaternions. *J. Opt. Soc. America*, A4(4):629–642.
Huang, T.S. and Netravali, A.N. (1994). Motion and structure from feature correspondences: a review. In *Proc. IEEE*, 82(2):252–268.
Kass, M., Witkin, A. and Terzopoulos, D. (1988). Snakes: active contour models. *Int. Journal of Computer Vision*, 1(4):321–331.
Koenderink, J.J. (1984). What does the occluding contour tell us about solid shape. *Perception*, 13:321–330.
Koenderink, J.J. (1990). *Solid Shape*, MIT Press, Cambridge, Massachusetts.
Koenderink, J.J. and Van Doorn, A.J. (1976). The singularities of the visual mapping. *Biological Cybernetics*, 24:51–59.
Koenderink, J.J. and Van Doorn, A.J. (1982). The shape of smooth objects and the way contours end. *Perception*, 11:129–137.
Koenderink, J.J. and Van Doorn, A.J. (1991). Affine structure from motion. *J. Opt. Soc. America*, 8(2):377–385.
Longuet-Higgins, H.C. (1981). A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135.
Luong, Q.-T. and Faugeras, O.D. (1996). The fundamental matrix: theory, algorithms, and stability analysis. *Int. Journal of Computer Vision*, 17(1):43–76.
Luong, Q.-T. and Vieville, T. (1996). Canonic representations for the geometries of multiple projective views. *Computer Vision and Image Understanding*, 64(2):193–229.
Maybank, S.J. (1991). The projective geometry of ambiguous surfaces. *Phil. Trans. Royal Society, London*, 332(1623):1–47.
Maybank, S.J. and Faugeras, O.D. (1992). A theory of self-calibration of a moving camera. *Int. Journal of Computer Vision*, 8(2):123–151.
Mundy, J. and Zisserman, A. (1992). *Geometric invariance in computer vision*, MIT Press, Cambridge (Mass.).
O'Neill, B. (1966, 1997). *Elementary Differential Geometry*, Academic Press, New

York.

Pollick, F.E. (1994). Perceiving shape from profiles. In *Perception and Pyschophysics*, 55(2):152–161.

Porrill, J. and Pollard, S. (1991). Curve matching and stereo calibration, *Image and Vision Computing*, 9:45–50.

Porteous, I.R. (1994). *Geometric Differentiation*, Cambridge University Press, Cambridge.

Press, W.H., Teukolsky, S.A., Vetterling, W.T. and Flannery, B.P. (1992). *Numerical Recipes in C: The Art of Scientific Computing*, 2nd edition, Cambridge University Press, Cambridge.

Rieger, J.H. (1986). Three-dimensional motion from fixed points of a deforming profile curve. *Optics Letters*, 11(3):123–125.

Rycroft, J.E. (1992). *A Geometrical Investigation into the Projections of Surfaces and Space Curves*, Ph.D. thesis, University of Liverpool.

Roberts, L.G. (1965). Machine perception of three-dimensional solids. In *Optical and electrooptical information processing*. Edited by J. Tippett, D. Berkowitz, L. Clapp, C. Koester and A. Vanderburgh, pages 159-197.

Sato, J. and Cipolla, R. (1998). Quasi-invariant parameterisations and matching of curves in images. *Int. Journal of Computer Vision*, 28(2):117–136.

Sato, J. and Cipolla, R. (1998). Affine reconstruction of curved surfaces from uncalibrated views of apparent contours. In *Proc. IEEE 6th Int. Conf. on Computer Vision*, (Bombay), pages 715–720.

Semple, J. and Kneebone, G. (1979). *Algebraic Projective Geometry*, Clarendon Press, Oxford.

Shapiro, L., Zisserman, A., and Brady, J. (1995). 3D motion recovery via affine epipolar geometry. *Int. Journal of Computer Vision*, 16(2):147–182.

Sotomayor, J. and Gutierrez, C. (1982). Structurally stable configurations of lines of principal curvature, *Astérisque* vol. 98–99, 195–215.

Springer, C. (1964). *Geometry and analysis of projective spaces*, Freeman.

Strang, G. (1988). *Linear Algebra and its Applications*, Harcourt-Brace-Jovanovich, 3rd edition.

Sullivan, S. and Ponce, J. (1998). Automatic model construction and pose estimation from photographs using triangular splines. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 20(10):1091–1096.

Szeliski, R. and Weiss, R. (1998). Robust shape recovery from occluding contours using a linear smoother. *Int. Journal of Computer Vision*, 28(1):27–44.

Torr, P.H.S. and Murray, D.W. (1997). The development and comparison of robust methods for estimating the fundamental matrix. *Int. Journal of Computer Vision*, 24(3):271–300.

Tsai, R.Y. and Huang, T.S. (1984). Uniqueness and estimation of three-dimensional motion parameters of ridid objects with curved surfaces. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 6(1):13–27.

Tomasi, C. and Kanade, T. (1992). Shape and motion from image streams under orthography: a factorization approach. *Int. Journal of Computer Vision*, 9(2):137–154.

Vaillant, R. and Faugeras, O.D. (1992). Using extremal boundaries for 3D object modelling. *IEEE Trans. Pattern Recognition and Machine Intelligence*, 14(2):157–173.

Zhang, Z. (1998). Determining the epipolar geometry and its uncertainty – a review. *Int. Journal of Computer Vision*, 27(2):161–195.