NOTICE WARNING CONCERNING COPYRIGHT RESTRICTIONS:
The copyright law of the United States (title 17, U.S. Code) governs the making of photocopies or other reproductions of copyrighted material. Any copying of this document without permission of its author may be prohibited by law.

Shape and Motion without Depth

Carlo Tomasi Takeo Kanade

May 1990

CMU-CS-90-128 2

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

This research was supported by the Defense Advanced Research Projects Agency (DOD) and monitored by the Avionics Laboratory, Air Force Wright Aeronautical Laboratories, Aeronautical Systems Division (AFSC), Wright-Patterson AFB, Ohio 45433-6543 under Contract F33615-87-C-1499, ARPA Order No. 4976, Amendment 20.

The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of DARPA or the U.S. government.

Keywords: computer vision, motion, shape, time-varying imagery

Abstract

Inferring the depth and shape of remote objects and the complete camera motion from a sequence of images is possible in principle, but is an ill-conditioned problem, because translation and rotation are hard to distinguish, and the size of the object is small with respect to its distance from the camera. We show how to overcome these problems by inferring shape and rotation without computing depth and camera translation as intermediate steps.

On a single epipolar plane, image measurements can be represented by an $F \times P$ matrix, obtained by tracking P points through F frames. We show that under orthographic projection this matrix is of rank 2.

Using this observation, we develop an algorithm to recover shape and camera rotation, based on singular value decomposition. The algorithm gives accurate results, and does not introduce smoothing in either shape or camera rotation.

Contents

Chapter 1

Introduction

In principle, the shape of an object can be computed from a sequence of images by estimating camera motion and depth, and inferring shape from the depth values.

However, when objects are distant from the camera, relative to their size, this computation is ill-conditioned. First, it is difficult to distinguish rotation from translation with adequate precision. Second, shape is computed from the small differences between large depth values.

These difficulties can be circumvented by inferring shape directly from variations in the distance between image features, without computing depth and camera translation as intermediate steps.

In this paper, we show how to infer shape and camera rotation from any number of features and frames, and reduce the computation to decomposing a matrix of image measurements.

The resulting algorithm, tested in simple situations, gives remarkably precise motion and shape estimates, without introducing smoothing effects into the result.

In 1979, Ullman proposed [Ullman, 1979] to compute shape and motion without going through depth. His first formulation assumed an orthographic projection model, and hence ignored the combined effects of depth and perspective distortion. He justified this simplification partly on the ground of mathematical tractability. The important point that computing depth leads to instability if the scene is remote did not receive all the emphasis it deserved.

Most of the work carrying out Ullman's proposal has concentrated on obtaining shape and motion with the minimal number of points and frames. These results are useful proofs of the existence of a solution. In this paper, we propose a way to incorporate any number of points and frames (greater than the minimum required) into the computation of shape and motion. For simplicity, we limit our consideration to one epipolar plane at a time, and assume that motion occurs in that plane. As a consequence, our images are single scanlines.

Our solution is based on two observations which, to our knowledge, have not appeared in the literature: under orthography, (1) the incidence relations among projection rays can be expressed as the degeneracy of the matrix of all the measurements; (2) the image coordinates of any two points in the epipolar plane trace an ellipse as the camera moves, if the coordinates are registered with respect to those of a third point.

Using these observations, we developed an algorithm that computes the shape of remote objects and the rotation of the camera. Since we use many, closely spaced frames, the results are insensitive to noise, and the correspondence problem is simplified.

As an illustration of the theory, we used our algorithm to recover the shape of a one-dollar silver coin (about 4 cm in diameter) at 3.5 meters distance from a real camera with a long lens. The total rotation of the camera was 30 degrees around the coin (and in the midplane of the coin). The error in the computed camera rotation is always less than one degree, and that in the shape of the coin is less than one percent of its diameter. These errors are mostly due to perspective effects, for which corrections are possible (but not made here).

In the following, we introduce our scenario, summarize the results, and sketch the relations of our work with previous literature on the subject. Section 2 proves the two geometric observations above. Section 3 shows how to use them to decompose the measurement matrix into shape and camera rotation. The experimental results in chapter 4 show the ability of the algorithm to deal with jerky rotations without smoothing its output. The conclusion (chapter 5) compares direct shape algorithms with algorithms which base the computation of shape on that of depth, and shows the former ones to be superior for remote scenes.

The Scenario

We assume that the camera produces an orthographic projection, rather than a perspective one. The world is still, and the camera moves in a plane, where it can freely rotate and/or translate. *P* features are visible in a given scanline, parallel to the plane of motion. Since the frames are taken frequently, it is easy to track the features from frame to frame. As the camera moves, it is panned so as to keep the features in the field of view.

In every frame, the image coordinate of an additional reference point is subtracted from the image coordinates of the P points. After F frames, an $F \times P$ matrix m of image measurements is available. This matrix is the input to the algorithm.

This is a rather artificial situation, but it approximates well what happens with a camera on an airplane, with suitable control mechanisms to align the camera scanlines with the direction of flight, and to keep the same object within the field of view. The farther away the objects are with respect to their size, the better the assumption of orthographic projection serves as an approximation.

The Results

This paper shows that if the measurements are noise-free, the measurement matrix m is highly degenerate (its rank is 2), and can be decomposed into the product of three smaller matrices: an $F \times 2$ matrix ρ, which encodes camera rotation, a $P \times 2$ matrix π, which encodes the positions of the world points, and a 2×2 diagonal matrix σ.

In reality, however, noise corrupts the measurements. The decomposition is still valid in an approximate sense, and σ tells how reliable the decomposition is.

The matrix m is factored into ρ, π, and σ by singular value decomposition [Golub and Reinsch, 1971], which is known to be efficient and numerically well behaved. If more points and frames are used than prescribed by equation-counting arguments (which require a minimum of three points, including the reference, and three frames), the effects of noise can be reduced.

The resulting shape and rotation algorithm is simple and efficient, and has been implemented and tested on small objects as distant as one hundred times their size (see chapter 4). The rotation errors are always smaller than one degree, and usually much smaller. The relative precision in the computed shape is of the order of the *relative depth range*, defined as the ratio between the size of the object along the optic rays and its distance from the camera.

The good performance of our algorithm derives from the fact that depth is not used as an intermediate result. For remote objects, the inference of depth is very sensitive to noise in the images, so that the quality of the depth estimates obtained by triangulation degrades as the relative range decreases. Consequently, the shape estimates worsen even faster, since the computation of shape from depth is itself ill-conditioned.

In our approach, instead, shape is related directly to the variations in the distances between image features from frame to frame. No triangulation is done, and the amount of camera translation becomes irrelevant.

Relations with Previous Work

Our goal is to compute camera motion and world point coordinates, relative to each other, from multiple frames.

In essence, our algorithm does what photogrammetrists for more than thirty years have known how to do by hand and with two frames at a time [Thompson, 1959]. Ullman proposed an automated solution to this problem eleven years ago [Ullman, 1979], and called it *structure-from-motion.*

Most of the initial efforts in this area have been devoted to finding closed-form solutions with a minimal or nearly-minimal number of points and/or frames (see, for instance, [Longuet-Higgins, 1981]).

In general, structure-from-motion is hard to solve. The major difficulty is the inherent sensitivity of the shape and motion results to noise in the image, especially when objects are distant. Performance degrades with reductions in the relative depth range. For instance, the algorithm presented in [Tsai and Huang, 1984] works very well for close objects, which is the intended goal of that paper, but the performance is likely to degrade when objects become more remote, and the relative depth range becomes smaller. If the images are noisy, few points and/or few frames give bad results, regardless of how good the math is.

The remedy is to use many frames and many points, exploiting redundancy to counteract noise. If frames are closely spaced, the correspondence problem is also easier to solve. This has been tried, with relatively good results, for the inference of depth when the motion of the camera is known. See for instance [Bolles *et al.*, 1987] or [Matthies *et al.*, 1989].

In [Spetsakis and Aloimonos, 1989], an interesting algorithm is presented for the case of unknown motion, using several frames and points and a perspective projection model. In spirit, our approach is akin to theirs: the projection lines of the same world point are a bundle (or pencil) of lines, and the resulting incidence relations between them allow casting the computation of shape and motion as a minimization problem. Our solution, however, does not recover depth or camera translation. We bypass this intermediate stage, and obtain a solution which is partial, but more reliable for remote scenes.

Chapter 2

The Decomposition Principles

In this chapter we introduce the two observations on which we base the computation of shape and motion. As we stated in the introduction, we consider only one scanline per frame, and assume that the camera moves in a plane parallel to the scanline.

In this plane, we define an orthogonal system of coordinates (X, Z), with the X axis along the scanline in the first frame. The origin of the system is a visible reference point on the object, as in figure 2.1.

The images are orthographic projections. Image points are *registered* by subtracting from their projections, x_{fp}, the projection of the reference point, x_{f0}:

$$m_{fp} = x_{fp} - x_{f0} \; . \tag{2.1}$$

There are P points, besides the reference point, and they are tracked through F frames. The registered measurements m_{fp} can then be collected in an $F \times P$ matrix

$$m = \begin{bmatrix} m_{11} & \cdots & m_{1P} \\ \vdots & & \vdots \\ m_{F1} & \cdots & m_{FP} \end{bmatrix} \; .$$

Registration is equivalent to translating every image along itself so that the reference point projects always to the same image location. In addition, we can translate every image along its projection rays so that it passes through the reference point. In summary, all images can be thought of as rotating around the reference point, as in figure 2.1.

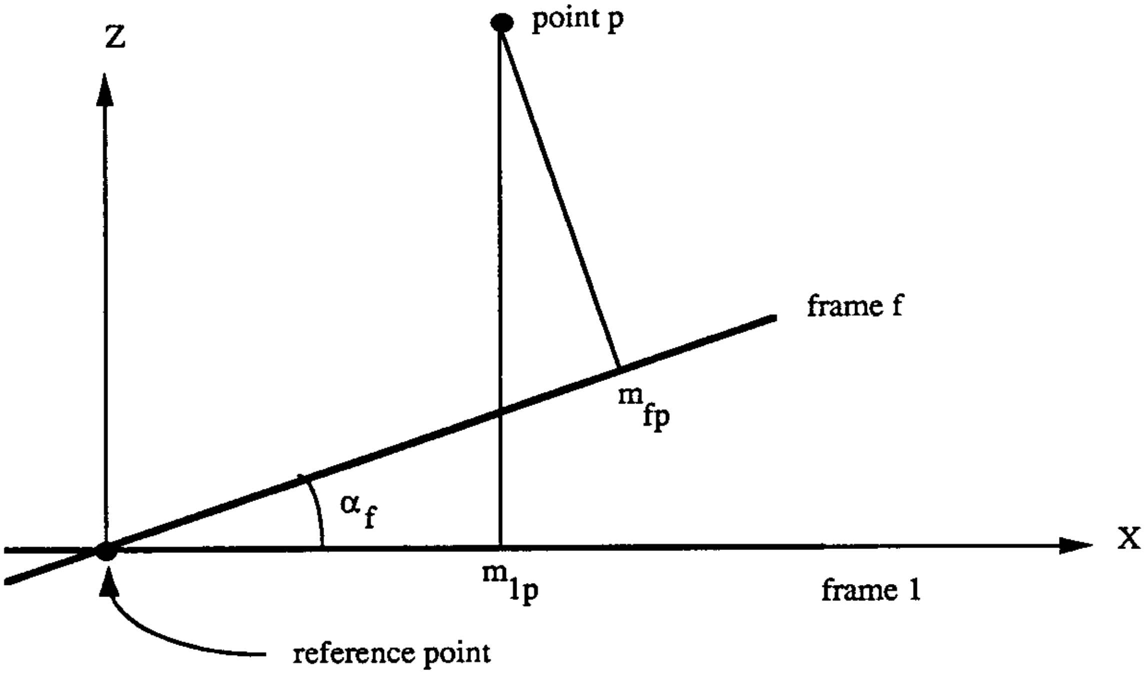

Figure 2.1: After registration, all image frames can be thought of as rotating around the visible reference point on the object.

From the figure, we see that the projection line of point p onto frame f is represented by the equation

$$c_f X + s_f Z = m_{fp} ,$$

where c_f and s_f are the cosine and sine of the angle α_f that frame f forms with frame 1 (thus, $\alpha_1 = 0$).

We now show two facts about the measurement matrix m. First, it is of rank 2. Second, given any two points p and q, the pairs (m_{fp}, m_{fq}) must be on an ellipse for all frames $f = 1, \ldots, F$.

The Rank Principle

Without noise, the rank of the measurement matrix m is two.

All the projection lines of point p belong to a pencil, since they must pass through point p itself. Therefore, for any three frames f, g, h, the projection line equations for point p,

$$\begin{aligned} c_f X + s_f Z &= m_{fp} \\ c_g X + s_g Z &= m_{gp} \\ c_h X + s_h Z &= m_{hp} , \end{aligned}$$

are linearly dependent, and the determinant

$$\det \begin{bmatrix} c_f & s_f & m_{fp} \\ c_g & s_g & m_{gp} \\ c_h & s_h & m_{hp} \end{bmatrix}$$

is equal to zero.

Thus, if we take any three points numbered p, q, r, and any three frames numbered f, g, h, we can write the *incidence* equations

$$\det \begin{bmatrix} c_f & s_f & m_{fp} \\ c_g & s_g & m_{gp} \\ c_h & s_h & m_{hp} \end{bmatrix} = \det \begin{bmatrix} c_f & s_f & m_{fq} \\ c_g & s_g & m_{gq} \\ c_h & s_h & m_{hq} \end{bmatrix} = \det \begin{bmatrix} c_f & s_f & m_{fr} \\ c_g & s_g & m_{gr} \\ c_h & s_h & m_{hr} \end{bmatrix} = 0 . \quad (2.2)$$

If we now read the matrices by columns, the incidence equations mean that the three vectors

$$\begin{bmatrix} m_{fp} \\ m_{gp} \\ m_{hp} \end{bmatrix} \quad \begin{bmatrix} m_{fq} \\ m_{gq} \\ m_{hq} \end{bmatrix} \quad \begin{bmatrix} m_{fr} \\ m_{gr} \\ m_{hr} \end{bmatrix}$$

all belong to the plane spanned by the two vectors

$$\begin{bmatrix} c_f \\ c_g \\ c_h \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} s_f \\ s_g \\ s_h \end{bmatrix} ,$$

and must therefore be coplanar, so that

$$\det \begin{bmatrix} m_{fp} & m_{fq} & m_{fr} \\ m_{gp} & m_{gq} & m_{gr} \\ m_{hp} & m_{hq} & m_{hr} \end{bmatrix} = 0 .$$

Thus, any third order determinant extracted from the exact measurement matrix $m = [m_{fp}]$ is equal to zero: the rank of m is smaller than 3. In appendix A we prove that, unless all points are aligned, some 2×2 determinant extracted from the matrix m must be non-zero, so that the rank of m is exactly 2. [1] The row space and the column space of m are two-dimensional.

Geometrically, this result means that the rows of m (one row per frame), interpreted as points in a P-dimensional space, must lie on a plane through the origin, call it the *frame plane*. The same holds for the columns of m (one column per point), interpreted as points in an F-dimensional space.

Intuitively, the rank principle says that the $F \times P$ measurements are not unrelated: they could be described in a simpler way by giving F frame angles and P points, if only these were known. That this degeneracy takes on the form of a simple rank equation (rank(m) = 2) is due to the linear nature of the orthographic projection equation.

The Ellipse Principle

> The registered image projections (m_{fp}, m_{fq}) of two points p and q, lie on an ellipse for all frames $f = 1, \ldots, F$.

[1] We henceforth ignore the case of all-aligned points.

The incidence equations (2.2) hold for any triple of projection lines relative to the same point. Then, the rank of the $F \times 3$ matrix

$$\left[\begin{array}{ccc} c & s & m_p \end{array} \right] = \left[\begin{array}{ccc} c_1 & s_1 & m_{1p} \\ \vdots & \vdots & \vdots \\ c_F & s_F & m_{Fp} \end{array} \right]$$

is also two. Since this holds for any p, we conclude that the cosine and sine vectors c and s belong to the frame plane. Any two independent vectors m_p and m_q (they are independent if points p and q are not aligned with the reference point) span the frame plane, and there must be four numbers (for a given pair p and q) $\alpha_c^{(pq)}$, $\beta_c^{(pq)}$, $\alpha_s^{(pq)}$, $\beta_s^{(pq)}$ such that

$$\begin{array}{rcl} c & = & \alpha_c^{(pq)} m_p + \beta_c^{(pq)} m_q \\ s & = & \alpha_s^{(pq)} m_p + \beta_s^{(pq)} m_q \ . \end{array}$$

By squaring and adding these two vector equations component by component, we obtain the following F equations

$$[(\alpha_c^{(pq)})^2+(\alpha_s^{(pq)})^2]m_{fp}^2+[(\beta_c^{(pq)})^2+(\beta_s^{(pq)})^2]m_{fq}^2+2(\alpha_c^{(pq)}\beta_c^{(pq)}+\alpha_s^{(pq)}\beta_s^{(pq)})m_{fp}m_{fq} = 1 \ ,$$

for p and q fixed and $f = 1, \ldots, F$. These equations say that all pairs (m_{fp}, m_{fq}) lie on the same ellipse, centered at the origin. We can draw $P(P-1)/2$ ellipses, one for every pair of points p and q.

Intuitively, this can be understood by the following thought experiment: if the camera were to rotate at uniform angular velocity, the projection of each point would be a sinusoidal function of time. If two such sinusoids represent the orthogonal coordinates of a point moving on a plane, the point traces an ellipse. In fact, this is how Lissajous figures are drawn on an oscilloscope. Let us now remove the condition of constant camera rotation. The phase relation between the two sinusoids is preserved, because the two coordinates of each point on the ellipse refer to the same camera frame. Therefore, we obtain the same ellipse, but sampled at irregular intervals.

Chapter 3

The Algorithm: Dealing with Noise

When images are noisy, the measurement matrix m will not be exactly of rank 2. However, the rank principle can be extended to the case of noisy measurements. We do this by using the concept of Singular Value Decomposition (SVD) [Golub and Reinsch, 1971] to introduce the notion of approximate rank. The ellipse principle is also readily extended, by replacing interpolation (the points are *on* the ellipse) with fitting (the points are *near* an ellipse).

In this chapter, we examine these extensions, and show how to use the extended principles to compute shape and motion from a matrix of noisy image measurements.

Assuming [1] that $F \geq P$, m can be decomposed [Golub and Reinsch, 1971] into an $F \times P$ matrix ρ, a diagonal $P \times P$ matrix σ, and a $P \times P$ matrix π, such that

$$\begin{aligned} m &= \rho\sigma\pi^T \\ \rho^T\rho = \pi^T\pi = \pi\pi^T &= I \\ \sigma_1 \geq \ldots &\geq \sigma_P \end{aligned} \qquad (3.1)$$

where I is the $P \times P$ identity matrix, and the *singular values* $\sigma_1, \ldots, \sigma_P$ are the diagonal entries of σ. This is called the *Singular Value Decomposition* (SVD) of the matrix m.

We can now restate the rank principle for noisy measurements.

> The first two singular values of the noisy measurement matrix m are much greater than the others:
>
> $$\sigma_1\,,\ \sigma_2 \gg \sigma_3\ . \qquad (3.2)$$

[1] This assumption is not crucial: if $F < P$, everything can be repeated for the transpose of m.

Golub and Reinsch [Golub and Reinsch, 1971] give an efficient and well behaved algorithm to compute the decomposition. Consider now the matrix μ which is obtained by setting to zero all the singular values after σ_2 in the decomposition (3.1):

$$\mu = \sigma_1 \rho_1 \pi_1^T + \sigma_2 \rho_2 \pi_2^T \ , \tag{3.3}$$

where the first two columns of ρ are denoted by ρ_1 and ρ_2, and the first two columns of π are denoted by π_1 and π_2. It can be shown [Forsythe *et al.*, 1977] that the 2-norm of the (matrix) difference between m and μ is smaller than σ_3. Hence, the value of σ_3 can serve to assess the quality of the approximation $m \approx \mu$. If equation (3.2) holds, we can expect μ to be a clean version of m, after removing noise in the least square error sense.

The two vectors ρ_1 and ρ_2 are a basis for the frame plane, that is, for the column space of μ. Then, we can apply the ellipse principle to these vectors, rather than to two columns μ_p and μ_q of the measurement matrix; the $P(P-1)/2$ ellipses found in the previous chapter, one for every pair of points p and q, are now replaced by one ellipse, whose coefficients account for *all* of the measurements through the vectors ρ_1 and ρ_2:

$$(\alpha_c^2 + \alpha_s^2)\rho_{f1}^2 + (\beta_c^2 + \beta_s^2)\rho_{f2}^2 + 2(\alpha_c\beta_c + \alpha_s\beta_s)\rho_{f1}\rho_{f2} \approx 1 \ . \tag{3.4}$$

The reason for the approximate equality is that the two vectors ρ_1 and ρ_2 are a basis only for the *best estimate* of the frame plane, not for the *true* frame plane. Therefore, if we require the two vectors c and s to lie on the estimated measurement plane, the normalization conditions $c_f^2 + s_f^2 = 1$ will hold only approximately.

The remaining steps needed to complete the solution are the following:

- find the coefficients $a^2 = \alpha_c^2 + \alpha_s^2$, $b^2 = \beta_c^2 + \beta_s^2$, and $d = \alpha_c\beta_c + \alpha_s\beta_s$ of the ellipse by solving the following overconstrained $F \times 3$ system of equations in the least square error sense:

$$\begin{bmatrix} \rho_{11}^2 & \rho_{12}^2 & 2\rho_{11}\rho_{12} \\ \rho_{21}^2 & \rho_{22}^2 & 2\rho_{21}\rho_{22} \\ \vdots & \vdots & \vdots \\ \rho_{F1}^2 & \rho_{F2}^2 & 2\rho_{F1}\rho_{F2} \end{bmatrix} \begin{bmatrix} a^2 \\ b^2 \\ d \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \ ;$$

- find α_c, β_c, α_s, β_s, from a^2, b^2, d by imposing the additional constraint that

$\alpha_1 = 0$ (the first frame is chosen as the X axis). This yields

$$\alpha_c = \frac{d + \beta_s^2 r}{\beta_c} \quad \beta_c = \sqrt{\frac{d(d + 2rb^2) + b^4 r^2}{a^2 + 2dr + b^2 r^2}}$$

$$\alpha_s = -r\beta_s \qquad \beta_s = \sqrt{b^2 - \beta_c^2}$$

where $r = \rho_{12}/\rho_{11}$ (ratio of the first components of ρ_2 and ρ_1);

- compute

  $$c = \alpha_c \rho_1 + \beta_c \rho_2 \qquad s = \alpha_s \rho_1 + \beta_s \rho_2 \,. \tag{3.5}$$

  These are the two vectors on the frame plane which best satisfy the normalization conditions $c_f^2 + s_f^2 = 1$;

- find two vectors c' and s' that satisfy the normalization equations exactly, and that are as close as possible to c and s. Here, the correct metric is the Euclidean metric in the space of the measurements m_{fp}: we want to move from c to c' and from s to s' while perturbing the values of the measurements as little as possible. As shown in appendix B, this is equivalent to changing the vectors ρ_1 and ρ_2 into two new vectors ρ_1' and ρ_2' so as to minimize $\sum_{f=1}^{F}[\sigma_1(\rho_{f1}' - \rho_{f1})^2 + \sigma_2(\rho_{f2}' - \rho_{f2})^2]$, subject to the normalization constraints. This is a simple Lagrange minimization problem. Its solution yields the cosines and sines of the frame angles α_f, that is, the camera rotation;

- compute the coordinates X_p and Z_p of every object point p by finding the least square error intersection of all its projection lines. This is done in appendix C.

These steps have been implemented in a computer program, which was tested on several image sequences. The next chapter describes an illustrative experiment.

Chapter 4

An Experiment

The purpose of the experiment described in this chapter is to illustrate the rank an ellipse principles, demonstrate the good quality of the results, and quantify the influence of perspective effects on the accuracy of the motion estimates.

The key parameter is the relative depth range, which we defined as the ratio of the object size along the projection rays and the distance between camera and object. The relative errors in the computed shape are of the same order as the relative depth range, and modeling inaccuracies that are small with respect to it can be ignored.

We put a one-dollar coin (about 4 cm in diameter) approximately 3.5 meters away from a Sony CCD camera with a 300 mm Tokina lens. Thus, the relative depth range was $4/350 \approx 0.011$. Figure 4.1 shows the setup.

The camera was moved in the plane of the coin, so that only the edge of the coin was visible in every frame. The motion was roughly circular around a point in the vicinity of the coin. Only the rotation component was controlled with an accurate positioning mechanism, so that a precise reference was available for performance evaluation.

The edge of the coin was approximately aligned with the image scanlines, thus yielding easy-to-track image features (the thin vertical notches on the coin edge). The first 101 frames were taken in steps of 0.1 degrees between consecutive frames; after that, the velocity was doubled to 0.2 degrees per frame, and 100 more frames were taken; thus, the overall rotation was 30 degrees. The 201 scanlines are stacked together in figure 4.2, top to bottom. This figure is what is called an epipolar plane in [Bolles *et al.*, 1987].

The image was filtered with a thirteen-tap finite impulse response approximation to a Laplacian of a Gaussian, and the zero crossings of the result (figure 4.3) were used as features in the experiment (104 crossings were found).

The rank principle is illustrated graphically by the similarity of figures 4.4 and 4.5. Figure 4.4 shows the crossing of figure 4.3 after registration (equation (2.1)). To obtain figure 4.5, we decomposed the matrix m representing the registered crossings, set to zero all the singular values except the first two, and reconstructed the measurement matrix from the first two columns of the SVD factors (equation (3.3)). The rank principle says that the only differences between figure 4.4 and figure 4.5, under orthography, are due to noise.

The singular values are plotted in figure 4.6; without noise, and if the projection were exactly orthographic, only the first two values would be different from zero. The third value (σ_3) reflects essentially the effect of perspective.

Figure 4.7 illustrates the ellipse principle. It shows the points (ρ_{f1}, ρ_{f2}) from the left factor of the singular value decomposition of the measurement matrix m, and the best fit ellipse, as defined by equation (3.4).

In spite of perspective effects and unmodeled small variations in depth, the quality of both shape and motion results is remarkably good. Figure 4.8 shows the computed and the true rotation. The error is always smaller than one degree, and almost everywhere much smaller than that. The algorithm assumes no motion models, and does no smoothing. As a result, the sharp change in rotational velocity is preserved in the motion output.

Figure 4.9 shows the shape results, and the best circular fit to them. The accuracy of shape is of the order of the relative depth range (1 percent), even if variations in depth during the motion of the camera were of the order of the coin size.

To get an idea of how perspective effects influence the accuracy of the results, we tested our algorithm on a sequence of simulated, noise-free images similar to those of our coin experiment. A circular object with 10 features is placed at various depths from the camera. For each depth, a pinhole camera moves and rotates by 30 degrees in 30 steps. Figure 4.10 plots the relative error in the total computed rotation as a function of the relative depth range. While algorithms based on depth give worse motion estimates as objects are moved farther away, our algorithm improves (for a constant total rotation angle), because it approximates orthography better and better.

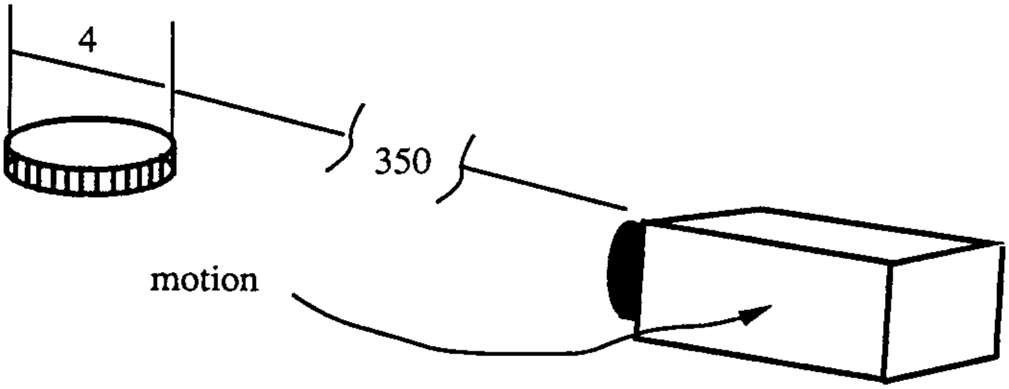

Figure 4.1: The setup in our experiment. Measures are in centimeters.

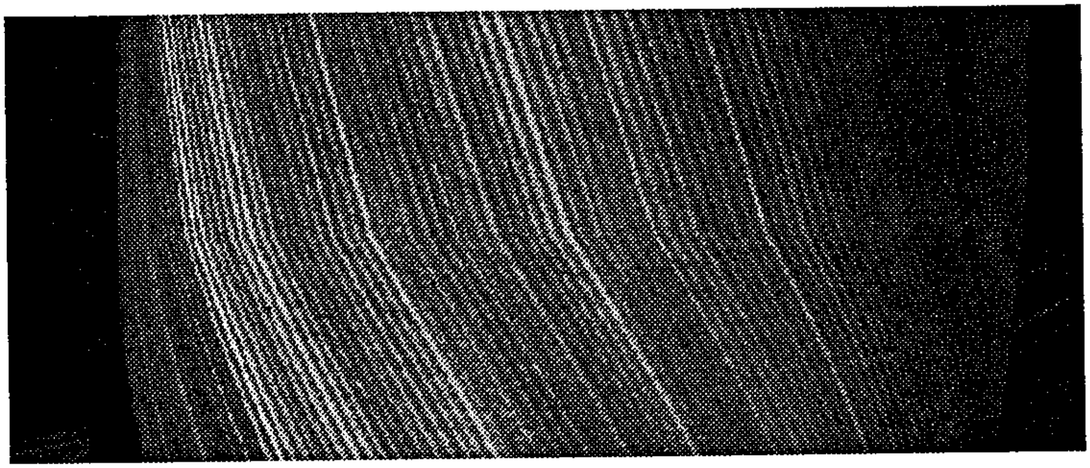

Figure 4.2: The input to the algorithm; each scanline is a new frame, and represents the edge of a one-dollar coin seen from a new angle. In [Bolles *et al.*, 1987], a figure like this is called an epipolar plane. We use it to recover shape and rotation, instead of depth given known motion.

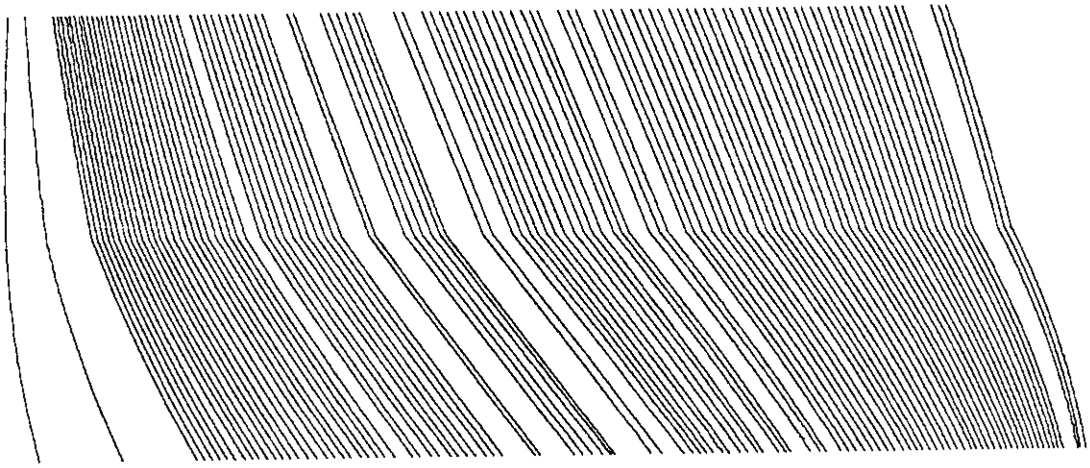

Figure 4.3: The zero crossings from figure 4.2.

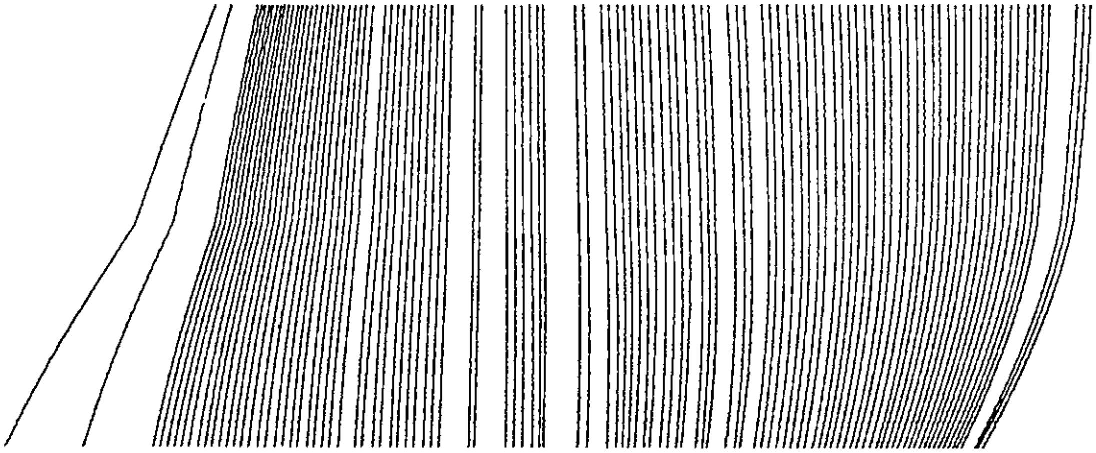

Figure 4.4: The zero crossings of figure 4.3 after registration. See equation 2.1.

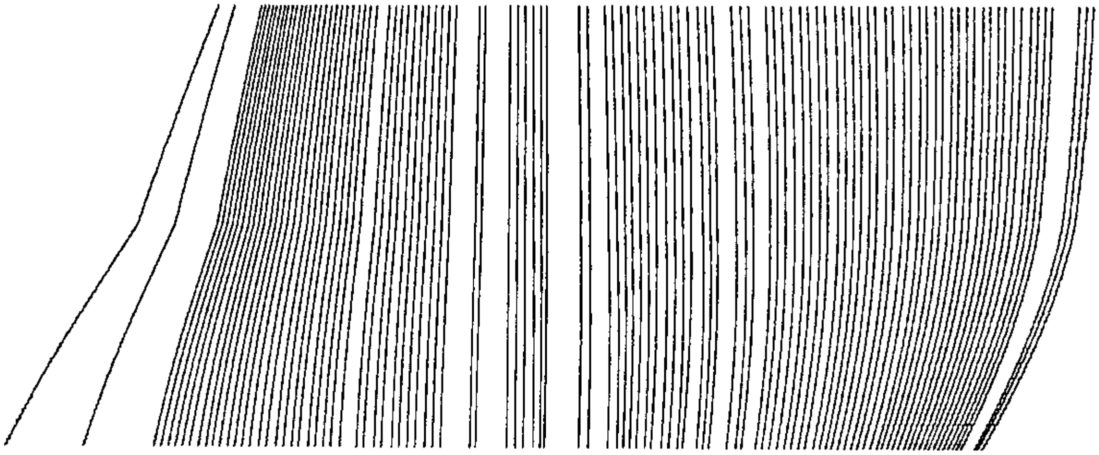

Figure 4.5: Registered zero crossings reconstructed after suppressing all but the first two singular values of the measurement matrix.

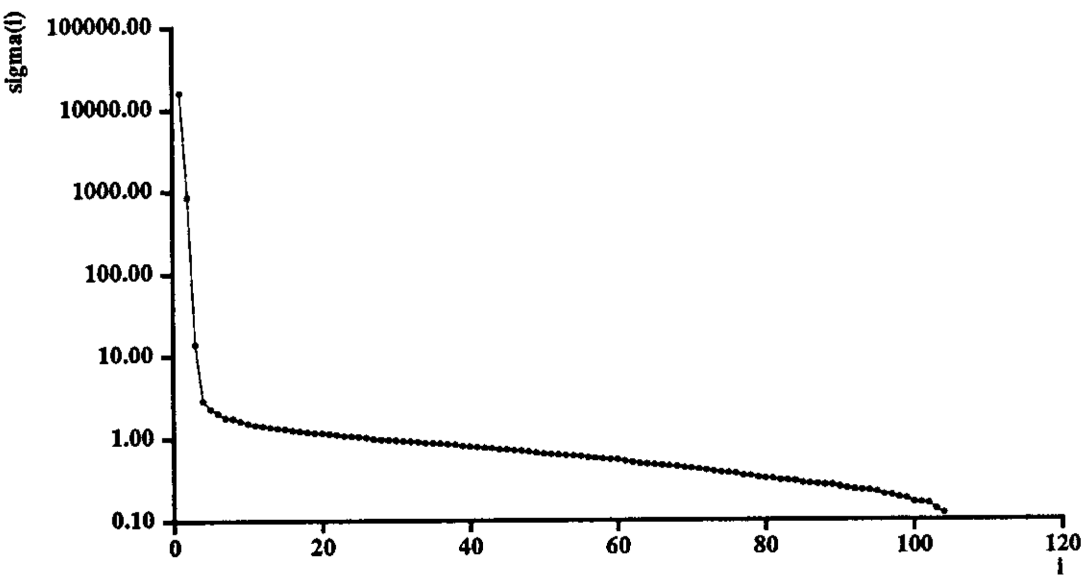

Figure 4.6: Singular values of the measurement matrix.

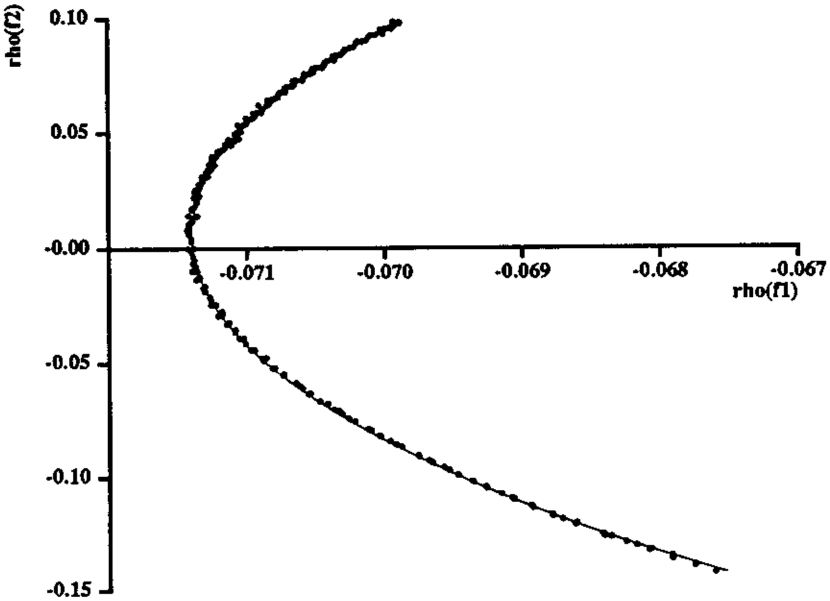

Figure 4.7: The ellipse which best fits the columns of the ρ matrix (dots are the actual values of (ρ_{f1}, ρ_{f2})).

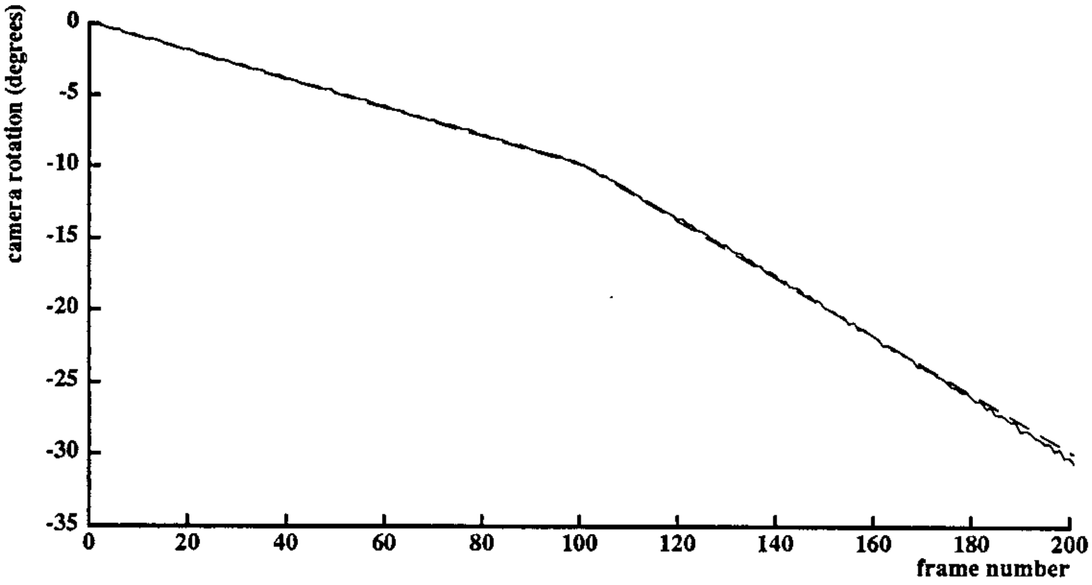

Figure 4.8: Computed (solid) versus true (dashed) camera rotation.

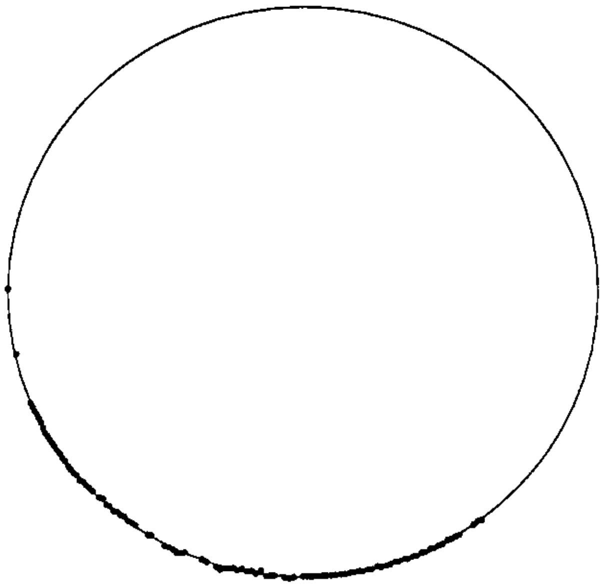

Figure 4.9: Computed shape (dots) of a one-dollar coin, with the best fit circle.

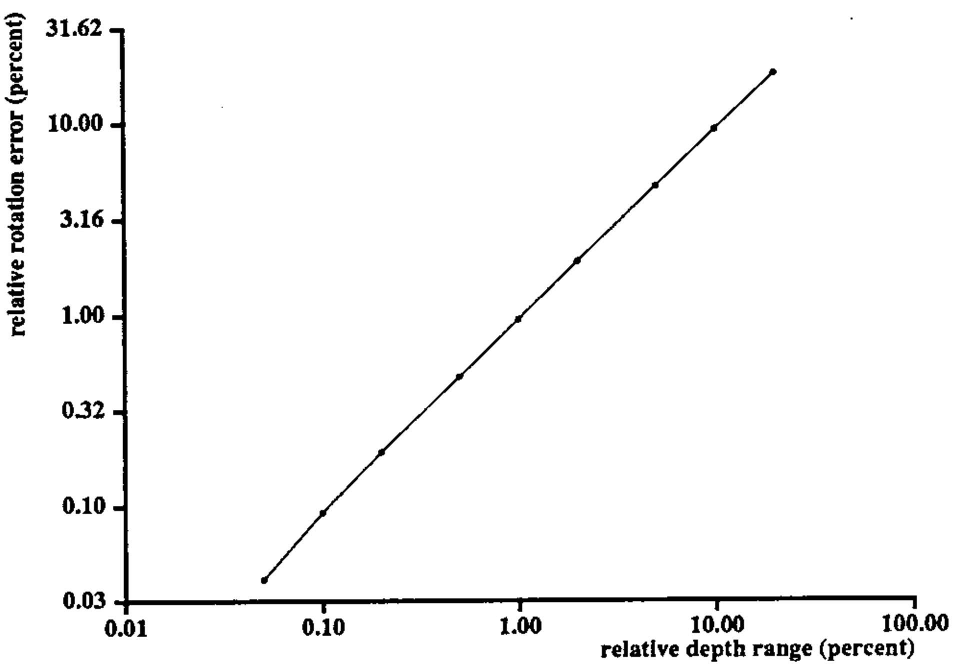

Figure 4.10: The motion error due to perspective distortion decreases when the relative depth range becomes smaller. These results were obtained by simulating noise-free images of a circular object with 10 features, and a pin-hole camera rotating by 30 degrees in 30 frames.

Chapter 5

Conclusion: Depth versus Shape Algorithms

The algorithm presented in this paper infers the shape of remote objects and the rotation of the camera. It is a *shape* algorithm. It does not compute either depth or camera translation.

Algorithms such as the ones described in [Tsai and Huang, 1984], [Heel, 1989], [Spetsakis and Aloimonos, 1989], on the other hand, represent depth explicitly, and compute it from the image sequence. They are *depth* algorithms.

Depth algorithms give a more complete answer. They compute all components of motion, up to a scale factor, and the depth information they supply allows, in principle, computing shape as well.

However, depth algorithms do not work if objects are very distant from the camera with respect to their size. When the relative depth range is very small, as for instance in aerial cartography and reconnaissance, the values of depth are poorly constrained by the image sequence, and it is hard to distinguish rotation from translation.

In these situations, the completeness of depth algorithms is not only useless, but harmful. A shape algorithm gives a more stable and accurate answer, because it computes shape and camera rotation directly from image deformations. It does not use depth as an intermediate result, and it need not distinguish translation from rotation.

The results of this paper can be extended along four independent directions: accuracy, threedimensionality, completeness, and efficiency.

Accuracy can be increased by correcting for perspective effects. Once a good shape estimate has been computed, the solution can be perturbed with a steepest descent search to account for the slight divergence of projection rays in each frame. Furthermore, if relative changes in depth are large with respect to the relative depth range, looming effects must be estimated and accounted for.

The algorithm can be extended to three dimensions. For obvious reasons of applicability, this is the direction we have chosen to pursue first in our future research.

Completeness: if a motion model is available, depth and translation can be estimated independently. Shape and rotation, computed by our algorithm, would be *inputs* to a separate depth and translation algorithm, possibly together with external motion information. Shape and depth are often several orders of magnitude apart. We have shown that they should be estimated separately, not that depth cannot be estimated.

Our implementation of the algorithm uses an efficient singular value decomposition routine. However, it treats a whole batch of frames at once. An incremental implementation would be more desirable. The feasibility of this is being investigated.

Acknowledgements

Thanks to Steve Shafer for kindly making his Calibrated Imaging Laboratory available for our experiments, and to Raul Valdes-Perez for helpful comments on the manuscript.

Bibliography

[Bolles *et al.*, 1987]
R. C. BOLLES, H. H. BAKER, AND D. H. MARIMONT. Epipolar-plane image analysis: An approach to determining structure from motion. *International Journal of Computer Vision*, 1(1):7–55, 1987.

[Forsythe *et al.*, 1977]
G. E. FORSYTHE, M. MALCOLM, AND C. B. MOLER. *Computer Methods for Mathematical Computations*. Prentice-Hall, Englewood Cliffs, NJ, 1977.

[Golub and Reinsch, 1971]
G. H. GOLUB AND C. REINSCH. Singular value decomposition and least squares solutions, In *Handbook for Automatic Computation*, volume 2, chapter I/10, pages 134–151. Springer Verlag, New York, NY, 1971.

[Heel, 1989]
J. HEEL. Dynamic motion vision. In *Proceedings of the DARPA Image Understanding Workshop*, pages 702–713, Palo Alto, Ca, May 23-26 1989.

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

[Matthies *et al.*, 1989]
L. MATTHIES, T. KANADE, AND R. SZELISKI. Kalman filter-based algorithms for estimating depth from image sequences. *International Journal of Computer Vision*, 3(3):209–236, September 1989.

[Spetsakis and Aloimonos, 1989]
M. E. SPETSAKIS AND J. Y. ALOIMONOS. Optimal motion estimation. In

Proceedings of the IEEE Workshop on Visual Motion, pages 229–237, Irvine, California, March 1989.

[Thompson, 1959]
E. H. THOMPSON. A rational algebraic formulation of the problem of relative orientation. *Photogrammetric Record*, 3(14):152–159, 1959.

[Tsai and Huang, 1984]
R. Y. TSAI AND T. S. HUANG. Uniqueness and estimation of three-dimensional motion parameters of rigid objects with curved surfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6(1):13–27, January 1984.

[Ullman, 1979]
S. ULLMAN. *The Interpretation of Visual Motion*. The MIT Press, Cambridge, Ma, 1979.

Appendix A

The Minors of the Measurement Matrix

In this appendix, we relate the three determinants

$$\Delta_{12}^{(12)} = \det \begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix} \quad \Delta_{13}^{(12)} = \det \begin{bmatrix} m_{11} & m_{12} \\ m_{31} & m_{32} \end{bmatrix} \quad \Delta_{23}^{(12)} = \det \begin{bmatrix} m_{21} & m_{22} \\ m_{31} & m_{32} \end{bmatrix}$$

to intrinsic geometric parameters which describe the relative position of the three world points, and to the angles between frames.

It immediately follows from this interpretation that a necessary and sufficient condition for the three determinants above to be different from zero is that no two object points be coincident, no three points be aligned, and no two frames coincide.

Let d_p and γ_p be the magnitude and phase of the vector which joins the reference point with object point number p:

$$\begin{aligned} d_p &= \sqrt{X_p^2 + Z_p^2} \\ \gamma_p &= \arctan_2(Z_p, X_p) \end{aligned}$$

(see figure A.1).

Here $\arctan_2$ is the two-argument inverse tangent function, which differs from the one-argument function in that it returns the angle in the appropriate quadrant, and has no singularities:

$$\arctan_2(y,x) = \begin{cases} \arctan(y/x) & \text{if } x > 0 \\ \text{sign}(y)[\pi - \arctan(|y/x|)] & \text{if } x < 0 \\ 0 & \text{if } x = y = 0 \\ \text{sign}(y)\pi/2 & \text{if } x = 0,\ y = 0 \end{cases} .$$

Furthermore, let ψ_{fg} be the angle between frame f and frame g, measured counterclockwise from f to g (figure A.1).

Then, if m_{fp} is the projection of point p onto frame f (after registration), we have

$$\Delta_{fg}^{(pq)} = \det \begin{bmatrix} m_{fp} & m_{fq} \\ m_{gp} & m_{gq} \end{bmatrix} = d_p d_q \sin\gamma_{pq} \sin\psi_{fg} .$$

Proof

We introduce the angles ω_{fp} between frame f and the line from the origin to point p; the determinant $\Delta_{fg}^{(pq)}$ is easily expressed in terms of these angles:

$$\begin{aligned} \Delta_{fg}^{(pq)} &= \det \begin{bmatrix} d_p \cos\omega_{fp} & d_q \cos\omega_{fq} \\ d_p \cos\omega_{gp} & d_q \cos\omega_{gq} \end{bmatrix} \\ &= d_p d_q (\cos\omega_{fp} \cos\omega_{gq} - \cos\omega_{fq} \cos\omega_{gp}) \\ &= \frac{d_p d_q}{2} [\cos(\omega_{fp} + \omega_{gq}) + \cos(\omega_{fp} - \omega_{gq}) \\ &\quad - \cos(\omega_{fq} + \omega_{gp}) - \cos(\omega_{fq} - \omega_{gp}) . \end{aligned}$$

If we now observe that

$$\begin{aligned} \omega_{fq} &= \omega_{fp} + \gamma_{pq} \\ \omega_{fp} &= \psi_{fg} + \omega_{gp} = \psi_{fg} + \omega_{gq} - \gamma_{pq} \end{aligned}$$

we can write

$$\begin{aligned} \omega_{fp} + \omega_{gq} &= \omega_{fq} + \omega_{gp} = 2\omega_{fp} - \psi_{fg} + \gamma_{pq} \\ \omega_{fp} - \omega_{gq} &= \psi_{fg} - \gamma_{pq} \\ \omega_{fq} - \omega_{gp} &= \psi_{fg} + \gamma_{pq} , \end{aligned}$$

so that

$$\Delta_{fg}^{(pq)} = \frac{d_p d_q}{2} [\cos(\psi_{fg} - \gamma_{pq}) - \cos(\psi_{fg} + \gamma_{pq})] = d_p d_q \sin\gamma_{pq} \sin\psi_{fg} ,$$

as promised.

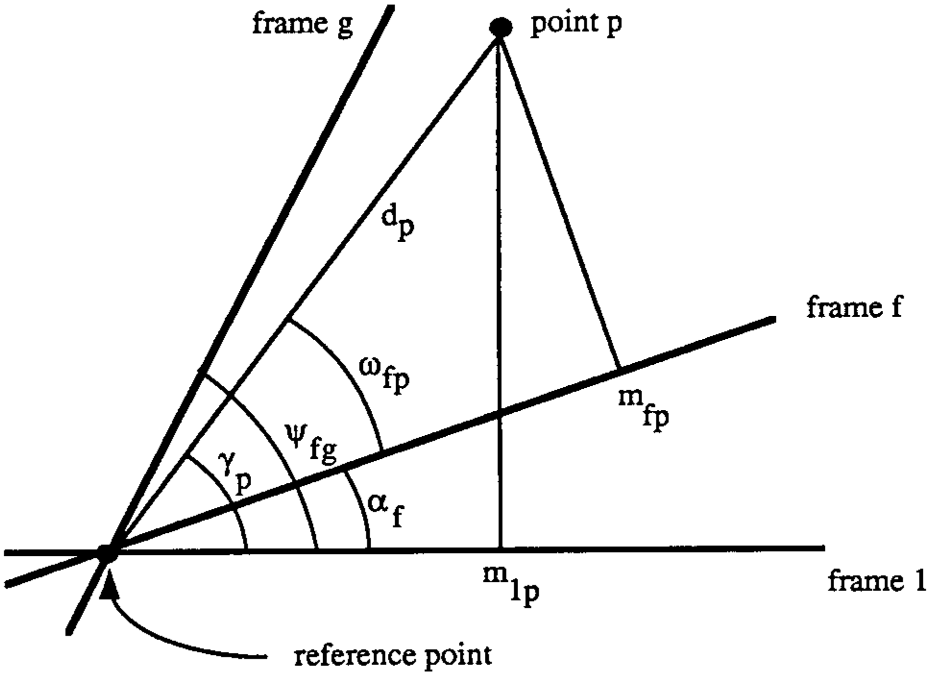

Figure A.1: The angles defined in appendix A.

Appendix B

The Normalization Equations for a Noisy Measurement Matrix

On page 12, we computed the cosines c_f and sines s_f of the frame angles α_f in two steps.

We first found those values of c_f and s_f that lie on the frame plane, and that best satisfy the normalization conditions $c_f^2 + s_f^2 = 1$. We then perturbed c_f and s_f into new values c_f' and s_f' that satisfy the normalization equations exactly, and that are as close as possible to c_f and s_f.

We identified the correct metric for measuring the amount of this perturbation as the Euclidean metric in the space of the original, registered image measurements m_{fp}.

In this appendix, we show that solving this problem is equivalent to changing the the first two columns ρ_1 and ρ_2 of the left factor of the singular value decomposition of the measurement matrix m into two new vectors ρ_1' and ρ_2' so as to minimize the sum

$$\sum_{f=1}^{F}[\sigma_1(\rho_{f1}' - \rho_{f1})^2 + \sigma_2(\rho_{f2}' - \rho_{f2})^2] ,$$

subject to the normalization constraints

$$(c_f')^2 + (s_f')^2 = 1$$

for $f = 1, \ldots, F$.

From the definition of the clean measurement matrix μ (equation 3.3), we see that if we change ρ_{f1} and/or ρ_{f2}, we alter only one row of $\hat{x}$: in fact, the f-th row of that equation is

$$\left[\begin{array}{ccc} \hat{x}_{f1} & \dots & \hat{x}_{fP} \end{array}\right] = \rho_{f1}\sigma_1\phi_1^T + \rho_{f2}\sigma_2\phi_2^T$$

or, in matrix notation,

$$\hat{x}_{f*}^T = \left[\begin{array}{cc} \rho_{f1} & \rho_{f2} \end{array}\right] \sigma \left[\begin{array}{c} \phi_1^T \\ \phi_2^T \end{array}\right] .$$

This is intuitive: the coefficients ρ_{f1} and ρ_{f2} regard only the measurements in frame number f, so it stands to reason that changing these coefficients affects only measurements in frame f.

Then, a change $\epsilon^T = (\epsilon_1, \epsilon_2)$ in (ρ_{f1}, ρ_{f2}) results in a change

$$\eta^T = (\eta_1, \dots, \eta_P) = \epsilon^T \sigma \left[\begin{array}{c} \phi_1^T \\ \phi_2^T \end{array}\right]$$

in $\hat{x}_{f*}^T = (\hat{x}_{f1}, \dots, \hat{x}_{fP})$. The squared norm of η is

$$\|\eta\|^2 = \eta^T\eta = \epsilon^T \sigma \left[\begin{array}{c} \phi_1^T \\ \phi_2^T \end{array}\right] \left[\begin{array}{cc} \phi_1 & \phi_2 \end{array}\right] \sigma\epsilon = \epsilon^T\sigma^2\epsilon ,$$

where σ (and therefore σ^2) is diagonal. Notice that the simplicity of the result follows from the orthonormal nature of the matrix ϕ (equation 3.1).

Thus,

$$\|\eta\| = \sqrt{e^T e} ,$$

where

$$e = \sigma\epsilon = \sigma_1\epsilon_1 + \sigma_2\epsilon_2 .$$

As a consequence, we can almost use a Euclidean metric in the space of the points (ρ_{f1}, ρ_{f2}), except that the two coordinates must be scaled by σ_1 and σ_2.

The problem of computing (ρ'_{f1}, ρ'_{f2}) from (ρ_{f1}, ρ_{f2}) is now easily stated: find the point (ρ'_{f1}, ρ'_{f2}) such that the norm of the vector

$$\left[\begin{array}{cc} \sigma_1(\rho'_{f1} - \rho_{f1}) & \sigma_2(\rho'_{f2} - \rho_{f2}) \end{array}\right]$$

is minimized, subject to the normalization constraint

$$(c_f')^2 + (s_f')^2 = 1 \ .$$

We can rewrite this constraint in terms of the vectors ρ_1' and ρ_2' by noticing that

$$c' \ = \ \alpha_c \rho_1' + \beta_c \rho_2' \tag{B.1}$$

$$s' \ = \ \alpha_s \rho_1' + \beta_s \rho_2' \tag{B.2}$$

(compare with equation 3.5).

If we introduce the matrix

$$A = \begin{bmatrix} \alpha_c & \beta_c \\ \alpha_s & \beta_s \end{bmatrix} \ ,$$

the normalization constraints become

$$\begin{bmatrix} \rho_{f1}' & \rho_{f2}' \end{bmatrix} A^T A \begin{bmatrix} \rho_{f1}' \\ \rho_{f2}' \end{bmatrix} = 1 \ .$$

The solution to this constrained minimization problem is a simple application of the technique of Lagrange multipliers. The Euler equation is

$$\sigma(\rho' - \rho) + \lambda A^T A \rho' = 0 \ ,$$

where for brevity we let $\rho = (\rho_{f1}, \rho_{f2})^T$, and similarly for ρ'. This yields

$$\rho' = (\sigma + \lambda A^T A)^{-1} \sigma \rho \ . \tag{B.3}$$

By replacing this result into the constraint equation $(\rho')^T A^T A \rho = 1$, we obtain a fourth order equation in the Lagrange multiplier λ:

$$\rho^T \sigma (\sigma + \lambda A^T A)^{-1} A^T A (\sigma + \lambda A^T A)^{-1} \sigma \rho = 1 \ ,$$

whose solutions determine the candidates for λ. To find ρ', replace the solutions in turn into equation B.3, and check which one yields the smaller norm for the difference vector.

Appendix C

The Intersection of the Projection Lines

The last step in the computation of shape is to compute the coordinates X_p and Z_p of the object points. For a given point p, this can be done by intersecting the projection lines of the point.

Since there are F projection lines for each point, the solution is overconstrained. In this appendix, we show that the minimization problem solved in appendix B in order to enforce the normalization equations yields also the solution to our intersection problem. The point coordinates X_p and Z_p can then be computed directly from the perturbed vectors ρ_1' and ρ_2' found in appendix B (equation B.3).

The frame angles α_f were determined from a set of noisy measurements; therefore, we cannot expect the F projection lines of point p,

$$c_f X + s_f Z = m_{fp} \quad \text{for} \quad f = 1, \dots, F \; ,$$

to intersect exactly at one point. This does not even hold for the estimated projection lines

$$c_f X + s_f Z = \mu_{fp} \quad \text{for} \quad f = 1, \dots, F \; ,$$

since, although the clean measurement matrix μ *is* of rank two, the sines and cosines were computed from the modified versions ρ_1' and ρ_2' of ρ_1 and ρ_2.

However, if we now let

$$\hat{\mu} = \sigma_1 \rho_1' \pi_1^T + \sigma_2 \rho_2' \pi_2^T \tag{C.1}$$

(compare with equation 3.3), the F lines

$$c_f X + s_f Z = \hat{\mu}_{fp} \quad \text{for} \quad f = 1, \ldots, F \tag{C.2}$$

do all intersect at one point.

In fact, on one hand, entry (f, p) of equation C.1 is

$$\hat{\mu}_{fp} = \sigma_1 \rho'_{f1} \pi_{p1} + \sigma_2 \rho'_{f2} \pi_{p2} \,. \tag{C.3}$$

On the other hand, if we replace the expressions for c_f and s_f given by B.2 into the projection line equations C.2 we obtain

$$\hat{\mu}_{fp} = (\alpha_c X + \alpha_s Z)\rho'_{f1} + (\beta_c X + \beta_s Z)\rho'_{f2} \quad \text{for} \quad f = 1, \ldots, F \,. \tag{C.4}$$

These F lines intersect if there is a point (X, Z) which satisfies all the F equations simultaneously. By comparing equation C.4 with equation C.3, we see that such a point exists if there is a solution to the system

$$\begin{aligned} \alpha_c X + \alpha_s Z &= \sigma_1 \pi_{p1} \\ \beta_c X + \beta_s Z &= \sigma_2 \pi_{p2} \,, \end{aligned}$$

or, in matrix notation

$$A \begin{bmatrix} X \\ Z \end{bmatrix} = \sigma \begin{bmatrix} \pi_{p1} \\ \pi_{p2} \end{bmatrix} .$$

We already know that the matrix

$$A = \begin{bmatrix} \alpha_c & \alpha_s \\ \beta_c & \beta_s \end{bmatrix}$$

is non singular, so that the desired solution is

$$\begin{bmatrix} X_p \\ Z_p \end{bmatrix} = A^{-1} \sigma \begin{bmatrix} \pi_{p1} \\ \pi_{p2} \end{bmatrix} .$$