11

Computation of the Fundamental Matrix F

This chapter describes numerical methods for estimating the fundamental matrix given a set of point correspondences between two images. We begin by describing the equations on F generated by point correspondences in two images, and their minimal solution. The following sections then give linear methods for estimating F using algebraic distance, and then various geometric cost functions and solution methods including the MLE ("Gold Standard") algorithm, and Sampson distance.

An algorithm is then described for automatically obtaining point correspondences, so that F may be estimated directly from an image pair. We discuss the estimation of F for special camera motions.

The chapter also covers a method of image rectification based on the computed F.

11.1 Basic equations

The fundamental matrix is defined by the equation

$$\mathbf{x}'^{\mathsf{T}}\mathtt{F}\mathbf{x} = 0 \tag{11.1}$$

for any pair of matching points $\mathbf{x} \leftrightarrow \mathbf{x}'$ in two images. Given sufficiently many point matches $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$ (at least 7), equation (11.1) can be used to compute the unknown matrix F. In particular, writing $\mathbf{x} = (x, y, 1)^{\mathsf{T}}$ and $\mathbf{x}' = (x', y', 1)^{\mathsf{T}}$ each point match gives rise to one linear equation in the unknown entries of F. The coefficients of this equation are easily written in terms of the known coordinates $\mathbf{x}$ and $\mathbf{x}'$. Specifically, the equation corresponding to a pair of points $(x, y, 1)$ and $(x', y', 1)$ is

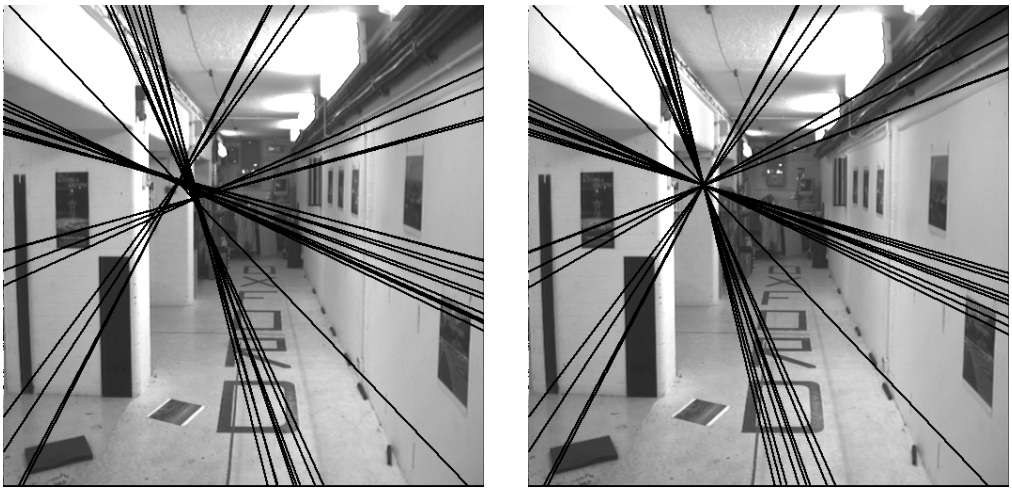
$$x'xf_{11} + x'yf_{12} + x'f_{13} + y'xf_{21} + y'yf_{22} + y'f_{23} + xf_{31} + yf_{32} + f_{33} = 0. \tag{11.2}$$

Denote by $\mathbf{f}$ the 9-vector made up of the entries of F in row-major order. Then (11.2) can be expressed as a vector inner product

$$(x'x, x'y, x', y'x, y'y, y', x, y, 1)\,\mathbf{f} = 0.$$

From a set of n point matches, we obtain a set of linear equations of the form

$$\mathtt{A}\mathbf{f} = \begin{bmatrix} x'_1x_1 & x'_1y_1 & x'_1 & y'_1x_1 & y'_1y_1 & y'_1 & x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\ x'_nx_n & x'_ny_n & x'_n & y'_nx_n & y'_ny_n & y'_n & x_n & y_n & 1 \end{bmatrix} \mathbf{f} = \mathbf{0}. \tag{11.3}$$



a b

Fig. 11.1. **Epipolar lines.** *(a) the effect of a non-singular fundamental matrix. Epipolar lines computed as* $\mathbf{l}' = \mathtt{F}\mathbf{x}$ *for varying* $\mathbf{x}$ *do not meet in a common epipole. (b) the effect of enforcing singularity using the SVD method described here.*

This is a homogeneous set of equations, and $\mathbf{f}$ can only be determined up to scale. For a solution to exist, matrix $\mathtt{A}$ must have rank at most 8, and if the rank is exactly 8, then the solution is unique (up to scale), and can be found by linear methods – the solution is the generator of the right null-space of $\mathtt{A}$.

If the data is not exact, because of noise in the point coordinates, then the rank of $\mathtt{A}$ may be greater than 8 (in fact equal to 9, since $\mathtt{A}$ has 9 columns). In this case, one finds a least-squares solution. Apart from the specific form of the equations (compare (11.3) with (4.3–*p*89)) the problem is essentially the same as the estimation problem considered in section 4.1.1(*p*90). Refer to the algorithm 4.1(*p*91). The least-squares solution for $\mathbf{f}$ is the singular vector corresponding to the smallest singular value of $\mathtt{A}$, that is, the last column of $\mathtt{V}$ in the SVD $\mathtt{A} = \mathtt{U}\mathtt{D}\mathtt{V}^{\mathsf{T}}$. The solution vector $\mathbf{f}$ found in this way minimizes $\|\mathtt{A}\mathbf{f}\|$ subject to the condition $\|\mathbf{f}\| = 1$. The algorithm just described is the essence of a method called the 8-point algorithm for computation of the fundamental matrix.

11.1.1 The singularity constraint

An important property of the fundamental matrix is that it is singular, in fact of rank 2. Furthermore, the left and right null-spaces of $\mathtt{F}$ are generated by the vectors representing (in homogeneous coordinates) the two epipoles in the two images. Most applications of the fundamental matrix rely on the fact that it has rank 2. For instance, if the fundamental matrix is not singular then computed epipolar lines are not coincident, as is demonstrated by figure 11.1. The matrix $\mathtt{F}$ found by solving the set of linear equations (11.3) will not in general have rank 2, and we should take steps to enforce this constraint. The most convenient way to do this is to correct the matrix $\mathtt{F}$ found by the SVD solution from $\mathtt{A}$. Matrix $\mathtt{F}$ is replaced by the matrix $\mathtt{F}'$ that minimizes the Frobenius norm $\|\mathtt{F} - \mathtt{F}'\|$ subject to the condition $\det \mathtt{F}' = 0$. A convenient method of

doing this is to again use the SVD. In particular, let $\mathtt{F} = \mathtt{U}\mathtt{D}\mathtt{V}^\mathsf{T}$ be the SVD of $\mathtt{F}$, where $\mathtt{D}$ is a diagonal matrix $\mathtt{D} = \mathrm{diag}(r, s, t)$ satisfying $r \geq s \geq t$. Then $\mathtt{F}' = \mathtt{U}\mathrm{diag}(r, s, 0)\mathtt{V}^\mathsf{T}$ minimizes the Frobenius norm of $\mathtt{F} - \mathtt{F}'$.

Thus, the 8-point algorithm for computation of the fundamental matrix may be formulated as consisting of two steps, as follows.

(i) **Linear solution.** A solution $\mathtt{F}$ is obtained from the vector $\mathbf{f}$ corresponding to the smallest singular value of $\mathtt{A}$, where $\mathtt{A}$ is defined in (11.3).

(ii) **Constraint enforcement.** Replace $\mathtt{F}$ by $\mathtt{F}'$, the closest singular matrix to $\mathtt{F}$ under a Frobenius norm. This correction is done using the SVD.

The algorithm thus stated is extremely simple, and readily implemented, assuming that appropriate linear algebra routines are available. As usual normalization is required, and we return to this in section 11.2.

11.1.2 The minimum case – seven point correspondences

The equation $\mathbf{x}_i'^\mathsf{T}\mathtt{F}\mathbf{x}_i = 0$ gives rise to a set of equations of the form $\mathtt{A}\mathbf{f} = \mathbf{0}$. If $\mathtt{A}$ has rank 8, then it is possible to solve for $\mathbf{f}$ up to scale. In the case where the matrix $\mathtt{A}$ has rank seven, it is still possible to solve for the fundamental matrix by making use of the singularity constraint. The most important case is when only 7 point correspondences are known (other cases are discussed in section 11.9). This leads to a 7×9 matrix $\mathtt{A}$, which generally will have rank 7.

The solution to the equations $\mathtt{A}\mathbf{f} = \mathbf{0}$ in this case is a 2-dimensional space of the form $\alpha\mathtt{F}_1 + (1-\alpha)\mathtt{F}_2$, where α is a scalar variable. The matrices $\mathtt{F}_1$ and $\mathtt{F}_2$ are obtained as the matrices corresponding to the generators $\mathbf{f}_1$ and $\mathbf{f}_2$ of the right null-space of $\mathtt{A}$. Now, we use the constraint that $\det \mathtt{F} = 0$. This may be written as $\det(\alpha\mathtt{F}_1 + (1-\alpha)\mathtt{F}_2) = 0$. Since $\mathtt{F}_1$ and $\mathtt{F}_2$ are known, this leads to a cubic polynomial equation in α. This polynomial equation may be solved to find the value of α. There will be either one or three real solutions (the complex solutions are discarded [Hartley-94c]). Substituting back in the equation $\mathtt{F} = \alpha\mathtt{F}_1 + (1-\alpha)\mathtt{F}_2$ gives one or three possible solutions for the fundamental matrix.

This method of computing one or three fundamental matrices for the minimum number of points (seven) is used in the robust algorithm of section 11.6. We return to the issue of the number of solutions in section 11.9.

11.2 The normalized 8-point algorithm

The 8-point algorithm is the simplest method of computing the fundamental matrix, involving no more than the construction and (least-squares) solution of a set of linear equations. If care is taken, then it can perform extremely well. The original algorithm is due to Longuet-Higgins [LonguetHiggins-81]. The key to success with the 8-point algorithm is proper careful normalization of the input data before constructing the equations to solve. The subject of normalization of input data has applications to many of the algorithms of this book, and is treated in general terms in section 4.4-(p104). In the case of the 8-point algorithm, a simple transformation (translation and

Objective

Given $n \geq 8$ image point correspondences $\{\mathbf{x}_i \leftrightarrow \mathbf{x}'_i\}$, determine the fundamental matrix $\mathtt{F}$ such that $\mathbf{x}'^{\mathsf{T}}_i \mathtt{F} \mathbf{x}_i = 0$.

Algorithm

(i) **Normalization:** Transform the image coordinates according to $\hat{\mathbf{x}}_i = \mathtt{T}\mathbf{x}_i$ and $\hat{\mathbf{x}}'_i = \mathtt{T}'\mathbf{x}'_i$, where $\mathtt{T}$ and $\mathtt{T}'$ are normalizing transformations consisting of a translation and scaling.

(ii) Find the fundamental matrix $\hat{\mathtt{F}}'$ corresponding to the matches $\hat{\mathbf{x}}_i \leftrightarrow \hat{\mathbf{x}}'_i$ by

  (a) **Linear solution:** Determine $\hat{\mathtt{F}}$ from the singular vector corresponding to the smallest singular value of $\hat{\mathtt{A}}$, where $\hat{\mathtt{A}}$ is composed from the matches $\hat{\mathbf{x}}_i \leftrightarrow \hat{\mathbf{x}}'_i$ as defined in (11.3).

  (b) **Constraint enforcement:** Replace $\hat{\mathtt{F}}$ by $\hat{\mathtt{F}}'$ such that $\det \hat{\mathtt{F}}' = 0$ using the SVD (see section 11.1.1).

(iii) **Denormalization:** Set $\mathtt{F} = \mathtt{T}'^{\mathsf{T}} \hat{\mathtt{F}}' \mathtt{T}$. Matrix $\mathtt{F}$ is the fundamental matrix corresponding to the original data $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$.

Algorithm 11.1. *The normalized 8-point algorithm for* $\mathtt{F}$.

scaling) of the points in the image before formulating the linear equations leads to an enormous improvement in the conditioning of the problem and hence in the stability of the result. The added complexity of the algorithm necessary to do this transformation is insignificant.

The suggested normalization is a translation and scaling of each image so that the centroid of the reference points is at the origin of the coordinates and the RMS distance of the points from the origin is equal to $\sqrt{2}$. This is carried out for essentially the same reasons as in chapter 4. The basic method is analogous to algorithm 4.2(p109) and is summarized in algorithm 11.1.

Note that it is recommended that the singularity condition should be enforced before denormalization. For a justification of this, refer to [Hartley-97c].

11.3 The algebraic minimization algorithm

The normalized 8-point algorithm includes a method for enforcing the singularity constraint on the fundamental matrix. The initial estimate $\mathtt{F}$ is replaced by the singular matrix $\mathtt{F}'$ that minimizes the difference $\|\mathtt{F}' - \mathtt{F}\|$. This is done using the SVD, and has the advantage of being simple and rapid.

Numerically, however, this method is not optimal, since all the entries of $\mathtt{F}$ do not have equal importance, and indeed some entries are more tightly constrained by the point-correspondence data than others. A more correct procedure would be to compute a covariance matrix from the entries of $\mathtt{F}$ in terms of the input data, and then to find the singular matrix $\mathtt{F}'$ closest to $\mathtt{F}$ in terms of Mahalanobis distance with respect to this covariance. Unfortunately, minimization of the Mahalanobis distance $\|\mathtt{F} - \mathtt{F}'\|_\Sigma$ cannot be done linearly for a general covariance matrix Σ, so this approach is unattractive.

An alternative procedure is to find the desired singular matrix $\mathtt{F}'$ directly. Thus, just as $\mathtt{F}$ is computed by minimizing the norm $\|\mathtt{A}\mathbf{f}\|$ subject to $\|\mathbf{f}\| = 1$, so one should aim

to find the *singular* matrix $\mathtt{F}'$ that minimizes $\|\mathtt{A}\mathbf{f}'\|$ subject to $\|\mathbf{f}'\| = 1$. It turns out not to be possible to do this by linear non-iterative means, chiefly because $\det \mathtt{F}' = 0$ is a cubic, rather than a linear constraint. Nevertheless, it will be seen that a simple iterative method is effective.

An arbitrary singular 3×3 matrix, such as the fundamental matrix $\mathtt{F}$, may be written as a product $\mathtt{F} = \mathtt{M}[\mathbf{e}]_\times$ where $\mathtt{M}$ is a non-singular matrix and $[\mathbf{e}]_\times$ is any skew-symmetric matrix, with $\mathbf{e}$ corresponding to the epipole in the first image.

Suppose we wish to compute the fundamental matrix $\mathtt{F}$ of the form $\mathtt{F} = \mathtt{M}[\mathbf{e}]_\times$ that minimizes the algebraic error $\|\mathtt{A}\mathbf{f}\|$ subject to the condition $\|\mathbf{f}\| = 1$. Let us assume for now that the epipole $\mathbf{e}$ is known. Later we will let $\mathbf{e}$ vary, but for now it is fixed. The equation $\mathtt{F} = \mathtt{M}[\mathbf{e}]_\times$ can be written in terms of the vectors $\mathbf{f}$ and $\mathbf{m}$ comprising the entries of $\mathtt{F}$ and $\mathtt{M}$ as an equation $\mathbf{f} = \mathtt{E}\mathbf{m}$ where $\mathtt{E}$ is a 9×9 matrix. Supposing that $\mathbf{f}$ and $\mathbf{m}$ contain the entries of the corresponding matrices in row-major order, then it can be verified that $\mathtt{E}$ has the form

$$\mathtt{E} = \begin{bmatrix} [\mathbf{e}]_\times & & \\ & [\mathbf{e}]_\times & \\ & & [\mathbf{e}]_\times \end{bmatrix}. \tag{11.4}$$

Since $\mathbf{f} = \mathtt{E}\mathbf{m}$, the minimization problem becomes: [1]

$$\text{Minimize } \|\mathtt{A}\mathtt{E}\mathbf{m}\| \text{ subject to the condition } \|\mathtt{E}\mathbf{m}\| = 1. \tag{11.5}$$

This minimization problem is solved using algorithm A5.6(*p*595). For the purposes of this algorithm one observes that $\text{rank}(\mathtt{E}) = 6$, since each of its diagonal blocks has rank 2.

11.3.1 Iterative estimation

The minimization (11.5) gives a way of computing an algebraic error vector $\mathtt{A}\mathbf{f}$ given a value for the epipole $\mathbf{e}$. This mapping $\mathbf{e} \mapsto \mathtt{A}\mathbf{f}$ is a map from $\mathbb{R}^3$ to $\mathbb{R}^9$. Note that the value of $\mathtt{A}\mathbf{f}$ is unaffected by scaling $\mathbf{e}$. Starting from an estimated value of $\mathbf{e}$ derived as the generator of the right null-space of an initial estimate of $\mathtt{F}$, one may iterate to find the final $\mathtt{F}$ that minimizes algebraic error. The initial estimate of $\mathtt{F}$ may be obtained from the 8-point algorithm, or any other simple algorithm. The complete algorithm for computation of $\mathtt{F}$ is given in algorithm 11.2.

Note the advantage of this method of computing $\mathtt{F}$ is that the iterative part of the algorithm consists of a very small parameter minimization problem, involving the estimation of only three parameters (the homogeneous coordinates of $\mathbf{e}$). Despite this, the algorithm finds the fundamental matrix that minimizes the algebraic error for all matched points. The matched points themselves do not come into the final iterative estimation.

[1] It does not do to minimize $\|\mathtt{A}\mathtt{E}\mathbf{m}\|$ subject to the condition $\|\mathbf{m}\| = 1$, since a solution to this occurs when $\mathbf{m}$ is a unit vector in the right null-space of $\mathtt{E}$. In this case, $\mathtt{E}\mathbf{m} = \mathbf{0}$, and hence $\|\mathtt{A}\mathtt{E}\mathbf{m}\| = 0$.

Objective

Find the fundamental matrix $\mathtt{F}$ that minimizes the algebraic error $\|\mathbf{A}\mathbf{f}\|$ subject to $\|\mathbf{f}\| = 1$ and $\det \mathtt{F} = 0$.

Algorithm

(i) Find a first approximation $\mathtt{F}_0$ for the fundamental matrix using the normalized 8-point algorithm 11.1. Then find the right null-vector $\mathbf{e}_0$ of $\mathtt{F}_0$.

(ii) Starting with the estimate $\mathbf{e}_i = \mathbf{e}_0$ for the epipole, compute the matrix $\mathtt{E}_i$ according to (11.4), then find the vector $\mathbf{f}_i = \mathtt{E}_i\mathbf{m}_i$ that minimizes $\|\mathbf{A}\mathbf{f}_i\|$ subject to $\|\mathbf{f}_i\| = 1$. This is done using algorithm A5.6(*p*595).

(iii) Compute the algebraic error $\boldsymbol{\epsilon}_i = \mathbf{A}\mathbf{f}_i$. Since $\mathbf{f}_i$ and hence $\boldsymbol{\epsilon}_i$ is defined only up to sign, correct the sign of $\boldsymbol{\epsilon}_i$ (multiplying by minus 1 if necessary) so that $\mathbf{e}_i^{\mathsf{T}}\mathbf{e}_{i-1} > 0$ for $i > 0$. This is done to ensure that $\boldsymbol{\epsilon}_i$ varies smoothly as a function of $\mathbf{e}_i$.

(iv) The previous two steps define a mapping $\mathbb{R}^3 \rightarrow \mathbb{R}^9$ mapping $\mathbf{e}_i \mapsto \boldsymbol{\epsilon}_i$. Now use the Levenberg–Marquardt algorithm (section A6.2(*p*600)) to vary $\mathbf{e}_i$ iteratively so as to minimize $\|\boldsymbol{\epsilon}_i\|$.

(v) Upon convergence, $\mathbf{f}_i$ represents the desired fundamental matrix.

Algorithm 11.2. *Computation of* $\mathtt{F}$ *with* $\det \mathtt{F} = 0$ *by iteratively minimizing algebraic error.*

11.4 Geometric distance

This section describes three algorithms which minimize a geometric image distance. The one we recommend, which is the Gold Standard method, unfortunately requires the most effort in implementation. The other algorithms produce extremely good results and are easier to implement, but are not optimal under the assumption that the image errors are Gaussian. Two important issues for each of the algorithms are the intitialization for the non-linear minimization, and the parametrization of the cost function. The algorithms are generally initialized by one of the linear algorithms of the previous section. An alternative, which is used in the automatic algorithm, is to select 7 correspondences and thus generate one or three solutions for $\mathtt{F}$. Various parametrizations are discussed in section 11.4.2. In all cases we recommend that the image points be normalized by a translation and scaling. This normalization does not skew the noise characteristics, so does not interfere with the optimality of the Gold Standard algorithm, which is described next.

11.4.1 The Gold Standard method

The Maximum Likelihood estimate of the fundamental matrix depends on the assumption of an error model. We make the assumption that noise in image point measurements obeys a Gaussian distribution. In that case the ML estimate is the one that minimizes the geometric distance (which is reprojection error)

$$\sum_i d(\mathbf{x}_i, \hat{\mathbf{x}}_i)^2 + d(\mathbf{x}'_i, \hat{\mathbf{x}}'_i)^2 \tag{11.6}$$

where $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$ are the measured correspondences, and $\hat{\mathbf{x}}_i$ and $\hat{\mathbf{x}}'_i$ are estimated "true" correspondences that satisfy $\hat{\mathbf{x}}_i'^{\mathsf{T}}\mathtt{F}\hat{\mathbf{x}}_i = 0$ exactly for some rank-2 matrix $\mathtt{F}$, the estimated fundamental matrix.

Objective

Given $n \geq 8$ image point correspondences $\{\mathbf{x}_i \leftrightarrow \mathbf{x}'_i\}$, determine the Maximum Likelihood estimate $\hat{\mathtt{F}}$ of the fundamental matrix.

The MLE involves also solving for a set of subsidiary point correspondences $\{\hat{\mathbf{x}}_i \leftrightarrow \hat{\mathbf{x}}'_i\}$, such that $\hat{\mathbf{x}}_i'^{\mathsf{T}}\hat{\mathtt{F}}\hat{\mathbf{x}}_i = 0$, and which minimizes

$$\sum_i d(\mathbf{x}_i, \hat{\mathbf{x}}_i)^2 + d(\mathbf{x}'_i, \hat{\mathbf{x}}'_i)^2.$$

Algorithm

(i) Compute an initial rank 2 estimate of $\hat{\mathtt{F}}$ using a linear algorithm such as algorithm 11.1.

(ii) Compute an initial estimate of the subsidiary variables $\{\hat{\mathbf{x}}_i, \hat{\mathbf{x}}'_i\}$ as follows:

  (a) Choose camera matrices $\mathtt{P} = [\mathtt{I} \mid \mathbf{0}]$ and $\mathtt{P}' = [[\mathbf{e}']_\times \hat{\mathtt{F}} \mid \mathbf{e}']$, where $\mathbf{e}'$ is obtained from $\hat{\mathtt{F}}$.

  (b) From the correspondence $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$ and $\hat{\mathtt{F}}$ determine an estimate of $\widehat{\mathbf{X}}_i$ using the triangulation method of chapter 12.

  (c) The correspondence consistent with $\hat{\mathtt{F}}$ is obtained as $\hat{\mathbf{x}}_i = \mathtt{P}\widehat{\mathbf{X}}_i,\ \hat{\mathbf{x}}'_i = \mathtt{P}'\widehat{\mathbf{X}}_i$.

(iii) Minimize the cost

$$\sum_i d(\mathbf{x}_i, \hat{\mathbf{x}}_i)^2 + d(\mathbf{x}'_i, \hat{\mathbf{x}}'_i)^2$$

over $\hat{\mathtt{F}}$ and $\widehat{\mathbf{X}}_i,\ i = 1, \ldots, n$. The cost is minimized using the Levenberg–Marquardt algorithm over $3n + 12$ variables: $3n$ for the n 3D points $\widehat{\mathbf{X}}_i$, and 12 for the camera matrix $\mathtt{P}' = [\mathtt{M} \mid \mathbf{t}]$, with $\hat{\mathtt{F}} = [\mathbf{t}]_\times \mathtt{M}$, and $\hat{\mathbf{x}}_i = \mathtt{P}\widehat{\mathbf{X}}_i,\ \hat{\mathbf{x}}'_i = \mathtt{P}'\widehat{\mathbf{X}}_i$.

Algorithm 11.3. *The Gold Standard algorithm for estimating* F *from image correspondences.*

This error function may be minimized in the following manner. A pair of camera matrices $\mathtt{P} = [\mathtt{I} \mid \mathbf{0}]$ and $\mathtt{P}' = [\mathtt{M} \mid \mathbf{t}]$ are defined. In addition one defines 3D points $\mathbf{X}_i$. Now letting $\hat{\mathbf{x}}_i = \mathtt{P}\mathbf{X}_i$ and $\hat{\mathbf{x}}'_i = \mathtt{P}'\mathbf{X}_i$, one varies $\mathtt{P}'$ and the points $\mathbf{X}_i$ so as to minimize the error expression. Subsequently F is computed as $\mathtt{F} = [\mathbf{t}]_\times \mathtt{M}$. The vectors $\hat{\mathbf{x}}_i$ and $\hat{\mathbf{x}}'_i$ will satisfy $\hat{\mathbf{x}}_i'^{\mathsf{T}}\mathtt{F}\hat{\mathbf{x}}_i = 0$. Minimization of the error is carried out using the Levenberg–Marquardt algorithm described in section A6.2(*p*600). An initial estimate of the parameters is computed using the normalized 8-point algorithm, followed by projective reconstruction, as described in chapter 12. Thus, estimation of the fundamental matrix using this method is effectively equivalent to projective reconstruction. The steps of the algorithm are summarized in algorithm 11.3.

It may seem that this method for computing F will be expensive in computing cost. However, the use of the sparse LM techniques means that it is not much more expensive than other iterative techniques, and details of this are given in section A6.5(*p*609).

11.4.2 Parametrization of rank-2 matrices

The non-linear minimization of the geometric distance cost functions requires a parametrization of the fundamental matrix which enforces the rank 2 property of the matrix. We describe three such parametrizations.

Over-parametrization. One way that we have already seen for parametrizing $\mathtt{F}$ is to write $\mathtt{F} = [\mathbf{t}]_\times \mathtt{M}$, where $\mathtt{M}$ is an arbitrary 3×3 matrix. This ensures that $\mathtt{F}$ is singular, since $[\mathbf{t}]_\times$ is. This way, $\mathtt{F}$ is parametrized by the nine entries of $\mathtt{M}$ and the three entries of $\mathbf{t}$ – a total of 12 parameters, more than the minimum number of parameters, which is 7. In general this should not cause a significant problem.

Epipolar parametrization. An alternative way of parametrizing $\mathtt{F}$ is by specifying the first two columns of $\mathtt{F}$, along with two multipliers α and β such that the third column may be written as a linear combination $\mathbf{f}_3 = \alpha\mathbf{f}_1 + \beta\mathbf{f}_2$. Thus, the fundamental matrix is parametrized as

$$\mathtt{F} = \begin{bmatrix} a & b & \alpha a + \beta b \\ c & d & \alpha c + \beta d \\ e & f & \alpha e + \beta f \end{bmatrix}. \tag{11.7}$$

This has a total of 8 parameters. To achieve a minimum set of parameters, one of the elements, for instance f, may be set to 1. In practice whichever of $a, \ldots, f$ has greatest absolute value is set to 1. This method ensures a singular matrix $\mathtt{F}$, while using the minimum number of parameters. The main disadvantage is that it has a singularity – it does not work when the first two columns of $\mathtt{F}$ are linearly dependent, for then it is not possible to write column 3 in terms of the first two columns. This problem can be significant, since it will occur in the case where the right epipole lies at infinity. For then $\mathtt{F}\mathbf{e} = \mathtt{F}(e_1, e_2, 0)^\mathsf{T} = \mathbf{0}$ and the first two columns are linearly dependent. Nevertheless, this parametrization is widely used and works well if steps are taken to avoid this singularity. Instead of using the first two columns as a basis, another pair of columns can be used, in which case the singularity occurs when the epipole is on one of the coordinate axes. In practice such singularities can be detected during the minimization and the parametrization switched to one of the alternative parametrizations.

Note that $(\alpha, \beta, -1)^\mathsf{T}$ is the right epipole for this fundamental matrix – the coordinates of the epipole occur explicitly in the parametrization. For best results, the parametrization should be chosen so that the largest entry (in absolute value) of the epipole is the one set to 1.

Note how the complete manifold of possible fundamental matrices is not covered by a single parametrization, but rather by a set of minimally parametrized patches. As a path is traced out through the manifold during a parameter minimization procedure, it is necessary to switch from one patch to another as the boundary between patches is crossed. In this case there are actually 18 different parameter patches, depending on which of $a, \ldots, f$ is greatest, and which pair of columns are taken as the basis.

Both epipoles as parameters. The previous parametrization uses one of the epipoles as part of the parametrization. For symmetry one may use both the epipoles as parameters. The resulting form of $\mathtt{F}$ is

$$\mathtt{F} = \begin{bmatrix} a & b & \alpha a + \beta b \\ c & d & \alpha c + \beta d \\ \alpha' a + \beta' c & \alpha' b + \beta' d & \alpha'\alpha a + \alpha'\beta b + \beta'\alpha c + \beta'\beta d \end{bmatrix}. \tag{11.8}$$

The two epipoles are $(\alpha, \beta, -1)^\mathsf{T}$ and $(\alpha', \beta', -1)^\mathsf{T}$. As above, one can set one of a, b, c, d to 1. To avoid singularities, one must switch between different choices of the two rows and two columns to use as the basis. Along with four choices of which of a, b, c, d to set to 1, there are a total of 36 parameter patches used to cover the complete manifold of fundamental matrices.

11.4.3 First-order geometric error (Sampson distance)

The concept of Sampson distance was discussed at length in section 4.2.6(*p*98). Here the Sampson approximation is used in the case of the variety defined by $\mathbf{x}'^\mathsf{T}\mathtt{F}\mathbf{x} = 0$ to provide a first-order approximation to the geometric error.

The general formula for the Sampson cost function is given in (4.13–*p*100). In the case of fundamental matrix estimation, the formula is even simpler, since there is only one equation per point correspondence (see also example 4.2(*p*100)). The partial-derivative matrix $\mathtt{J}$ has only one row, and hence $\mathtt{J}\mathtt{J}^\mathsf{T}$ is a scalar and (4.12–*p*99) becomes

$$\frac{\boldsymbol{\epsilon}^\mathsf{T}\boldsymbol{\epsilon}}{\mathtt{J}\mathtt{J}^\mathsf{T}} = \frac{(\mathbf{x}_i'^\mathsf{T}\mathtt{F}\mathbf{x}_i)^2}{\mathtt{J}\mathtt{J}^\mathsf{T}}.$$

From the definition of $\mathtt{J}$ and the explicit form of $\mathtt{A}_i = \mathbf{x}_i'^\mathsf{T}\mathtt{F}\mathbf{x}_i$ given in the left hand side of (11.2), we obtain

$$\mathtt{J}\mathtt{J}^\mathsf{T} = (\mathtt{F}\mathbf{x}_i)_1^2 + (\mathtt{F}\mathbf{x}_i)_2^2 + (\mathtt{F}^\mathsf{T}\mathbf{x}_i')_1^2 + (\mathtt{F}^\mathsf{T}\mathbf{x}_i')_2^2$$

where for instance $(\mathtt{F}\mathbf{x}_i)_j^2$ represents the square of the j-th entry of the vector $\mathtt{F}\mathbf{x}_i$. Thus, the cost function is

$$\sum_i \frac{(\mathbf{x}_i'^\mathsf{T}\mathtt{F}\mathbf{x}_i)^2}{(\mathtt{F}\mathbf{x}_i)_1^2 + (\mathtt{F}\mathbf{x}_i)_2^2 + (\mathtt{F}^\mathsf{T}\mathbf{x}_i')_1^2 + (\mathtt{F}^\mathsf{T}\mathbf{x}_i')_2^2}. \tag{11.9}$$

This gives a first-order approximation to geometric error, which may be expected to give good results if higher order terms are small in comparison to the first. The approximation has been used successfully in estimation algorithms by [Torr-97, Torr-98, Zhang-98]. Note that this approximation is undefined at the point in $\mathbb{R}^4$ determined by the two epipoles, as here $\mathtt{J}\mathtt{J}^\mathsf{T}$ is zero. This point should be avoided in any numerical implementation.

The key advantage of approximating the geometric error in this way is that the resulting cost function only involves the parameters of $\mathtt{F}$. This means that to first-order the Gold Standard cost function (11.6) is minimized *without* introducing a set of subsidiary variables, namely the coordinates of the n space points $\mathbf{X}_i$. Consequently a minimization problem with $7 + 3n$ degrees of freedom is reduced to one with only 7 degrees of freedom.

Symmetric epipolar distance. Equation (11.9) is similar in form to another cost function

$$\sum_i \quad d(\mathbf{x}_i', \mathtt{F}\mathbf{x}_i)^2 + d(\mathbf{x}_i, \mathtt{F}^\mathsf{T}\mathbf{x}_i')^2$$

$$= \sum_i (\mathbf{x}_i'^{\mathsf{T}} \mathtt{F} \mathbf{x}_i)^2 \left(\frac{1}{(\mathtt{F}\mathbf{x}_i)_1^2 + (\mathtt{F}\mathbf{x}_i)_2^2} + \frac{1}{(\mathtt{F}^{\mathsf{T}}\mathbf{x}_i')_1^2 + (\mathtt{F}^{\mathsf{T}}\mathbf{x}_i')_2^2} \right) \quad (11.10)$$

which minimizes the distance of a point from its projected epipolar line, computed in each of the images. However, this cost function seems to give slightly inferior results to (11.9) (see [Zhang-98]), and hence is not discussed further.

11.5 Experimental evaluation of the algorithms

Three of the algorithms of the previous sections are now compared by estimating F from point correspondences for a number of image pairs. The algorithms are:

(i) The normalized 8-point algorithm (algorithm 11.1).
(ii) Minimization of algebraic error whilst imposing the singularity constraint (algorithm 11.2).
(iii) The Gold Standard geometric algorithm (algorithm 11.3).

The experimental procedure was as follows. For each pair of images, a number n of matched points were chosen randomly from the matches and the fundamental matrix estimated and residual error (see below) computed. This experiment was repeated 100 times for each value of n and each pair of images, and the average residual error plotted against n. This gives an idea of how the different algorithms behave as the number of points is increased. The number of points used, n, ranged from 8 up to three-quarters of the total number of matched points.

Residual error

The error is defined as

$$\frac{1}{N} \sum_i^N d(\mathbf{x}_i', \mathtt{F}\mathbf{x}_i)^2 + d(\mathbf{x}_i, \mathtt{F}^{\mathsf{T}}\mathbf{x}_i')^2$$

where $d(\mathbf{x}, \mathbf{l})$ here is the distance (in pixels) between a point $\mathbf{x}$ and a line $\mathbf{l}$. The error is the squared distance between a point's epipolar line and the matching point in the other image (computed for both points of the match), averaged over all N matches. Note the error is evaluated over *all* N matched points, and not just the n matches used to compute F. The residual error corresponds to the epipolar distance defined in (11.10). Note that this particular error is *not* minimized directly by any of the algorithms evaluated here.

The various algorithms were tried with 5 different pairs of images. The images are presented in figure 11.2 and show the diversity of image types, and placement of the epipoles. A few of the epipolar lines are shown in the images. The intersection of the pencil of lines is the epipole. There was a wide variation in the accuracy of the matched points for the different images, though mismatches were removed in a pre-processing step.

Results. The results of these experiments are shown and explained in figure 11.3. They show that minimizing algebraic error gives essentially indistinguishable results from minimizing geometric error.

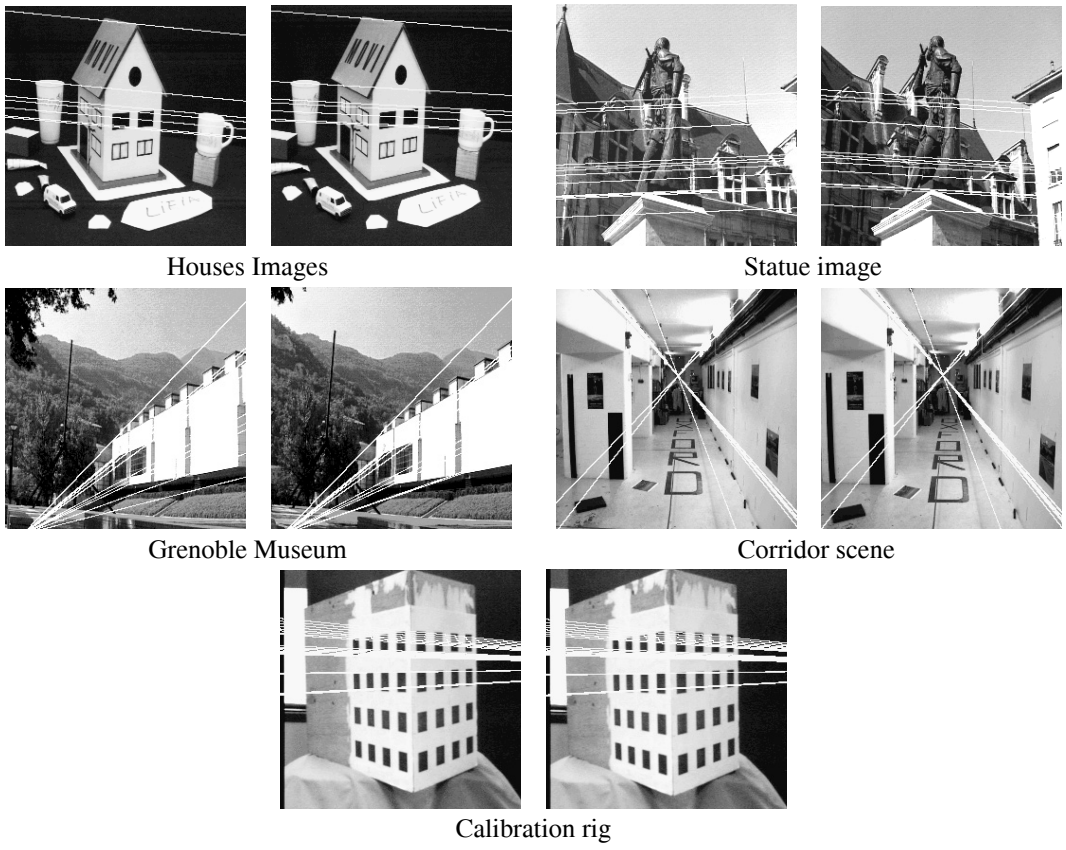


Fig. 11.2. *Image pairs used for the algorithm comparison. In the top two the epipoles are far from the image centres. In the middle two the epipoles are close (Grenoble) and in the image (Corridor). For the calibration images the matched points are known extremely accurately.*

11.5.1 Recommendations

Several methods of computing the fundamental matrix have been discussed in this chapter, and some pointers on which method to use are perhaps desirable. Briefly, these are our recommendations:

- Do not use the unnormalized 8-point algorithm.
- For a quick method, easy to implement, use the normalized 8-point algorithm 11.1. This often gives adequate results, and is ideal as a first step in other algorithms.
- If more accuracy is desired, use the algebraic minimization method, either with or without iteration on the position of the epipole.
- As an alternative that gives excellent results, use an iterative-minimization method that minimizes the Sampson cost function (11.9). This and the iterative algebraic method give similar results.
- To be certain of getting the best results, if Gaussian noise is a viable assumption, implement the Gold Standard algorithm.

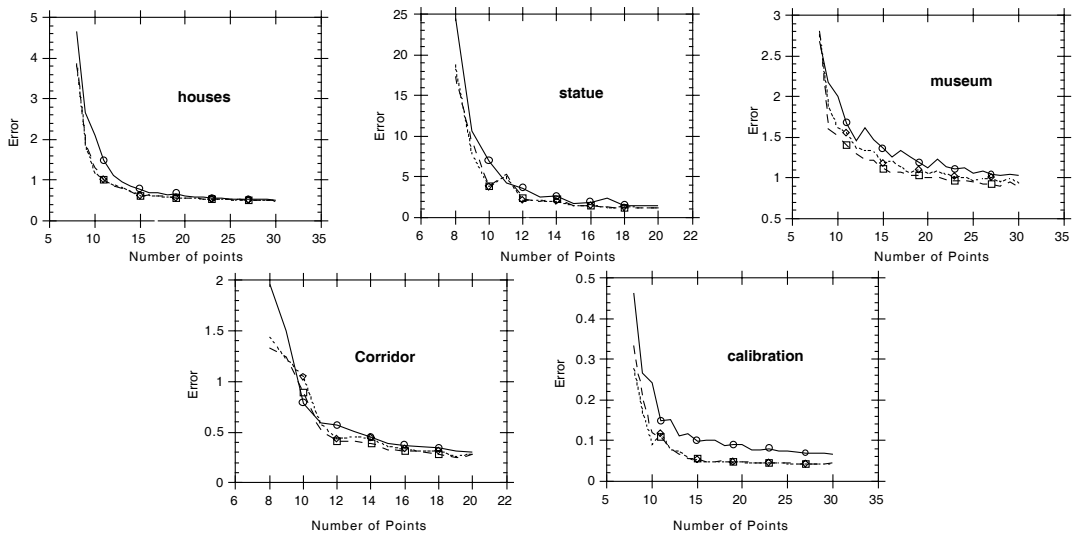


Fig. 11.3. **Results of the experimental evaluation of the algorithms.** *In each case, three methods of computing* F *are compared. Residual error is plotted against the number of points used to compute* F*. In each graph, the top (solid line) shows the results of the normalized 8-point algorithm. Also shown are the results of minimizing geometric error (long dashed line) and iteratively minimizing algebraic error subject to the determinant constraint (short dashed line). In most cases, the result of iteratively minimizing algebraic error is almost indistinguishable from minimizing geometric error. Both are noticeably better than the non-iterative normalized 8-point algorithm, though that algorithm also gives good results.*

11.6 Automatic computation of F

This section describes an algorithm to compute the epipolar geometry between two images automatically. The input to the algorithm is simply the pair of images, with no other *a priori* information required; and the output is the estimated fundamental matrix together with a set of interest points in correspondence.

The algorithm uses RANSAC as a search engine in a similar manner to its use in the automatic computation of a homography described in section 4.8(p123). The ideas and details of the algorithm are given there, and are not repeated here. The method is summarized in algorithm 11.4, with an example of its use shown in figure 11.4.

A few remarks on the method:

(i) **The RANSAC sample.** Only 7 point correspondences are used to estimate F. This has the advantage that a rank 2 matrix is produced, and it is not necessary to coerce the matrix to rank 2 as in the linear algorithms. A second reason for using 7 correspondences, rather than 8 say with a linear algorithm, is that the number of samples that must be tried in order to ensure a high probability of no outliers is exponential in the size of the sample set. For example, from table 4.3-(p119) for a 99% confidence of no outliers (when drawing from a set containing 50% outliers) twice as many samples are required for 8 correspondences as for 7. The slight disadvantage in using 7 correspondences is that it may result in 3 real solutions for F, and all 3 must be tested for support.

Objective Compute the fundamental matrix between two images.

Algorithm

(i) **Interest points:** Compute interest points in each image.
(ii) **Putative correspondences:** Compute a set of interest point matches based on proximity and similarity of their intensity neighbourhood.
(iii) **RANSAC robust estimation:** Repeat for N samples, where N is determined adaptively as in algorithm 4.5(*p*121):
  (a) Select a random sample of 7 correspondences and compute the fundamental matrix F as described in section 11.1.2. There will be one or three real solutions.
  (b) Calculate the distance $d_\perp$ for each putative correspondence.
  (c) Compute the number of inliers consistent with F by the number of correspondences for which $d_\perp < t$ pixels.
  (d) If there are three real solutions for F the number of inliers is computed for each solution, and the solution with most inliers retained.

  Choose the F with the largest number of inliers. In the case of ties choose the solution that has the lowest standard deviation of inliers.
(iv) **Non-linear estimation:** re-estimate F from all correspondences classified as inliers by minimizing a cost function, e.g. (11.6), using the Levenberg–Marquardt algorithm of section A6.2(*p*600).
(v) **Guided matching:** Further interest point correspondences are now determined using the estimated F to define a search strip about the epipolar line.

The last two steps can be iterated until the number of correspondences is stable.

Algorithm 11.4. *Algorithm to automatically estimate the fundamental matrix between two images using RANSAC.*

(ii) **The distance measure.** Given a current estimate of F (from the RANSAC sample) the distance $d_\perp$ measures how closely a matched pair of points satisfies the epipolar geometry. There are two clear choices for $d_\perp$: reprojection error, i.e. the distance minimized in the cost function (11.6) (the value may be obtained using the triangulation algorithm of section 12.5); or the Sampson approximation to reprojection error ($d_\perp^2$ is given by (11.9)). If the Sampson approximation is used, then the Sampson cost function should be used to iteratively estimate F. Otherwise distances used in RANSAC and elsewhere in the algorithm will be inconsistent.

(iii) **Guided matching.** The current estimate of F defines a search band in the second image around the epipolar line $\mathbf{Fx}$ of $\mathbf{x}$. For each corner $\mathbf{x}$ a match is sought within this band. Since the search area is restricted a weaker similarity threshold can be employed, and it is not necessary to enforce a "winner takes all" scheme.

(iv) **Implementation and run details.** For the example of figure 11.4, the search window was ± 300 pixels. The inlier threshold was $t = 1.25$ pixels. A total of 407 samples were required. The RMS pixel error after RANSAC was 0.34 (for 99 correspondences), and after MLE and guided matching it was 0.33 (for 157 correspondences). The guided matching MLE required 10 iterations of the Levenberg–Marquardt algorithm.

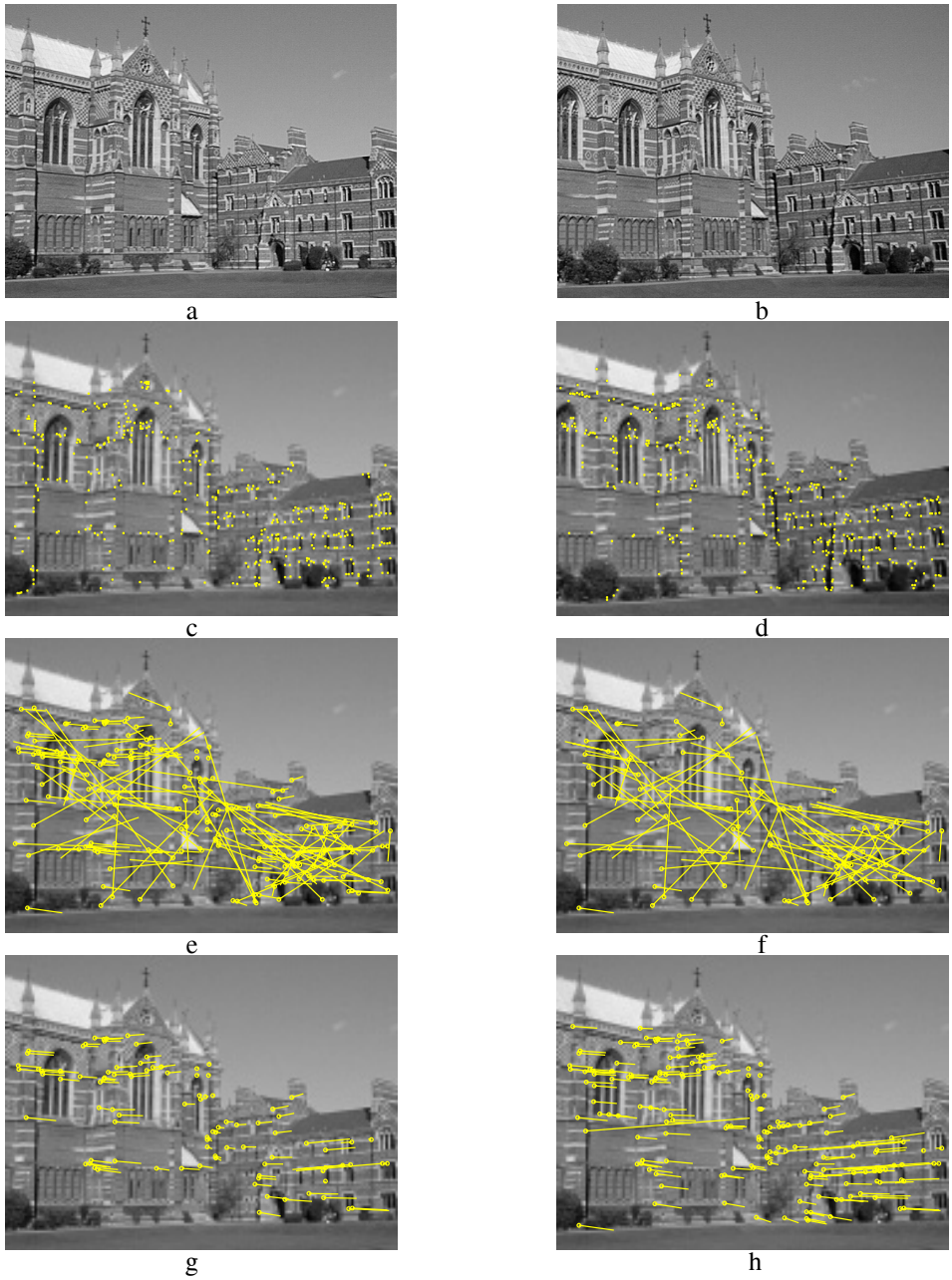


Fig. 11.4. **Automatic computation of the fundamental matrix between two images using RANSAC.** *(a) (b) left and right images of Keble College, Oxford. The motion between views is a translation and rotation. The images are* 640×480 *pixels. (c) (d) detected corners superimposed on the images. There are approximately 500 corners on each image. The following results are superimposed on the left image: (e) 188 putative matches shown by the line linking corners, note the clear mismatches; (f) outliers – 89 of the putative matches. (g) inliers – 99 correspondences consistent with the estimated* F*; (h) final set of 157 correspondences after guided matching and MLE. There are still a few mismatches evident, e.g. the long line on the left.*

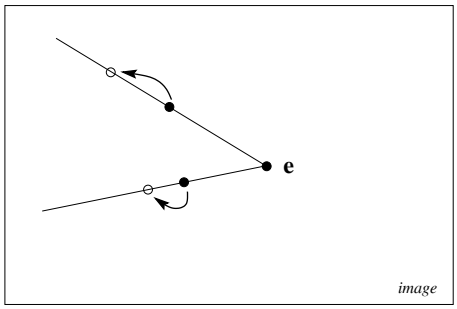


Fig. 11.5. *For a pure translation the epipole can be estimated from the image motion of two points.*

11.7 Special cases of F-computation

Certain special cases of motion, or partially known camera calibration, allow computation of the fundamental matrix to be simplified. In each case the number of degrees of freedom of the fundamental matrix is less than the 7 of general motion. We give three examples.

11.7.1 Pure translational motion

This is the simplest possible case. The matrix can be estimated linearly whilst simultaneously imposing the constraints that the matrix must satisfy, namely that it is skew-symmetric (see section 9.3.1(*p*247)), and thus has the required rank of 2. In this case $\mathtt{F} = [\mathbf{e}']_\times$, and has two degrees of freedom. It may be parametrized by the three entries of $\mathbf{e}'$.

Each point correspondence provides one linear constraint on the homogeneous parameters, as is clear from figure 11.5. The matrix can be computed uniquely from two point correspondences.

Note, in the general motion case if all 3D points are coplanar, which is a structure degeneracy (see section 11.9), the fundamental matrix cannot be determined uniquely from image correspondences. However, for pure translational motion this is not a problem (two 3D points are always coplanar). The only degeneracy is if the two 3D points are coplanar with both camera centres.

This special form also simplifies the Gold Standard estimation, and correspondingly triangulation for structure recovery. The Gold Standard estimation of the epipole from point correspondences under pure translation is identical to the estimation of a vanishing point given the end points of a set of imaged parallel lines, see section 8.6.1(*p*213).

11.7.2 Planar motion

In the case of planar motion, described in section 9.3.2(*p*250), we require that the symmetric part of F has rank 2, in addition to the standard rank 2 condition for the full matrix. It can be verified that the parametrization of (9.8–*p*252), namely $\mathtt{F} = [\mathbf{e}']_\times [\mathbf{l}_s]_\times [\mathbf{e}]_\times$, satisfies both these conditions. If unconstrained 3-vectors are used to represent $\mathbf{e}', \mathbf{l}_s$ and $\mathbf{e}$ then 9 parameters are used, whereas the fundamental matrix for planar motion has only 6 degrees of freedom. As usual this over-parametrization is not a problem.

An alternative parametrization with similar properties is

$$\mathtt{F} = \alpha[\mathbf{x_a}]_\times + \beta\left(\mathbf{l}_s\mathbf{l}_h^\mathsf{T} + \mathbf{l}_h\mathbf{l}_s^\mathsf{T}\right) \quad \text{with} \quad \mathbf{x}_\mathtt{a}^\mathsf{T}\mathbf{l}_h = 0$$

where α and β are scalars, and the meaning of the 3-vectors $\mathbf{x_a}, \mathbf{l}_s$ and $\mathbf{l}_h$ is evident from figure 9.11(*p*253)(a).

11.7.3 The calibrated case

In the case of calibrated cameras normalized image coordinates may be used, and the essential matrix E computed instead of the fundamental matrix. As with the fundamental matrix, the essential matrix may be computed using linear techniques from 8 points or more, since corresponding points satisfy the defining equation $\mathbf{x}_i'^\mathsf{T}\mathtt{E}\mathbf{x}_i = 0$.

Where the method differs from the computation of the fundamental matrix is in the enforcement of the constraints. For, whereas the fundamental matrix satisfies $\det \mathtt{F} = 0$, the essential matrix satisfies the additional condition that its two singular values are equal. This constraint may be handled by the following result, which is offered here without proof.

Result 11.1. *Let* E *be a* 3×3 *matrix with SVD given by* $\mathtt{E} = \mathtt{U}\mathtt{D}\mathtt{V}^\mathsf{T}$*, where* $\mathtt{D} = \mathrm{diag}(a, b, c)$ *with* $a \geq b \geq c$*. Then the closest essential matrix to* E *in Frobenius norm is given by* $\hat{\mathtt{E}} = \mathtt{U}\hat{\mathtt{D}}\mathtt{V}^\mathsf{T}$*, where* $\hat{\mathtt{D}} = \mathrm{diag}((a+b)/2, (a+b)/2, 0)$.

If the goal is to compute the two normalized camera matrices P and P′ as part of a reconstruction process, then it is not actually necessary to compute $\hat{\mathtt{E}}$ by multiplying out $\hat{\mathtt{E}} = \mathtt{U}\hat{\mathtt{D}}\mathtt{V}^\mathsf{T}$. Matrix P′ can be computed directly from the SVD according to result 9.19-(*p*259). The choice between the four solutions for P′ is determined by the consideration that the visible points must lie in front of the two cameras, as explained in section 9.6.3-(*p*259).

11.8 Correspondence of other entities

So far in this chapter only point correspondences have been employed, and the question naturally arises: can F be computed from the correspondence of image entities other than points? The answer is yes, but not from all types of entities. We will now discuss some common examples.

Lines. The correspondence of image lines between views places *no* constraint at all on F. Here a line is an infinite line, not a line segment. Consider the case of corresponding image points: the points in each image back-project to rays, one through each camera centre, and these rays intersect at the 3-space point. Now in general two lines in 3-space are skew (i.e. they do not intersect); so the condition that the rays intersect places a constraint on the epipolar geometry. In contrast in the case of corresponding image lines, the back-projection is a plane from each view. However, two planes in 3-space always intersect so there is no constraint on the epipolar geometry (there is a constraint in the case of 3-views).

In the case of parallel lines, the correspondence of vanishing points does provide a

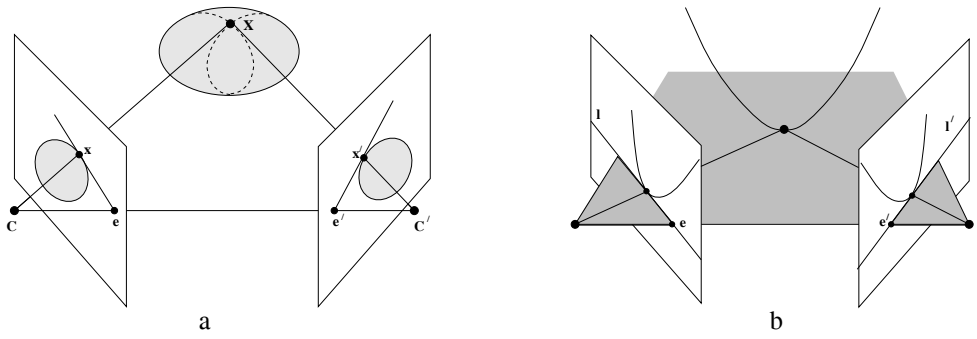


Fig. 11.6. **Epipolar tangency.** *(a) for a surface; (b) for a space curve – figure after Porrill and Pollard [Porrill-91]. In (a) the epipolar plane* $\mathbf{CC}'\mathbf{X}$ *is tangent to the surface at* $\mathbf{X}$*. The imaged outline is tangent to the epipolar lines at* $\mathbf{x}$ *and* $\mathbf{x}'$ *in the two views. The dashed curves on the surface are the contour generators. In (b) the epipolar plane is tangent to the space curve. The corresponding epipolar lines* $\mathbf{l} \leftrightarrow \mathbf{l}'$ *are tangent to the imaged curve.*

constraint on $\mathtt{F}$. However, a vanishing point has the same status as any finite point, i.e. it provides one constraint.

Space curves and surfaces. As illustrated in figure 11.6, at points at which the epipolar plane is tangent to a space curve the imaged curve is tangent to the corresponding epipolar lines. This provides a constraint on the 2 view geometry, i.e. if an epipolar line is tangent to an imaged curve in one view, then the corresponding epipolar line must be tangent to the imaged curve in the other view. Similarly, in the case of surfaces, at points at which the epipolar plane is tangent to the surface the imaged outline is tangent to the corresponding epipolar lines. Epipolar tangent points act effectively as point correspondences and may be included in estimation algorithms as described by [Porrill-91].

Particularly important cases are those of conics and quadrics which are algebraic objects and so algebraic solutions can be developed. Examples are given in the notes and exercises at the end of this chapter.

11.9 Degeneracies

A set of correspondences $\{\mathbf{x}_i \leftrightarrow \mathbf{x}'_i, i = 1, \ldots, n\}$ is geometrically degenerate with respect to $\mathtt{F}$ if it fails to uniquely define the epipolar geometry, or equivalently if there exist linearly independent rank-2 matrices, $\mathtt{F}_j$, $j = 1, 2$, such that

$$\mathbf{x}_i'^{\mathsf{T}}\mathtt{F}_1\mathbf{x}_i^{\mathsf{T}} = 0 \quad \text{and} \quad \mathbf{x}_i'^{\mathsf{T}}\mathtt{F}_2\mathbf{x}_i = 0 \qquad (1 \leq i \leq n) \ .$$

The subject of degeneracy is investigated in detail in chapter 22. However, a brief preview is given now for the two important cases of scene points on a ruled quadric, or on a plane.

Provided the two camera centres are not coincident the epipolar geometry *is* uniquely defined. It can always be computed from the camera matrices $\mathtt{P}, \mathtt{P}'$ as in (9.1–*p*244) for example. What is at issue here are configurations where the epipolar geometry cannot be estimated from point correspondences. An awareness of the degeneracies of

$\dim(N) = 1$: Unique solution – no degeneracy.

Arises from $n \geq 8$ point correspondences in general position. If $n > 8$ then the point correspondences must be perfect (i.e. noise-free).

$\dim(N) = 2$: 1 or 3 solutions.

Arises in the case of seven point correspondences, and also in the case of $n > 7$ perfect point correspondences where the 3D points and camera centres lie on a ruled quadric referred to as a critical surface. The quadric may be non-degenerate (a hyperboloid of one sheet) or degenerate.

$\dim(N) = 3$: Two-parameter family of solutions.

Arises if $n \geq 6$ perfect point correspondences are related by a homography, $\mathbf{x}'_i = \mathtt{H}\mathbf{x}_i$.

- Rotation about the camera centre (a degenerate motion).
- All world points on a plane (a degenerate structure).

Table 11.1. *Degeneracies in estimating* F *from point correspondences, classified by the dimension of the null-space* N *of* A *in (11.3–*p*279).*

estimation algorithms is important because configurations "close to" degenerate ones are likely to lead to a numerically ill-conditioned estimation. The degeneracies are summarized in table 11.1.

11.9.1 Points on a ruled quadric

It will be shown in chapter 22 that degeneracy occurs if both camera centres and all the 3D points lie on a (ruled) quadric surface referred to as the *critical surface* [Maybank-93]. A ruled quadric may be non-degenerate (a hyperboloid of one sheet – a cooling tower) or degenerate (for instance two planes, cones, and cylinders) – see section 3.2.4(*p*74); but a critical surface cannot be an ellipsoid or hyperboloid of two sheets. For a critical surface configuration there are three possible fundamental matrices.

Note that in the case of just 7 point correspondences, together with the two camera centres there are 9 points in total. A general quadric has 9 degrees of freedom, and one may always construct a quadric through 9 points. In the case where this quadric is a ruled quadric it will be a critical surface, and there will be three possible solutions for F. The case where the quadric is not ruled corresponds to the case where there is only one real solution for F.

11.9.2 Points on a plane

An important degeneracy is when all the points lie in a plane. In this case, all the points plus the two camera centres lie on a ruled quadric surface, namely the degenerate quadric consisting of two planes – the plane through the points, plus a plane passing through the two camera centres.

Two views of a planar set of points are related via a 2D projective transformation H. Thus, suppose that a set of correspondences $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$ is given for which $\mathbf{x}'_i = \mathtt{H}\mathbf{x}_i$. Any number of points $\mathbf{x}_i$ and the corresponding points $\mathbf{x}'_i = \mathtt{H}\mathbf{x}_i$ may be given.

The fundamental matrix corresponding to the pair of cameras satisfies the equation $\mathbf{x}_i'^{\mathsf{T}}\mathtt{F}\mathbf{x}_i = \mathbf{x}_i'^{\mathsf{T}}(\mathtt{F}\mathtt{H}^{-1})\mathbf{x}_i' = 0$. This set of equations is satisfied whenever $\mathtt{F}\mathtt{H}^{-1}$ is skew-symmetric. Thus, the solution for F is any matrix of the form $\mathtt{F} = \mathtt{S}\mathtt{H}$, where S is skew-symmetric. Now, a 3×3 skew-symmetric matrix S may be written in the form $\mathtt{S} = [\mathbf{t}]_\times$, for any 3-vector $\mathbf{t}$. Thus, S has three degrees of freedom, and consequently so does F. More precisely, the correspondences $\mathbf{x}_i \leftrightarrow \mathbf{x}_i'$ lead to a three-parameter family of possible fundamental matrices F (note, one of the parameters accounts for scaling the matrix so there is only a *two*-parameter family of homogeneous matrices). The equation matrix A derived from the set of correspondences must therefore have rank no greater than 6.

From the decomposition of $\mathtt{F} = \mathtt{S}\mathtt{H}$, it follows from result 9.9(*p*254) that the pair of camera matrices $[\mathtt{I} \mid \mathbf{0}]$ and $[\mathtt{H} \mid \mathbf{t}]$ correspond to the fundamental matrix F. Here, the vector $\mathbf{t}$ may take on any value. If point $\mathbf{x}_i = (x_i, y_i, 1)^{\mathsf{T}}$ and $\mathbf{x}_i' = \mathtt{H}\mathbf{x}_i$, then one verifies that the point $\mathbf{X}_i = (x, y, 1, 0)^{\mathsf{T}}$ maps to $\mathbf{x}_i$ and $\mathbf{x}_i'$ through the two cameras. Thus, the points $\mathbf{X}_i$ constitute a reconstruction of the scene.

11.9.3 No translation

If the two camera centres are coincident then the epipolar geometry is not defined. In addition, formulae such as result 9.9(*p*254) give a value of 0 for the fundamental matrix. In this case the two images are related by a 2D homography (see section 8.4.2(*p*204)).

If one attempts to find the fundamental matrix then, as shown above, there will be at least a 2-parameter family of solutions for F. Even if the camera motion involves no translation, then a method such as the 8-point algorithm used to compute the fundamental matrix will still produce a matrix F satisfying $\mathbf{x}_i'^{\mathsf{T}}\mathtt{F}\mathbf{x}_i = 0$, where F has the form $\mathtt{F} = \mathtt{S}\mathtt{H}$, H is the homography relating the points, and S is an essentially arbitrary skew-symmetric matrix. Points $\mathbf{x}_i$ and $\mathbf{x}_i'$ related by H will satisfy this relationship.

11.10 A geometric interpretation of F-computation

The estimation of F from a set of image correspondences $\{\mathbf{x}_i \leftrightarrow \mathbf{x}_i'\}$ has many similarities with the problem of estimating a conic from a set of 2D points $\{x_i, y_i\}$ (or a quadric from a set of 3D points).

The equation $\mathbf{x}'^{\mathsf{T}}\mathtt{F}\mathbf{x} = 0$ is a single constraint in x, y, x', y' and so defines a surface (variety) $\mathcal{V}$ of codimension 1 (dimension 3) in $\mathbb{R}^4$. The surface is a quadric because the equation is quadratic in the coordinates x, y, x', y' of $\mathbb{R}^4$. There is a natural mapping from projective 3-space to the variety $\mathcal{V}$ that takes any 3D point to the quadruple $(x, y, x', y')^{\mathsf{T}}$ of the corresponding image points in the two views. The quadric form is evident if $\mathbf{x}'^{\mathsf{T}}\mathtt{F}\mathbf{x} = 0$ is rewritten as

$$\begin{pmatrix} x & y & x' & y' & 1 \end{pmatrix} \begin{bmatrix} 0 & 0 & f_{11} & f_{21} & f_{31} \\ 0 & 0 & f_{12} & f_{22} & f_{32} \\ f_{11} & f_{12} & 0 & 0 & f_{13} \\ f_{21} & f_{22} & 0 & 0 & f_{23} \\ f_{31} & f_{32} & f_{13} & f_{23} & 2f_{33} \end{bmatrix} \begin{pmatrix} x \\ y \\ x' \\ y' \\ 1 \end{pmatrix} = 0 \ .$$

The case of conic fitting is a good (lower-dimensional) model of F estimation. To

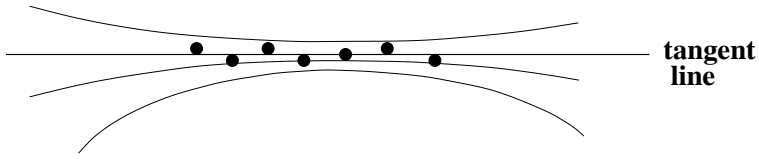


Fig. 11.7. *Estimating a conic from point data (shown as •) may be poorly conditioned. All of the conics shown have residuals within the point error distribution. However, even though there is ambiguity in the estimated conic, the tangent line is well defined, and can be computed from the points.*

bring out the analogy between the two estimation problems: a point (x_i, y_i) places one constraint on the 5 degrees of freedom of a conic as described in section 2.2.3(*p*30):

$$ax_i^2 + bx_iy_i + cy_i^2 + dx_i + ey_i + f = 0.$$

Similarly, a point correspondence (x_i, y_i, x_i', y_i') places one constraint on the (8) degrees of freedom of $\mathtt{F}$ as (11.2–*p*279):

$$x_i'x_if_{11} + x_i'y_if_{12} + x_i'f_{13} + y_i'x_if_{21} + y_i'y_if_{22} + y_i'f_{23} + x_if_{31} + y_if_{32} + f_{33} = 0.$$

It is not quite an exact analogy, since the defining relationship expressed by the fundamental matrix is bilinear in the two sets of indices, as is also evident from the zeros in the quadric matrix above, whereas in the case of a conic section the equation is an arbitrary quadratic. Also the surface defined by $\mathtt{F}$ must satisfy an additional constraint arising from $\det(\mathtt{F}) = 0$, and there is no such constraint in the conic fitting analogue.

The problems of extrapolation when data has only been fitted to a small section of a conic are well known, and similar issues arise in fitting the fundamental matrix to data. Indeed, there are cases where the data is sufficient to determine an accurate tangent line to the conic, but insufficient to determine the conic itself, see figure 11.7. In the case of the fundamental matrix the tangent plane to the quadric in $\mathbb{R}^4$ is the affine fundamental matrix (chapter 14), and this approximation may be fitted when perspective effects are small.

11.11 The envelope of epipolar lines

One of the uses of the fundamental matrix is to determine epipolar lines in a second image corresponding to points in a first image. For instance, if one is seeking matched points between two images, the match of a given point $\mathbf{x}$ in the first image may be found by searching along the epipolar line $\mathtt{F}\mathbf{x}$ in the second image. In the presence of noise, of course, the matching point will not lie precisely on the line $\mathtt{F}\mathbf{x}$ because the fundamental matrix will be known only within certain bounds, expressed by its covariance matrix. In general, instead of searching along the epipolar line only, it will be necessary to search in a region on either side of the line $\mathtt{F}\mathbf{x}$. We will now consider how the covariance matrix of the fundamental matrix may be used to determine the region in which to search.

Let $\mathbf{x}$ be a point and $\mathtt{F}$ be a fundamental matrix for which one has computed a covariance matrix $\Sigma_{\mathtt{F}}$. The point $\mathbf{x}$ corresponds to an epipolar line $\mathbf{l} = \mathtt{F}\mathbf{x}$, and one

may transfer the covariance matrix $\Sigma_{\mathtt{F}}$ to a covariance matrix $\Sigma_{\mathtt{l}}$ according to result 5.6-(p139). Also by result 5.6(p139), the mean value of the epipolar line is given by $\bar{\mathtt{l}} = \bar{\mathtt{F}}\mathbf{x}$. To avoid singular cases, the vector $\mathbf{l}$ representing an epipolar line is normalized so that $\|\mathbf{l}\| = 1$. Then the mapping $\mathbf{x} \mapsto \mathbf{l}$ is given by $\mathbf{l} = (\mathtt{F}\mathbf{x})/\|\mathtt{F}\mathbf{x}\|$. If $\mathtt{J}$ is the Jacobian matrix of this mapping with respect to the entries of $\mathtt{F}$, then $\mathtt{J}$ is a 3×9 matrix, and $\Sigma_{\mathtt{l}} = \mathtt{J}\Sigma_{\mathtt{F}}\mathtt{J}^{\mathsf{T}}$.

Though the constraint $\|\mathbf{l}\| = 1$ is the most convenient constraint, the following analysis applies for any constraint used to confine the vector representing the epipolar line to vary on a 2-dimensional surface in $\mathbb{R}^3$. In this case, the covariance matrix $\Sigma_{\mathtt{l}}$ is singular, having rank 2, since no variation is allowed in the direction normal to the constraint surface. For a particular instance of $\mathbf{l}$, the deviation from the mean, $\bar{\mathbf{l}} - \mathbf{l}$, must be along the constraint surface, and hence (in the linear approximation) perpendicular to the null-space of $\Sigma_{\mathtt{l}}$.

For the remainder of this derivation, $\bar{\mathbf{l}}$, the vector representing the mean epipolar line, will be denoted by $\mathbf{m}$, so as to avoid confusing notation. Now, assuming a Gaussian distribution for the vectors $\mathbf{l}$ representing the epipolar line, the set of all lines having a given likelihood is given by the equation

$$(\mathbf{l} - \mathbf{m})^{\mathsf{T}}\Sigma_{\mathtt{l}}^{+}(\mathbf{l} - \mathbf{m}) = k^2 \tag{11.11}$$

where k is some constant. To analyze this further, we apply an orthogonal change of coordinates such that $\Sigma_{\mathtt{l}}$ becomes diagonal. Thus, one may write

$$\mathtt{U}\Sigma_{\mathtt{l}}\mathtt{U}^{\mathsf{T}} = \Sigma'_{\mathtt{l}} = \begin{bmatrix} \tilde{\Sigma}'_{\mathtt{l}} & \mathbf{0} \\ \mathbf{0}^{\mathsf{T}} & 0 \end{bmatrix}$$

where $\tilde{\Sigma}'_{\mathtt{l}}$ is a 2×2 non-singular diagonal matrix. Applying the same transformation to the lines, one defines 2-vectors $\mathbf{m}' = \mathtt{U}\mathbf{m}$ and $\mathbf{l}' = \mathtt{U}\mathbf{l}$. Since $\mathbf{l}' - \mathbf{m}'$ is orthogonal to the null-space $(0, 0, 1)^{\mathsf{T}}$ of $\Sigma'_{\mathtt{l}}$, both $\mathbf{m}'$ and $\mathbf{l}'$ have the same third coordinate. By multiplying $\mathtt{U}$ by a constant as necessary, one may assume that this coordinate is 1. Thus we may write $\mathbf{l}' = (\tilde{\mathbf{l}}'^{\mathsf{T}}, 1)^{\mathsf{T}}$ and $\mathbf{m}' = (\tilde{\mathbf{m}}'^{\mathsf{T}}, 1)^{\mathsf{T}}$ for certain 2-vectors $\tilde{\mathbf{l}}'$ and $\tilde{\mathbf{m}}'$. Then, one verifies that

$$\begin{aligned} k^2 &= (\mathbf{l} - \mathbf{m})^{\mathsf{T}}\Sigma_{\mathtt{l}}^{+}(\mathbf{l} - \mathbf{m}) \\ &= (\mathbf{l}' - \mathbf{m}')^{\mathsf{T}}\Sigma_{\mathtt{l}}'^{+}(\mathbf{l}' - \mathbf{m}') \\ &= (\tilde{\mathbf{l}}' - \tilde{\mathbf{m}}')^{\mathsf{T}}\tilde{\Sigma}_{\mathtt{l}}'^{-1}(\tilde{\mathbf{l}}' - \tilde{\mathbf{m}}'). \end{aligned}$$

This equation expands out to

$$\tilde{\mathbf{l}}'^{\mathsf{T}}\tilde{\Sigma}_{\mathtt{l}}'^{-1}\tilde{\mathbf{l}}' - \tilde{\mathbf{m}}'^{\mathsf{T}}\tilde{\Sigma}_{\mathtt{l}}'^{-1}\tilde{\mathbf{l}}' - \tilde{\mathbf{l}}'^{\mathsf{T}}\tilde{\Sigma}_{\mathtt{l}}'^{-1}\tilde{\mathbf{m}}' + \tilde{\mathbf{m}}'^{\mathsf{T}}\tilde{\Sigma}_{\mathtt{l}}'^{-1}\tilde{\mathbf{m}}' - k^2 = 0$$

which may be written as

$$(\tilde{\mathbf{l}}'^{\mathsf{T}} \;\; 1) \begin{bmatrix} \tilde{\Sigma}_{\mathtt{l}}'^{-1} & -\tilde{\Sigma}_{\mathtt{l}}'^{-1}\tilde{\mathbf{m}}' \\ -\tilde{\mathbf{m}}'^{\mathsf{T}}\tilde{\Sigma}_{\mathtt{l}}'^{-1} & \tilde{\mathbf{m}}'^{\mathsf{T}}\tilde{\Sigma}_{\mathtt{l}}'^{-1}\tilde{\mathbf{m}}' - k^2 \end{bmatrix} \begin{pmatrix} \tilde{\mathbf{l}}' \\ 1 \end{pmatrix} = 0$$

or equivalently (as one may verify)

$$(\tilde{\mathbf{l}}'^{\mathsf{T}} \;\; 1) \begin{bmatrix} \tilde{\mathbf{m}}'\tilde{\mathbf{m}}'^{\mathsf{T}} - k^2\tilde{\Sigma}'_{\mathbf{l}} & \tilde{\mathbf{m}}' \\ \tilde{\mathbf{m}}'^{\mathsf{T}} & 1 \end{bmatrix}^{-1} \begin{pmatrix} \tilde{\mathbf{l}}' \\ 1 \end{pmatrix} = 0. \tag{11.12}$$

Finally, this is equivalent to

$$\mathbf{l}'^{\mathsf{T}}[\mathbf{m}'\mathbf{m}'^{\mathsf{T}} - k^2\Sigma'_{\mathbf{l}}]^{-1}\mathbf{l}' = 0. \tag{11.13}$$

This shows that the lines satisfying (11.11) form a line conic defined by the matrix $(\mathbf{m}'\mathbf{m}'^{\mathsf{T}} - k^2\Sigma'_{\mathbf{l}})^{-1}$. The corresponding point conic, which forms the envelope of the lines, is defined by the matrix $\mathbf{m}'\mathbf{m}'^{\mathsf{T}} - k^2\Sigma'_{\mathbf{l}}$. One may now transform back to the original coordinate system to determine the envelope of the lines in the original coordinate system. The transformed conic is

$$\mathtt{C} = \mathtt{U}^{\mathsf{T}}(\mathbf{m}'\mathbf{m}'^{\mathsf{T}} - k^2\Sigma'_{\mathbf{l}})\mathtt{U} = \mathbf{m}\mathbf{m}^{\mathsf{T}} - k^2\Sigma_{\mathbf{l}}. \tag{11.14}$$

Note that when $k = 0$, the conic $\mathtt{C}$ degenerates to $\mathbf{m}\mathbf{m}^{\mathsf{T}}$, which represents the set of points lying on the line $\mathbf{m}$. As k increases, the conic becomes a hyperbola the two branches of which lie on opposite sides of the line $\mathbf{m}$.

Suppose we want to choose k so that some fraction α of the epipolar lines lie inside the region bounded by this hyperbola. The value $k^2 = (\mathbf{l} - \mathbf{m})^{\mathsf{T}}\Sigma_{\mathbf{l}}^{+}(\mathbf{l} - \mathbf{m})$ of (11.11) follows a χ^2_n distribution, and the cumulative chi-squared distribution $F_n(k^2) = \int_0^{k^2} \chi^2_n(\xi)d\xi$ represents the probability that the value of a χ^2_n random variable is less than k^2 (the χ^2_n and F_n distributions are defined in section A2.2(*p*566)). Applying this to a random line $\mathbf{l}$, one sees that in order to ensure that a fraction α of lines lie within region bounded by the hyperbola defined by (11.14), one must choose k^2 such that $F_2(k^2) = \alpha$ ($n = 2$ since the covariance matrix $\Sigma_{\mathbf{l}}$ has rank 2). Thus, $k^2 = F_2^{-1}(\alpha)$, and for a value of $\alpha = 0.95$, for instance, one finds that $k^2 = 5.9915$. The corresponding hyperbola given by (11.14) is $\mathtt{C} = \mathbf{m}\mathbf{m}^{\mathsf{T}} - 5.9915\,\Sigma_{\mathbf{l}}$. To sum up this discussion:

Result 11.2. *If* $\mathbf{l}$ *is a random line obeying a Gaussian distribution with mean* $\bar{\mathbf{l}}$ *and covariance matrix* $\Sigma_{\mathbf{l}}$ *of rank 2, then the plane conic*

$$\mathtt{C} = \bar{\mathbf{l}}\,\bar{\mathbf{l}}^{\mathsf{T}} - k^2\Sigma_{\mathbf{l}} \tag{11.15}$$

represents an equal-likelihood contour bounding some fraction of all instances of $\mathbf{l}$*. If* $F_2(k^2)$ *represents the cumulative* χ^2_2 *distribution, and* k^2 *is chosen such that* $F_2^{-1}(k^2) = \alpha$*, then a fraction* α *of all lines lie within the region bounded by* $\mathtt{C}$*. In other words with probability* α *the lines lie within this region.*

In applying this formula, one must be aware that it represents only an approximation, since epipolar lines are not normally distributed. We have consistently made the assumption that the distributions may be correctly transformed using the Jacobian, that is an assumption of linearity. This assumption will be most reasonable for distributions with small variance, and close to the mean. Here, we are applying it to find the region in which as many as 95% of samples fall, namely almost the whole of the error

distribution. In this case, the assumption of a Gaussian distribution of errors is less tenable.

11.11.1 Verification of epipolar line covariance

We now present some examples of epipolar line envelopes, confirming and illustrating the theory developed above. Before doing this, however, a direct verification of the theory will be given, concerning the covariance matrix of epipolar lines. Since the 3×3 covariance matrix of a line is not easily understood quantitatively, we consider the variance of the direction of epipolar lines. Given a line $\mathbf{l} = (l_1, l_2, l_3)^\mathsf{T}$, the angle representing its direction is given by $\theta = \arctan(-l_1/l_2)$. Letting $\mathtt{J}$ equal the 1×3 Jacobian matrix of the mapping $\mathbf{l} \rightarrow \theta$, one finds the variance of the angle θ to be $\sigma_\theta^2 = \mathtt{J}\Sigma_\mathbf{l}\mathtt{J}^\mathsf{T}$. This result may be verified by simulation, as follows.

One considers a pair of images for which point correspondences have been identified. The fundamental matrix is computed from the point correspondences and the points are then corrected so as to correspond precisely under the epipolar mapping (as described in section 12.3). A set of n of these corrected correspondences are used to compute the covariance matrix of the fundamental matrix $\mathtt{F}$. Then, for a further set of "test" corrected points $\mathbf{x}_i$ in the first image, the mean and covariance of the corresponding epipolar line $\mathbf{l}'_i = \mathtt{F}\mathbf{x}_i$ are computed, and subsequently the mean and variance of the orientation direction of this line are computed. This gives the theoretical values of these quantities.

Next, Monte Carlo simulation is done, in which Gaussian noise is added to the coordinates of the points used to compute $\mathtt{F}$. Using the computed $\mathtt{F}$, the epipolar lines corresponding to each of the test points are computed, and subsequently their angle, and the deviation of the angle from the mean. This is done many times, and the standard deviation of angle is computed, and finally compared with the theoretical value. The results of this are shown in figure 11.8 for the statue image pair of figure 11.2-(*p*289).

Epipolar envelopes for statue image. The statue image pair of figure 11.2(*p*289) is interesting because of the large depth variation across the image. There are close points (on the statue) and distant points (on the building behind) in close proximity in the images. The fundamental matrix was computed from several points. A point in the first image (see figure 11.9) was selected and Monte Carlo simulation was used to compute several possible epipolar lines corresponding to a noise level of 0.5 pixels in each matched point coordinate. To test the theory, the mean and covariance of the epipolar line were next computed theoretically. The 95% envelope of the epipolar lines was computed and drawn in the second image. The results are shown in figure 11.10 for different numbers of points used to compute $\mathtt{F}$. The 95% envelope for $n = 15$ corresponds closely to the simulated envelope of the lines.

The results shown in figure 11.10 show the practical importance of computing the epipolar envelopes in point matching. Thus, suppose one is attempting to find a match for the foreground point in figure 11.9. If the epipolar line is computed from just 10 point matches, then epipolar search is unlikely to succeed, given the width of the

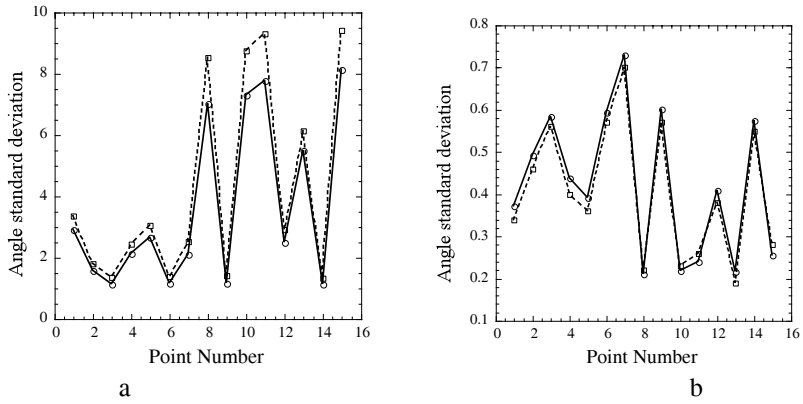


Fig. 11.8. *Comparison of theoretical and Monte Carlo simulated values of orientation angle of epipolar lines for 15 test points form the statue image pair of figure 11.2(*p289*). The horizontal axis represents the point number (1 to 15) and the vertical axis the standard deviation of angle. (a) the results when the epipolar structure (fundamental matrix) is computed from 15 points. (b) the results when 50 point matches are used.* **Note :** *the horizontal axis of these graphs represent discrete points numbered 1 to 15. The graphs are shown as a continuous curve only for visual clarity.*

envelope. Even for $n = 15$, the width of the envelope at the level of the correct match is several tens of pixels. For $n = 25$, the situation is more favourable. Note that this instability is inherent in the problem, and not the result of any specific algorithm for computing F.

An interesting point concerns the location of the narrowest point of the envelope. In this case, it appears to be close to the correct match position for the background point in figure 11.9. The match for the foreground point (leg of statue) lies far from the narrowest point of the envelope. Though the precise location of the narrow point of the envelope is not fully understood, it appears that in this case, most points used to the computation of F are on the background building. This biases towards the supposition that other matched points lie close to the plane of the building. The match for a point at significantly different depth is less precisely known.

Matching points close to the epipole – the corridor scene. In the case where the points to be matched are close to the epipole, then the determination of the epipolar line is more unstable, since any uncertainty in the position of the epipole results in uncertainty in the slope of the epipolar line. In addition, as one approaches this unstable position, the linear approximations implicit in the derivation of (11.14) become less tenable. In particular, the distribution of the epipolar lines deviates from a normal distribution.

11.12 Image rectification

This section gives a method for image rectification, the process of resampling pairs of stereo images taken from widely differing viewpoints in order to produce a pair of "matched epipolar projections". These are projections in which the epipolar lines run parallel with the x-axis and match up between views, and consequently disparities between the images are in the x-direction only, i.e. there is no y disparity.

a b

Fig. 11.9. *(a) The point in the first image used to compute the epipolar envelopes in the second images. Note the ambiguity of which point is to be found in the second image. The marked point may represent the point on the statue's leg (foreground) or the point on the building behind the statue (background). In the second image, these two points are quite separate, and the epipolar line must pass through them both. (b) Computed corresponding epipolar lines computed from $n = 15$ point matches. The different lines correspond to different instances of injected noise in the matched points. Gaussian noise of 0.5 pixels in each coordinate was added to the ideal matched point positions before computing the epipolar line corresponding to the selected point. The ML estimator (Gold Standard algorithm) was used to compute* F*. This experiment shows the basic instability of the computation of the epipolar lines from small numbers of points. To find the point matching the selected point in the image at left, one needs to search over the regions covered by all these epipolar lines.*

The method is based on the fundamental matrix. A pair of 2D projective transformations are applied to the two images in order to match the epipolar lines. It is shown that the two transformations may be chosen in such a way that matching points have approximately the same x-coordinate as well. In this way, the two images, if overlaid on top of each other, will correspond as far as possible, and any disparities will be parallel to the x-axis. Since the application of arbitrary 2D projective transformations may distort the image substantially, the method for finding the pair of transformations subjects the images to a minimal distortion.

In effect, transforming the two images by the appropriate projective transformations reduces the problem to the epipolar geometry produced by a pair of identical cameras placed side by side with their principal axes parallel. Many stereo matching algorithms described in previous literature have assumed this geometry. After this rectification the search for matching points is vastly simplified by the simple epipolar structure and by the near-correspondence of the two images. It may be used as a preliminary step to comprehensive image matching.

11.12.1 Mapping the epipole to infinity

In this section we will discuss the question of finding a projective transformation H of an image that maps the epipole to a point at infinity. In fact, if epipolar lines are to be transformed to lines parallel with the x-axis, then the epipole should be mapped to

Fig. 11.10. *The 95% envelopes of epipolar lines are shown for a noise level of* 0.5 *pixels, with* F *being computed from* $n = 10, 15, 25$ *and* 50 *points. In each case, Monte Carlo simulated results agreed closely with these results (though not shown here). For the case* $n = 15$*, compare with figure 11.9. Note that for* $n = 10$*, the epipolar envelope is very wide (*> 90 *degrees), showing that one can have very little confidence in an epipolar line computed from 10 points in this case. For* $n = 15$*, the envelope is still quite wide. For* $n = 25$ *and* $n = 50$*, the epipolar line is known with quite good precision. Of course, the precise shape of the envelope depends strongly on just what matched points are used to compute the epipolar structure.*

the particular infinite point $(1, 0, 0)^\mathsf{T}$. This leaves many degrees of freedom (in fact four) open for H, and if an inappropriate H is chosen, severe projective distortion of the image can take place. In order that the resampled image should look somewhat like the original image, we may put closer restrictions on the choice of H.

One condition that leads to good results is to insist that the transformation H should act as far as possible as a rigid transformation in the neighbourhood of a given selected point $\mathbf{x}_0$ of the image. By this is meant that to first-order the neighbourhood of $\mathbf{x}_0$ may undergo rotation and translation only, and hence will look the same in the original and resampled images. An appropriate choice of point $\mathbf{x}_0$ may be the centre of the image. For instance, this would be a good choice in the context of aerial photography if the view is known not to be excessively oblique.

For the present, suppose $\mathbf{x}_0$ is the origin and the epipole $\mathbf{e} = (f, 0, 1)^\mathsf{T}$ lies on the

x-axis. Now consider the following transformation

$$\mathtt{G} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -1/f & 0 & 1 \end{bmatrix}. \tag{11.16}$$

This transformation takes the epipole $(f, 0, 1)^\mathsf{T}$ to the point at infinity $(f, 0, 0)^\mathsf{T}$ as required. A point $(x, y, 1)^\mathsf{T}$ is mapped by $\mathtt{G}$ to the point $(\hat{x}, \hat{y}, 1)^\mathsf{T} = (x, y, 1 - x/f)^\mathsf{T}$. If $|x/f| < 1$ then we may write

$$(\hat{x}, \hat{y}, 1)^\mathsf{T} = (x, y, 1 - x/f)^\mathsf{T} = (x(1 + x/f + \ldots), y(1 + x/f + \ldots), 1)^\mathsf{T}.$$

The Jacobian is

$$\frac{\partial(\hat{x}, \hat{y})}{\partial(x, y)} = \begin{bmatrix} 1 + 2x/f & 0 \\ y/f & 1 + x/f \end{bmatrix}$$

plus higher order terms in x and y. Now if $x = y = 0$ then this is the identity map. In other words, $\mathtt{G}$ is approximated (to first-order) at the origin by the identity mapping.

For an arbitrarily placed point of interest $\mathbf{x}_0$ and epipole $\mathbf{e}$, the required mapping $\mathtt{H}$ is a product $\mathtt{H} = \mathtt{GRT}$ where $\mathtt{T}$ is a translation taking the point $\mathbf{x}_0$ to the origin, $\mathtt{R}$ is a rotation about the origin taking the epipole $\mathbf{e}'$ to a point $(f, 0, 1)^\mathsf{T}$ on the x-axis, and $\mathtt{G}$ is the mapping just considered taking $(f, 0, 1)^\mathsf{T}$ to infinity. The composite mapping is to first-order a rigid transformation in the neighbourhood of $\mathbf{x}_0$.

11.12.2 Matching transformations

In the previous section it was shown how the epipole in one image may be mapped to infinity. Next, it will be seen how a map may be applied to the other image to match up the epipolar lines. We consider two images J and J'. The intention is to resample these two images according to transformations $\mathtt{H}$ to be applied to J and $\mathtt{H}'$ to be applied to J'. The resampling is to be done in such a way that an epipolar line in J is matched with its corresponding epipolar line in J'. More specifically, if $\mathbf{l}$ and $\mathbf{l}'$ are any pair of corresponding epipolar lines in the two images, then $\mathtt{H}^{-\mathsf{T}}\mathbf{l} = \mathtt{H}'^{-\mathsf{T}}\mathbf{l}'$. (Recall that $\mathtt{H}^{-\mathsf{T}}$ is the line map corresponding to the point map $\mathtt{H}$.) Any pair of transformations satisfying this condition will be called a *matched pair* of transformations.

Our strategy in choosing a matched pair of transformations is to choose $\mathtt{H}'$ first to be some transformation that sends the epipole $\mathbf{e}'$ to infinity as described in the previous section. We then seek a matching transformation $\mathtt{H}$ chosen so as to minimize the sum-of-squared distances

$$\sum_i d(\mathtt{H}\mathbf{x}_i, \mathtt{H}'\mathbf{x}_i')^2. \tag{11.17}$$

The first question to be determined is how to find a transformation matching $\mathtt{H}'$. That question is answered in the following result.

Result 11.3. *Let J and J' be images with fundamental matrix $\mathtt{F} = [\mathbf{e}']_\times\mathtt{M}$, and let $\mathtt{H}'$ be a projective transformation of J'. A projective transformation $\mathtt{H}$ of J matches $\mathtt{H}'$ if and*

only if $\mathtt{H}$ *is of the form*

$$\mathtt{H} = (\mathtt{I} + \mathtt{H}'\mathbf{e}'\mathbf{a}^\mathsf{T})\mathtt{H}'\mathtt{M} \tag{11.18}$$

for some vector $\mathbf{a}$.

Proof. If $\mathbf{x}$ is a point in J, then $\mathbf{e}\times\mathbf{x}$ is the epipolar line in the first image, and $\mathtt{F}\mathbf{x}$ is the epipolar line in the second image. Transformations $\mathtt{H}$ and $\mathtt{H}'$ are a matching pair if and only if $\mathtt{H}^{-\mathsf{T}}(\mathbf{e}\times\mathbf{x}) = \mathtt{H}'^{-\mathsf{T}}\mathtt{F}\mathbf{x}$. Since this must hold for all $\mathbf{x}$ we may write equivalently $\mathtt{H}^{-\mathsf{T}}[\mathbf{e}]_\times = \mathtt{H}'^{-\mathsf{T}}\mathtt{F} = \mathtt{H}'^{-\mathsf{T}}[\mathbf{e}']_\times\mathtt{M}$ or, applying result A4.3(*p*582),

$$[\mathtt{H}\mathbf{e}]_\times\mathtt{H} = [\mathtt{H}'\mathbf{e}']_\times\mathtt{H}'\mathtt{M}. \tag{11.19}$$

In view of lemma 9.11(*p*255), this implies $\mathtt{H} = (\mathtt{I} + \mathtt{H}'\mathbf{e}'\mathbf{a}^\mathsf{T})\mathtt{H}'\mathtt{M}$ as required.
To prove the converse, if (11.18) holds, then

$$\begin{aligned}\mathtt{H}\mathbf{e} &= (\mathtt{I} + \mathtt{H}'\mathbf{e}'\mathbf{a}^\mathsf{T})\mathtt{H}'\mathtt{M}\mathbf{e} = (\mathtt{I} + \mathtt{H}'\mathbf{e}'\mathbf{a}^\mathsf{T})\mathtt{H}'\mathbf{e}' \\ &= (1 + \mathbf{a}^\mathsf{T}\mathtt{H}'\mathbf{e}')\mathtt{H}'\mathbf{e}' = \mathtt{H}'\mathbf{e}'.\end{aligned}$$

This, along with (11.18), is sufficient for (11.19) to hold, and so $\mathtt{H}$ and $\mathtt{H}'$ are matching transformations. □

We are particularly interested in the case when $\mathtt{H}'$ is a transformation taking the epipole $\mathbf{e}'$ to a point at infinity $(1,0,0)^\mathsf{T}$. In this case, $\mathtt{I} + \mathtt{H}'\mathbf{e}'\mathbf{a}^\mathsf{T} = \mathtt{I} + (1,0,0)^\mathsf{T}\mathbf{a}^\mathsf{T}$ is of the form

$$\mathtt{H}_\mathrm{A} = \begin{bmatrix} a & b & c \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{11.20}$$

which represents an affine transformation. Thus, a special case of result 11.3 is

Corollary 11.4. *Let* J *and* J' *be images with fundamental matrix* $\mathtt{F} = [\mathbf{e}']_\times\mathtt{M}$*, and let* $\mathtt{H}'$ *be a projective transformation of* J' *mapping the epipole* $\mathbf{e}'$ *to the infinite point* $(1,0,0)^\mathsf{T}$*. A transformation* $\mathtt{H}$ *of* J *matches* $\mathtt{H}'$ *if and only if* $\mathtt{H}$ *is of the form* $\mathtt{H} = \mathtt{H}_\mathrm{A}\mathtt{H}_0$*, where* $\mathtt{H}_0 = \mathtt{H}'\mathtt{M}$ *and* $\mathtt{H}_\mathrm{A}$ *is an affine transformation of the form (11.20).*

Given $\mathtt{H}'$ mapping the epipole to infinity, we may use this corollary to make the choice of a matching transformation $\mathtt{H}$ to minimize the disparity. Writing $\hat{\mathbf{x}}'_i = \mathtt{H}'\mathbf{x}'_i$ and $\hat{\mathbf{x}}_i = \mathtt{H}_0\mathbf{x}_i$, the minimization problem (11.17) is to find $\mathtt{H}_\mathrm{A}$ of the form (11.20) such that

$$\sum_i d(\mathtt{H}_\mathrm{A}\hat{\mathbf{x}}_i, \hat{\mathbf{x}}'_i)^2 \tag{11.21}$$

is minimized.

In particular, let $\hat{\mathbf{x}}_i = (\hat{x}_i, \hat{y}_i, 1)^\mathsf{T}$, and let $\hat{\mathbf{x}}'_i = (\hat{x}'_i, \hat{y}'_i, 1)^\mathsf{T}$. Since $\mathtt{H}'$ and $\mathtt{M}$ are known, these vectors may be computed from the matched points $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$. Then the quantity to be minimized (11.21) may be written as

$$\sum_i (a\hat{x}_i + b\hat{y}_i + c - \hat{x}'_i)^2 + (\hat{y}_i - \hat{y}'_i)^2.$$

Since $(\hat{y}_i - \hat{y}'_i)^2$ is a constant, this is equivalent to minimizing

$$\sum_i (a\hat{x}_i + b\hat{y}_i + c - \hat{x}'_i)^2.$$

This is a simple linear least-squares parameter minimization problem, and is easily solved using linear techniques (see section A5.1(*p*588)) to find a, b and c. Then H_{A} is computed from (11.20) and $\mathtt{H}$ from (11.18). Note that a linear solution is possible because $\mathtt{H}_{\mathrm{A}}$ is an affine transformation. If it were simply a projective transformation, this would not be a linear problem.

11.12.3 Algorithm outline

The resampling algorithm will now be summarized. The input is a pair of images containing a common overlap region. The output is a pair of images resampled so that the epipolar lines in the two images are horizontal (parallel with the x-axis), and such that corresponding points in the two images are as close to each other as possible. Any remaining disparity between matching points will be along the the horizontal epipolar lines. A top-level outline of the algorithm is as follows.

(i) Identify a seed set of image-to-image matches $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$ between the two images. Seven points at least are needed, though more are preferable. It is possible to find such matches by automatic means.

(ii) Compute the fundamental matrix $\mathtt{F}$ and find the epipoles $\mathbf{e}$ and $\mathbf{e}'$ in the two images.

(iii) Select a projective transformation $\mathtt{H}'$ that maps the epipole $\mathbf{e}'$ to the point at infinity, $(1, 0, 0)^\mathsf{T}$. The method of section 11.12.1 gives good results.

(iv) Find the matching projective transformation $\mathtt{H}$ that minimizes the least-squares distance

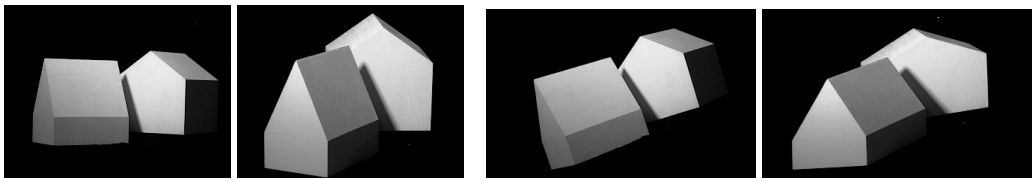
$$\sum_i d(\mathtt{H}\mathbf{x}_i, \mathtt{H}'\mathbf{x}'_i). \tag{11.22}$$

The method used is a linear method described in section 11.12.2.

(v) Resample the first image according to the projective transformation $\mathtt{H}$ and the second image according to the projective transformation $\mathtt{H}'$.

Example 11.5. Model house images

Figure 11.11(a) shows a pair of images of some wooden block houses. Edges and vertices in these two images were extracted automatically and a small number of common vertices were matched by hand. The two images were then resampled according to the methods described here. The results are shown in figure 11.11(b). In this case, because of the wide difference in viewpoint, and the three-dimensional shape of the objects, the two images even after resampling look quite different. However, it is the case that any point in the first image will now match a point in the second image with the same y-coordinate. Therefore, in order to find further point matches between the images only a 1-dimensional search is required. △



a b

Fig. 11.11. **Image rectification.** *(a) A pair of images of a house. (b) Resampled images computed from (a) using a projective transformation computed from* F*. Note, corresponding points in (b) match horizontally.*

a b

Fig. 11.12. **Image rectification using affinities.** *(a) A pair of original images and (b) a detail of the images rectified using affine transformations. The average y-disparity after rectification is of the order of 3 pixels in a* 512×512 *image. (For correctly rectified images the y-disparity should be zero.)*

11.12.4 Affine rectification

The theory discussed in this section can equally be applied to affine resampling. If the two cameras can be well approximated by affine cameras, then one can rectify the images using just affine transformations. To do this, one uses the affine fundamental matrix (see section 14.2(*p*345)) instead of the general fundamental matrix. The above method with only minor variations can then be applied to compute a pair of matching affine transformations. Figure 11.12 shows a pair of images rectified using affine transformations.

11.13 Closure

11.13.1 The literature

The basic idea behind the computation of the fundamental matrix is given in [LonguetHiggins-81], which is well worth reading. It addresses the case of calibrated matrices only, but the principles apply to the uncalibrated case as well. A good reference for the uncalibrated case is [Zhang-98] which considers most of the best methods. In addition, that paper considers the uncertainty envelopes of epipolar lines, following earlier work by Csurka *et al.* [Csurka-97]. A more detailed study of the 8-point algorithm in the uncalibrated case is given in [Hartley-97c]. Weng *et al.* [Weng-89] used Sampson approximation for the fundamental matrix cost function. The SVD method of coercing the estimated F to have rank 2 was suggested by Tsai & Huang [Tsai-84].

There is a wealth of literature on conic fitting – minimizing algebraic distance

[Bookstein-79]; approximating geometric distance [Sampson-82, Pratt-87, Taubin-91]; optimal fitting [Kanatani-94]; and fitting special forms [Fitzgibbon-99].

11.13.2 Notes and exercises

(i) Six point correspondences constrain $\mathbf{e}$ and $\mathbf{e}'$ to a plane cubic in each image ([Faugeras-93], page 298). The cubic also passes through the six points in each image. A sketch derivation of these results follows. Given six correspondences, the null-space of $\mathtt{A}$ in (11.3–*p*279) will be 3-dimensional. Then the solution is $\mathtt{F} = \alpha_1\mathtt{F}_1 + \alpha_2\mathtt{F}_2 + \alpha_3\mathtt{F}_3$, where $\mathtt{F}_i$ denotes the matrices corresponding to the vectors spanning the null-space. The epipole satisfies $\mathtt{F}\mathbf{e} = \mathbf{0}$, so that $[(\mathtt{F}_1\mathbf{e}), (\mathtt{F}_2\mathbf{e}), (\mathtt{F}_3\mathbf{e})](\alpha_1, \alpha_2, \alpha_3)^\mathsf{T} = \mathbf{0}$. Since this equation has a solution it follows that $\det[(\mathtt{F}_1\mathbf{e}), (\mathtt{F}_2\mathbf{e}), (\mathtt{F}_3\mathbf{e})] = 0$ which is a cubic in $\mathbf{e}$.

(ii) Show that the image correspondence of four coplanar points and a quadric outline determines the fundamental matrix up to a two-fold ambiguity (Hint, see algorithm 13.2(*p*336)).

(iii) Show that the corresponding images of a (plane) conic are equivalent to two constraints on $\mathtt{F}$. See [Kahl-98b] for details.

(iv) Suppose that a stereo pair of images is acquired by a camera translating forward along its principal axis. Can the geometry of image rectification described in section 11.12 be applied in this case? See [Pollefeys-99a] for an alternative rectification geometry.