6

Fundamentals of Multiple-view Geometry

Spela Ivekovic[1], Andrea Fusiello[2] and Emanuele Trucco[1]

[1]*Heriot-Watt University, Edinburgh, United Kingdom*
[2]*University of Verona, Verona, Italy*

6.1 INTRODUCTION

This chapter introduces the basic geometric concepts of multiple-view computer vision. The focus is on geometric models of perspective cameras, and the constraints and properties such models generate when multiple cameras observe the same 3D scene.

Geometric vision is an important and well-studied part of computer vision. A wealth of useful results has been achieved in the last 15 years and reported in comprehensive monographies, e.g., Faugeras (1993); Faugeras and Luong (2001); Hartley and Zisserman (2003), a sign of maturity for a research subject.

Geometric vision plays an essential role in image-based 3D communications. An estimate of the 3D structure of objects in view enables shape-specific, hence potentially very efficient coding (e.g., MPEG4-3DMC or 3D mesh coding). If a 3D model is available, as for instance in the case of the human figure in newscasting or videoconferencing sequences, it is possible to match it to the images, transmit only the model parameters, and animate a CAD figure (avatar) at the receiver. Transmitting rich 3D information, such as that computed by stereo algorithms, enables the receiver to support user-driven 3D effects, e.g., changing the viewpoint interactively. Indeed several chapters of this book draw from the geometric vision repertoire, which motivates this introduction.

It is worth reminding the reader that geometry is an important, but not the only important aspect of computer vision, and in particular of multiple-view vision. The information brought by each image pixel is two-fold: its *position* and its *colour* (or brightness, for a monochrome image). Ultimately, each computer vision system must start with brightness values, and, to smaller or greater depth, link such values to the 3D world.

3D Videocommunication — Algorithms, concepts and real-time systems in human centred communication
Edited by O. Schreer, P. Kauff and T. Sikora

This chapter is organized as follows. Section 6.2 introduces the basic geometric model of perspective projections, the celebrated *pinhole camera*. Section 6.3 moves on to the case of two cameras, the classic stereo system. We discuss how the pinhole model can be used to derive useful geometric constraints on the position of corresponding points, that is, projections of the same scene point in the two images. The concept of correspondence is a cornerstone of multiple-view vision. In this chapter we assume *known correspondences*, and explore their use in multiple-view vision. Algorithms for computing correspondences are presented in Chapter 7. Section 6.3 also addresses the important, practical problem of *reconstruction*: under what conditions can we estimate the position and shape of objects from two views? Section 6.4 extends our investigation to the general case of multiple (more than two) views. A few concluding remarks are given in Section 6.5.

6.2 PINHOLE CAMERA GEOMETRY

The pinhole camera is at the heart of the geometric model of imaging. It is described by its *optical centre* **C** (also known as the *camera projection centre*) and the *image plane*. The distance of the image plane from **C** is the *focal length* f. The line from the camera centre perpendicular to the image plane is called the *principal axis* or *optical axis* of the camera. The plane parallel to the image plane containing the optical centre is called the *principal plane* or *focal plane* of the camera.

A 3D point is projected onto the image plane with the line containing the point and the optical centre (Figure 6.1). Let the centre of projection be the origin of a Euclidean coordinate system wherein the z-axis is the principal axis. The relationship between the 3D coordinates of a scene point and the coordinates of its projection onto the image plane is described by the *central* or *perspective projection*.

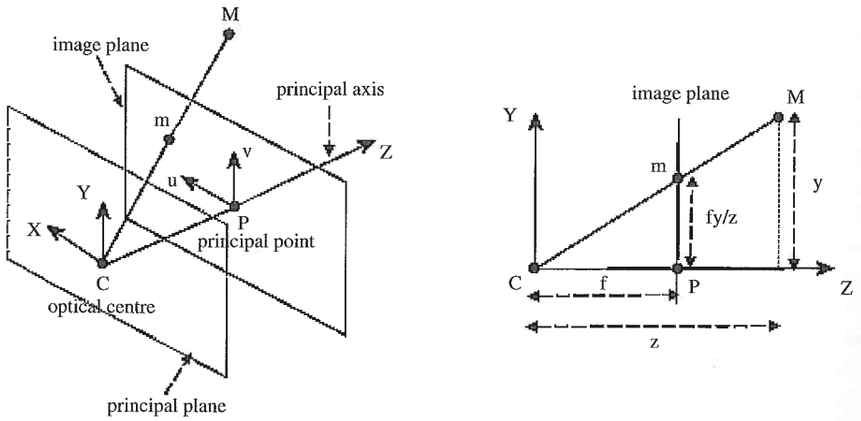

Figure 6.1 Pinhole camera geometry. The left figure illustrates the projection of the point **M** on the image plane by drawing the line through the camera centre **C** and the point to be projected. The right figure illustrates the same situation in the YZ plane, showing the similar triangles used to compute the position of the projected point **m** in the image plane

By similar triangles it is readily seen that the 3D point $(x, y, z)^T$ is mapped to the point $(fx/z, fy/z, f)^T$ on the image plane. If the world and image points are represented by homogeneous vectors, then perspective projection can be expressed in terms of matrix multiplication as

$$\begin{pmatrix} fx \\ fy \\ z \end{pmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \tag{6.1}$$

The matrix describing the linear mapping is called the *camera projection matrix* P and Equation (6.1) can be written simply as:

$$z\mathbf{m} = P\mathbf{M} \tag{6.2}$$

where $\mathbf{M} = (x, y, z, 1)^T$ are the homogeneous coordinates of the 3D point and $\mathbf{m} = (fx/z, fy/z, 1)^T$ are the homogeneous coordinates of the image point.

The projection matrix P in Equation (6.1) represents the simplest possible case, as it contains only information about the focal distance f. In general, the camera projection matrix is a 3×4 full-rank matrix and, being homogeneous, it has 11 degrees of freedom. Using QR factorization, it can be shown that any 3×4 full rank matrix P can be factorized as

$$P = K[R|\mathbf{t}] \tag{6.3}$$

where K is upper triangular (nonsingular), R is a rotation matrix, and $\mathbf{t}$ is a translation vector.

K is the *camera calibration matrix*; it encodes the transformation from camera coordinates to pixel coordinates. It depends on the so-called *intrinsic* parameters, i.e., focal distance f, image centre coordinates in pixels o_x, o_y, and pixel size in mm s_x, s_y along the two axes of the camera photosensor (Trucco and Verri 1998):

$$K = \begin{bmatrix} f/s_x & 0 & o_x \\ 0 & f/s_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \tag{6.4}$$

The *extrinsic* parameters describe the position and orientation of the camera with respect to an external (world) coordinate system and are stored in the rotation matrix R and the translation vector $\mathbf{t}$.

The camera projection centre **C** is the only point for which the projection is not defined, i.e.:

$$P\begin{pmatrix} \mathbf{C} \\ 1 \end{pmatrix} = \mathbf{0} \tag{6.5}$$

After solving for **C** we obtain:

$$\mathbf{C} = -P_{3\times3}^{-1} P_{\cdot,4} \tag{6.6}$$

where $P_{3\times3}$ is the matrix composed of the first three rows and first three columns of P, and $P_{\cdot,4}$ is the fourth column of P.

The projection can be geometrically modelled by rays through the optical centre and the point in space that is being projected onto the image plane (Figure 6.1). The *optical ray* of

an image point $\mathbf{m} = (u, v, 1)^T$ is the locus of points in space that projects onto $\mathbf{m}$. It can be described as a parametric line passing through the camera projection centre $\mathbf{C}$ and the point at infinity that projects onto $\mathbf{m}$:

$$\mathbf{M} = \begin{pmatrix} \mathbf{C} \\ 1 \end{pmatrix} + \lambda \begin{pmatrix} P_{3\times3}^{-1}\mathbf{m} \\ 0 \end{pmatrix}, \qquad \lambda \in \mathbb{R} \tag{6.7}$$

Notice that $\mathbf{C}$ is expressed in non-homogeneous coordinates, i.e., it is a three-vector. In general, the projection equation writes:

$$\zeta \mathbf{m} = P\mathbf{M} \tag{6.8}$$

where ζ is the distance of $\mathbf{M}$ from the focal plane of the camera (usually referred to as *depth*). Note that, except for a very special choice of the world reference frame, *this 'depth' does not coincide with the third coordinate of* $\mathbf{M}$. Moreover, P is a homogeneous quantity and unless it is properly normalized, ζ contains an arbitrary scale factor.

6.3 TWO-VIEW GEOMETRY

6.3.1 Introduction

The two-view geometry is the intrinsic geometry of two different perspective views of the same 3D scene (Figure 6.2). It is usually referred to as *epipolar geometry*. The two perspective views may be acquired simultaneously, for example in a stereo rig, or sequentially,

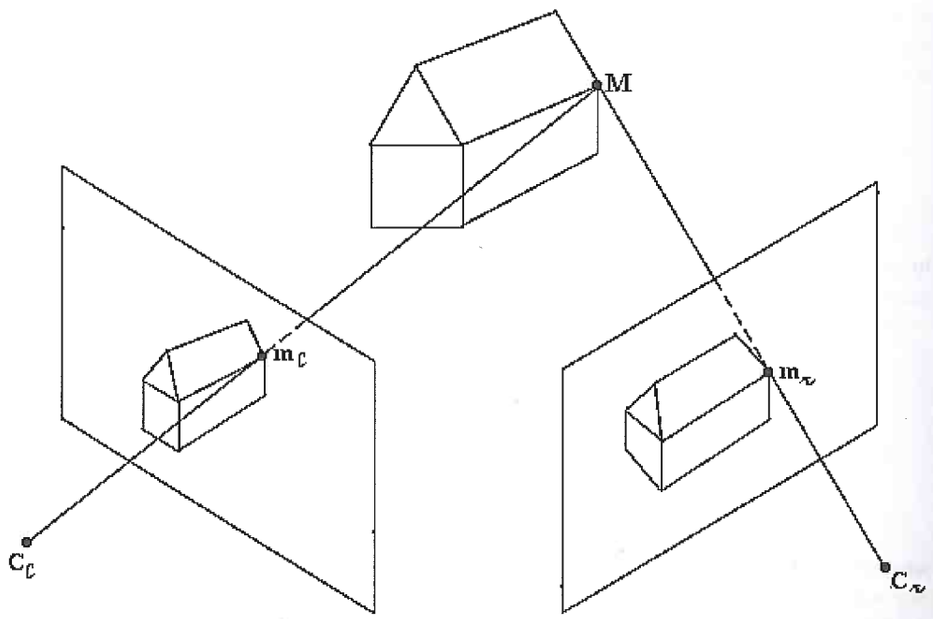

Figure 6.2 Two perspective views of the same 3D scene

for example by a moving camera. From the geometric viewpoint, the two situations are equivalent, but notice that the scene might change between successive snapshots.

Most 3D scene points must be visible in both views simultaneously. This is not true in the case of occlusions, i.e., points visible in only one camera. Any unoccluded 3D scene point $\mathbf{M} = (x, y, z, 1)^T$ is projected to the left and right view as $\mathbf{m}_l = (u_l, v_l, 1)^T$ and $\mathbf{m}_r = (u_r, v_r, 1)^T$, respectively (Figure 6.2). Image points $\mathbf{m}_l$ and $\mathbf{m}_r$ are called *corresponding points* as they represent projections of the same 3D scene point $\mathbf{M}$.

Algebraically, each perspective view has an associated 3×4 camera projection matrix P which represents the mapping between the 3D world and a 2D image. We will refer to the camera projection matrix of the left view as P_l and of the right view as P_r. The 3D point $\mathbf{M}$ is then imaged as Equation (6.9) in the left view, and Equation (6.10) in the right view:

$$\zeta_l \mathbf{m}_l = P_l \mathbf{M} \tag{6.9}$$

$$\zeta_r \mathbf{m}_r = P_r \mathbf{M} \tag{6.10}$$

Geometrically, the position of the image point $\mathbf{m}_l$ in the left image plane I_l can be found by drawing the optical ray through the left camera projection centre $\mathbf{C}_l$ and the scene point $\mathbf{M}$. The ray intersects the left image plane I_l at $\mathbf{m}_l$. Similarly, the optical ray connecting $\mathbf{C}_r$ and $\mathbf{M}$ intersects the right image plane I_r at $\mathbf{m}_r$. The relationship between image points $\mathbf{m}_l$ and $\mathbf{m}_r$ is given by the epipolar geometry, discussed in Section 6.3.2.

The knowledge of image correspondences enables scene reconstruction from images. The correctness of reconstruction crucially depends on the accuracy of image correspondences. What can be reconstructed, in turn, depends on the amount of *a priori* knowledge available about the stereo setup that was used to acquire the images and knowledge about the scene itself. This is discussed in Section 6.3.4.

6.3.2 Epipolar Geometry

The epipolar geometry describes the geometric relationship between two perspective views of the same 3D scene. The key finding, discussed below, is that *corresponding image points must lie on particular image lines*, which can be computed without the information on the camera calibration. This implies that, given a point in one image, one can search for the corresponding point in the other image along a line and not in a 2D region, a significant reduction in complexity.

Figure 6.3 illustrates the rules of the epipolar geometry. Any 3D point $\mathbf{M}$ and the camera projection centres $\mathbf{C}_l$ and $\mathbf{C}_r$ define a plane that is called the *epipolar plane*. The projections of the point $\mathbf{M}$, image points $\mathbf{m}_l$ and $\mathbf{m}_r$, also lie in the epipolar plane since they lie on the rays connecting the corresponding camera projection centre and point $\mathbf{M}$. The corresponding epipolar lines, $\mathbf{l}_l$ and $\mathbf{l}_r$, are the intersections of the epipolar plane with the image planes. The line connecting the camera projection centres $(\mathbf{C}_l, \mathbf{C}_r)$ is called the *baseline*. The baseline intersects each image plane in a point called *epipole*. By construction, the left epipole $\mathbf{e}_l$ is the image of the right camera projection centre $\mathbf{C}_r$ in the left image plane. Similarly, the right epipole $\mathbf{e}_r$ is the image of the left camera projection centre $\mathbf{C}_l$ in the right image plane. All epipolar lines in the left image go through $\mathbf{e}_l$ and all epipolar lines in the right image go through $\mathbf{e}_r$.

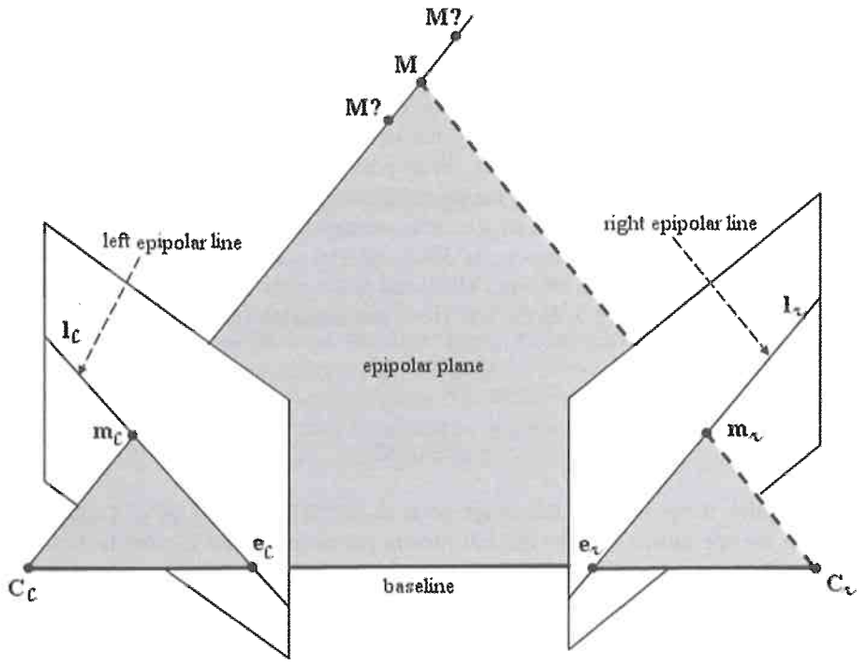

Figure 6.3 The epipolar geometry and epipolar constraint

The epipolar constraint. It is clear from the previous discussion that an epipolar plane is completely defined by the camera projection centres and one image point, say, in the left image. Therefore, given a point $\mathbf{m}_l$, one can determine the epipolar line in the right image on which the corresponding point $\mathbf{m}_r$ must lie. The logical next step is to determine the equation of the epipolar line in one image given a point in the other image.

The equation of the epipolar line can be derived from the equation describing the optical ray. As we mentioned before, the right epipolar line corresponding to the left image point $\mathbf{m}_l$ geometrically represents the projection (Equation 6.8) of the optical ray through $\mathbf{m}_l$ (Equation 6.7) onto the right image plane:

$$\zeta_r \mathbf{m}_r = P_r \mathbf{M} = P_r \begin{pmatrix} \mathbf{C}_l \\ 1 \end{pmatrix} + \lambda_l P_r \begin{pmatrix} P_{3\times3,l}^{-1} \mathbf{m}_l \\ 0 \end{pmatrix} \tag{6.11}$$

If we now simplify Equation (6.11) we obtain the description of the right epipolar line:

$$\zeta_r \mathbf{m}_r = \mathbf{e}_r + \lambda_l P_{3\times3,r} P_{3\times3,l}^{-1} \mathbf{m}_l \tag{6.12}$$

This is the equation of a line through the right epipole $\mathbf{e}_r$ and the image point $\mathbf{m}_l' = P_{3\times3,r} P_{3\times3,l}^{-1} \mathbf{m}_l$ which represents the projection of the point at infinity, lying on the optical ray of $\mathbf{m}_l$, onto the right image plane. The equation for the left epipolar line is obtained in a similar way.

As we mentioned, the epipolar constraint facilitates the search for corresponding points in two images. The correspondence search can be further simplified by *rectification*. Rectification determines a transformation of each image such that pairs of corresponding epipolar lines become collinear and parallel to one of the image axes, usually the horizontal one. The correspondence search is then reduced to a 1D search along the trivially identified scanline. We will postpone a more detailed description of rectification to Section 6.3.3.

The epipolar geometry can be described analytically in several ways, depending on the amount of the *a priori* knowledge about the stereo system. We can identify three general cases.

- If both *intrinsic* and *extrinsic* camera parameters are known, we can describe the epipolar geometry in terms of the projection matrices (Equation 6.12).
- If only the *intrinsic* parameters are known, we work in normalized coordinates and the epipolar geometry is described by the *essential matrix*.
- If neither intrinsic nor extrinsic parameters are known, the epipolar geometry is described by the *fundamental matrix*.

The essential matrix E. If the intrinsic parameters are known, we can switch to *normalized coordinates* (note that this change of notation will hold throughout this section):

$$\mathbf{m} \leftarrow K^{-1}\mathbf{m} \tag{6.13}$$

Consider a pair of normalized cameras. Without loss of generality, we can fix the world reference frame onto the first camera, hence:

$$P_l = [I|0] \quad \text{and} \quad P_r = [R|\mathbf{t}] \tag{6.14}$$

With this choice, the unknown extrinsic parameters have been made explicit.

If we substitute these two particular instances of the camera projection matrices in Equation (6.12), we get

$$\zeta_r \mathbf{m}_r = \mathbf{t} + \lambda_l R \mathbf{m}_l \tag{6.15}$$

in other words, the point $\mathbf{m}_r$ lies on the line through the points $\mathbf{t}$ and $R\mathbf{m}_l$. In homogeneous coordinates, this can be written as follows:

$$\mathbf{m}_r^T \mathbf{t} \times (R\mathbf{m}_l) = 0 \tag{6.16}$$

as the homogeneous line through two points is expressed as their cross product. Similarly, a dot product of a point and a line is zero if the point lies on the line.

The cross product of two vectors can be written as a product of a skew-symmetric matrix and a vector. Equation (6.16) can therefore be equivalently written as

$$\mathbf{m}_r^T [\mathbf{t}]_\times R \mathbf{m}_l = 0 \tag{6.17}$$

where $[\mathbf{t}]_\times$ is the skew-symmetric matrix of the vector $\mathbf{t}$. If we multiply the matrices in Equation (6.17), we obtain a single matrix which describes the relationship between the corresponding image points $\mathbf{m}_l$ and $\mathbf{m}_r$ in normalized coordinates. This matrix is called the *essential matrix* E:

$$E \triangleq [\mathbf{t}]_\times R \tag{6.18}$$

and the relationship between two corresponding image points in normalized coordinates is expressed by the defining equation for the essential matrix:

$$\mathbf{m}_r^T E \mathbf{m}_l = 0 \tag{6.19}$$

E encodes only information on the extrinsic camera parameters. Its rank is two, since $\det[\mathbf{t}]_\times = 0$. The essential matrix is a homogeneous quantity. It has only five degrees of freedom: a 3D rotation and a 3D translation direction.

The Fundamental Matrix F. The fundamental matrix can be derived in a similar way to the essential matrix. All camera parameters are assumed unknown; we write therefore a general version of Equation (6.14):

$$P_l = K_l[I|0] \quad \text{and} \quad P_r = K_r[R|\mathbf{t}] \tag{6.20}$$

Inserting these two projection matrices into Equation (6.12), we get

$$\zeta_r \mathbf{m}_r = \mathbf{e}_r + \lambda_l K_r R K_l^{-1} \mathbf{m}_l \quad \text{with} \quad \mathbf{e}_r = K_r \mathbf{t} \tag{6.21}$$

which states that point $\mathbf{m}_r$ lies on the line through $\mathbf{e}_r$ and $K_r R K_l^{-1} \mathbf{m}_l$. (It is easy to see that the parameter λ_l is equal to ζ_l, the depth of the point $\mathbf{M}$ with respect to the left camera.) As in the case of the essential matrix, this can be written in homogeneous coordinates as:

$$\mathbf{m}_r^T [\mathbf{e}_r]_\times K_r R K_l^{-1} \mathbf{m}_l = 0 \tag{6.22}$$

The matrix

$$F = [\mathbf{e}_r]_\times K_r R K_l^{-1} \tag{6.23}$$

is the *fundamental matrix* F, giving the relationship between the corresponding image points in pixel coordinates. The defining equation for the fundamental matrix is therefore

$$\mathbf{m}_r^T F \mathbf{m}_l = 0 \tag{6.24}$$

F is the algebraic representation of the epipolar geometry. It is a 3×3, rank-two homogeneous matrix. It has only seven degrees of freedom since it is defined up to a scale and its determinant is zero. Notice that F is completely defined by pixel correspondences only (the intrinsic parameters are not needed).

For any point $\mathbf{m}_l$ in the left image, the corresponding epipolar line $\mathbf{l}_r$ in the right image can be expressed as

$$\mathbf{l}_r = F \mathbf{m}_l \tag{6.25}$$

Similarly, the epipolar line $\mathbf{l}_l$ in the left image for the point $\mathbf{m}_r$ in the right image can be expressed as

$$\mathbf{l}_l = F^T \mathbf{m}_r \tag{6.26}$$

The left epipole $\mathbf{e}_l$ is the right null-vector of the fundamental matrix and the right epipole is the left null-vector of the fundamental matrix:

$$F \mathbf{e}_l = 0 \tag{6.27}$$

$$\mathbf{e}_r^T F = 0 \tag{6.28}$$

One can see from the derivation that the essential and fundamental matrices are related through the camera calibration matrices K_l and K_r:

$$F = K_r^{-T} E K_l^{-1}. \tag{6.29}$$

Further properties of F are discussed in detail in Loung and Faugeras (1996).

Estimating F: the eight-point algorithm. Equation (6.24) applies to any pair of corresponding points $\mathbf{m}_l \leftrightarrow \mathbf{m}_r$. Writing Equation (6.24) as a set of linear constraints, for a sufficient number of correspondences, yields a linear system in the entries of F. This is the essence of the eight-point algorithm.

Each point correspondence gives rise to one linear equation in the unknown entries of F. For example, the equation corresponding to a pair of points $\mathbf{m}_l = (u_l, v_l, 1)$ and $\mathbf{m}_r = (u_r, v_r, 1)$ is

$$u_r u_l f_{1,1} + u_r v_l f_{1,2} + u_r f_{1,3} + v_r u_l f_{2,1} + v_r v_l f_{2,2} + v_r f_{2,3} + u_l f_{3,1} + v_l f_{3,2} + f_{3,3} = 0$$

where $f_{1,1}, f_{1,2}, \ldots, f_{3,3}$ are the unknown entries of F. All available point correspondences form a homogeneous set of equations with the entries of F as the unknowns. Noise generally present in the data suggests a least-squares solution.

This method does not explicitly enforce F to be singular, so it must be done *a posteriori*. It can be shown that the closest singular matrix in Frobenius norm to a given matrix F is the one obtained by forcing to zero the smallest singular value of F. Geometrically, the singularity constraint ensures that the epipolar lines meet in a common epipole.

As Hartley (1995) pointed out, it is crucial for this linear algorithm that input data is properly preconditioned, by a procedure called *standardization*: points are translated so that their centroid is at the origin and are scaled so that their average distance from the origin is $\sqrt{2}$.

In summary, the eight-point algorithm for computation of F is as follows:

- Standardize the input data. Apply the standardizing transformations T_l and T_r, consisting of translation and scaling, to the image coordinates: $\hat{\mathbf{m}}_l^i = T_l \mathbf{m}_l^i$ and $\hat{\mathbf{m}}_r^i = T_r \mathbf{m}_r^i$.
- Linear solution. Compute the fundamental matrix $\hat{F}$ by solving the homogeneous system of equations defined by the point matches $\hat{\mathbf{m}}_l^i \leftrightarrow \hat{\mathbf{m}}_r^i$.
- Enforce the singularity constraint. Replace $\hat{F}$ by $\hat{F}'$, such that $\det \hat{F}' = 0$, by setting the smallest singular value to zero.
- De-standardize the result. The resulting fundamental matrix, associated to the original point correspondences, is $F = T_r^T \hat{F}' T_l$.

This simple algorithm provides good results in many situations and can be used to initialize a variety of more accurate, iterative algorithms. Details of these can be found in Hartley and Zisserman (2003), Torr and Murray (1997) and Zhang (1998).