Now, using the equations

$$\begin{aligned} x &= -(x_{\mathrm{im}} - o_x)s_x, \\ y &= -(y_{\mathrm{im}} - o_y)s_y, \end{aligned}$$

we get

$$\begin{aligned} x_{\mathrm{im}} &= -\frac{f}{s_x}\frac{X_c}{Z_c} + o_x, \\ y_{\mathrm{im}} &= -\frac{f}{s_y}\frac{Y_c}{Z_c} + o_y, \end{aligned}$$

relating the coordinates w.r.t. the camera reference frame to the pixel coordinates. This suggests using the parameters $f_x = f/s_x$ and $f_y = f/s_y$ instead of the parameters $f, s_x, s_y$. In fact, all we need are the parameters $f_x = f/s_x$ and $\alpha = s_y/s_x$, called the *aspect ratio*. Without loss of generality, it can also be assumed that $(o_x, o_y)$ are known. Then we have a total of eight parameters.

One way of solving the calibration problem is to try estimating $f_x, \alpha$, the rotation matrix $R$, and the translation vector $\mathbf{T}$ from $N$ image points $(x_i, y_i)$, projections of $N$ suitably chosen world points $(X_i, Y_i, Z_i)$, using the system of equations obtained from the projection matrix. It turns out that if $N \geq 7$ and the points are not coplanar, the rank of the system is 7, and the system has a nontrivial solution (up to a scalar) that can be found using SVD methods (see Chapter 22, Trucco and Verri [178], or Jain, Katsuri, and Schunck [100]).

Another method consists in estimating the whole projection matrix $M$, which depends on 11 parameters, and then extracting extrinsic and intrinsic parameters. Again, SVD methods are used (see Trucco and Verri [178], and Faugeras [59]).

Cayley's formula can also be used to solve the calibration cameras, as explained in Faugeras [59]. Other problems in computer vision can be reduced to problems in projective geometry (see Faugeras [59]).

In computer graphics, it is also necessary to convert the 3D world coordinates of a point to a two-dimensional representation on a *view plane*. This is achieved by a so-called *viewing system* using a projective transformation. For details on viewing systems see Watt [189] or Foley, van Dam, Feiner, and Hughes [63].

Projective spaces are also the right framework to deal with rational curves and rational surfaces. Indeed, in the projective framework it is easy to deal with vanishing denominators and with "infinite" values of the parameter(s).

It is much less obvious that projective geometry has applications to efficient communication, error-correcting codes, and cryptography, as very nicely explained by Beutelspacher and Rosenbaum [22]. We sketch these applications very briefly, referring our readers to [22] for details. We begin with efficient communication. Suppose that eight students would like to exchange information to do their homework economically. The idea is that each student