Chapter 5

The pinhole camera model

Here we develop a basic camera model. The hardest part of this model is keeping track of the different coordinate systems. Let us therefore summarize them at the outset. First we'll define a camera coordinate system, centred at the focus of the camera with its X and Y axes aligned parallel to, and the Z axis perpendicular to, the image plane. The image plane is also provided with a coordinate system to record the position of features on the image. In practice of course, positions will be measured in pixel coordinates, so ultimately we'll have to make provision to measure in pixel coordinates. The object or scene to be captured is described in terms of a world coordinate system. It is therefore often convenient to fix the world coordinate system to the object or scene.

5.1 Derivation of the camera matrix

This section is about deriving a relationship between the various coordinate systems mentioned above. Written in homogeneous coordinates it is a linear relationship expressed in terms of a matrix called the camera matrix.

5.1.1 Central projection in homogeneous coordinates

Here we consider the central projection of a point $\underline{X} = \begin{bmatrix} X & Y & Z \end{bmatrix}^T$ in the camera coordinate system onto the image plane. The image plane is located at $Z = f$ in the camera coordinate system where f is known as the focal length of the camera. The point where the Z axis pierces the image plane is known as the principal point, and the Z axis is called the principal axis. The origin of the image coordinate system is chosen, for now, as the principal point and its x and y axes are aligned with the X and Y axes of the camera coordinate system. All of this is illustrated in Figure 5.1.

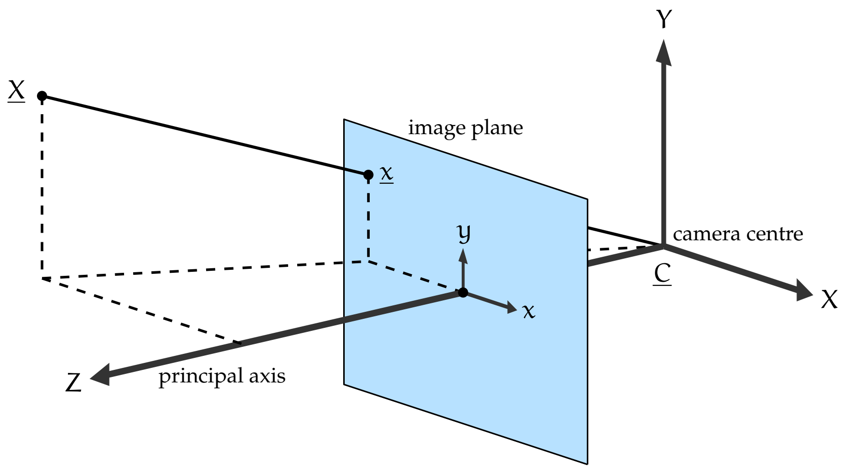


Figure 5.1: Illustrating basic pinhole camera geometry.

If a point $\underline{X} \in \mathbb{R}^3$ has coordinates $\begin{bmatrix} X & Y & Z \end{bmatrix}^T$ relative to the camera coordinate system, $\underline{X}$ projects onto the point $\underline{x}$ on the image plane along the line through $\underline{X}$ and the origin of the camera coordinate system (called the camera centre). Using similar triangles, it follows immediately that $\underline{x} = \begin{bmatrix} f\frac{X}{Z} & f\frac{Y}{Z} \end{bmatrix}^T$ (in Euclidean coordinates). This is a nonlinear map that becomes linear if homogeneous coordinates are used.

Realising that $\begin{bmatrix} f\frac{X}{Z} & f\frac{Y}{Z} \end{bmatrix}^T$ can be written as the homogeneous vector $\begin{bmatrix} fX & fY & Z \end{bmatrix}^T$, the map from homogeneous camera coordinates to homogeneous image coordinates is given by

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}, \tag{5.1}$$

from which we get the Euclidean image coordinates $x = \frac{u}{w}$ and $y = \frac{v}{w}$.

We can describe this central projection from $\underline{X}$ to $\underline{x}$ as

$$\underline{x} = P\underline{X}, \tag{5.2}$$

where P is the 3×4 homogeneous *camera matrix*, also sometimes known as the *projection matrix*. We may write this matrix as

$$P = \text{diag}(f, f, 1)\,[I \mid \underline{0}]\,, \tag{5.3}$$

where

$$\text{diag}(f, f, 1) = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \text{and} \quad [I \mid \underline{0}] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \tag{5.4}$$

The camera matrix derived above assumes that the origin of the image coordinate system is at the principal point $\underline{p}$. However, this is not usually the case in practice.

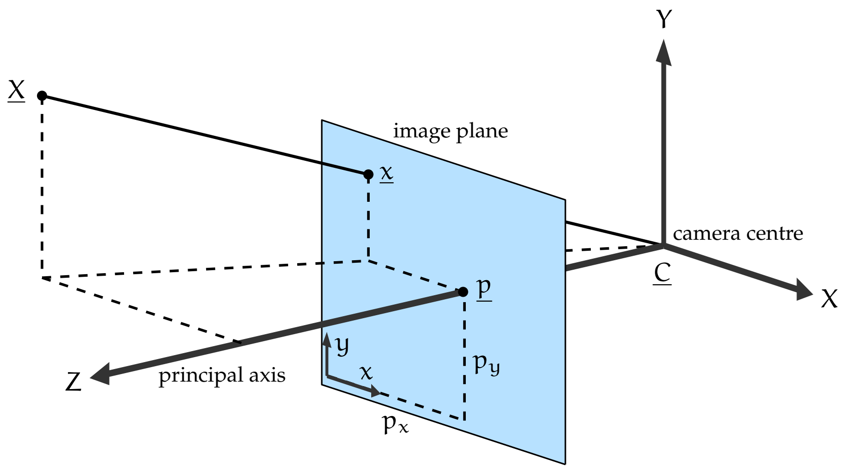


Figure 5.2: Illustrating pinhole camera geometry with offset image coordinates.

In fact, as explained in section 1.6, we would like to think of the origin of our image coordinates as one of the image corners. If the coordinates of the principal point $\underline{p}$ are (p_x, p_y) in the image coordinate system, as depicted in Figure 5.2, then the mapping of $\underline{X}$ to $\underline{x}$ can be expressed as

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad \text{is mapped to} \quad \begin{bmatrix} f\frac{X}{Z} + p_x \\ f\frac{Y}{Z} + p_y \end{bmatrix}, \tag{5.5}$$

or equivalently in homogeneous coordinates as

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} f\,X + Zp_x \\ f\,Y + Zp_y \\ Z \end{bmatrix} = \begin{bmatrix} f & 0 & p_x & 0 \\ 0 & f & p_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}. \tag{5.6}$$

By defining the camera's *calibration matrix* as

$$K = \begin{bmatrix} f & 0 & p_x \\ 0 & f & p_y \\ 0 & 0 & 1 \end{bmatrix}, \tag{5.7}$$

we may write the camera matrix P as

$$P = K\,[I \mid \underline{0}]\,. \tag{5.8}$$

Emphasizing the fact that we are projecting features described in terms of the camera coordinate system, we rewrite the projection as

$$\underline{x} = K\,[I \mid \underline{0}]\,\underline{X}_{\text{cam}}. \tag{5.9}$$

The next step is to introduce the world coordinate system, and relate it to the camera coordinate system.

5.1.2 The world coordinate system

In general, 3D objects are described in terms of coordinate systems fixed to the environment or scene being captured, as indicated in Figure 5.3. Consider a point $\underline{X}$ in world coordinates. Since we already know how to project a point in the camera coordinate system onto the image coordinate system, we only need to relate the world– and camera coordinate systems, i.e. $\underline{X}$ and $\underline{X}_{cam}$. Since the two coordinate systems are related by a rotation and a translation, as is clear from Figure 5.3, we may write

$$\underline{\widetilde{X}}_{cam} = R\big(\underline{\widetilde{X}} - \underline{\widetilde{C}}\big) = R\underline{\widetilde{X}} + \underline{t}, \tag{5.10}$$

where the tilde denotes Euclidean coordinates, so for example

$$\underline{X} = \begin{bmatrix} \underline{\widetilde{X}}^T & 1 \end{bmatrix}^T. \tag{5.11}$$

The Euclidean vector $\underline{\widetilde{C}}$ in (5.10) is the coordinates of the camera centre in the world coordinate system, and R is a 3×3 rotation matrix describing the rotation of the world coordinate system relative to the camera coordinate system.

With $\underline{X}_{cam}$ and $\underline{X}$ the homogeneous representations of $\underline{\widetilde{X}}_{cam}$ and $\underline{\widetilde{X}}$ respectively, we have

$$\underline{X}_{cam} = \begin{bmatrix} R & -R\underline{\widetilde{C}} \\ \underline{0}^T & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} R & -R\underline{\widetilde{C}} \\ \underline{0}^T & 1 \end{bmatrix} \underline{X}. \tag{5.12}$$

Combining this with (5.9), we get

$$\underline{x} = KR\big[I \mid -\underline{\widetilde{C}}\big]\underline{X},$$

where $\underline{X}$ is now given in the world coordinate system.

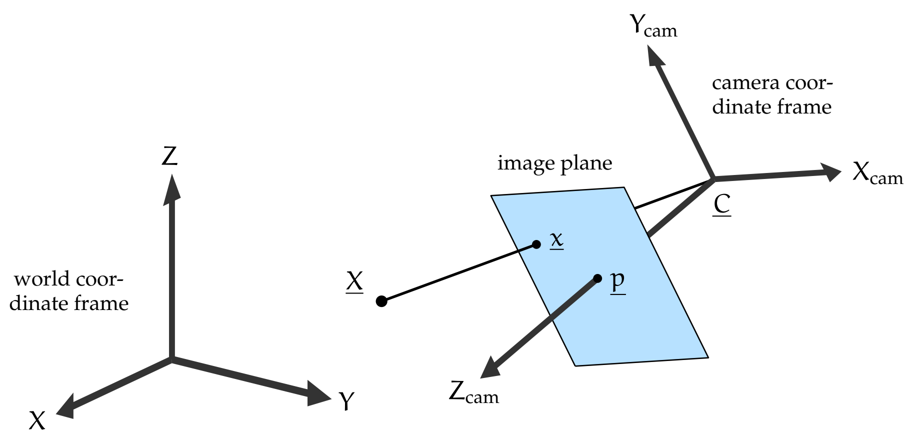


Figure 5.3: Pinhole camera geometry in a general world coordinate system.

Note that all the parameters that refer to the specific type of camera are contained in K; these parameters are referred to as the *intrinsic parameters*. R and $\underline{\widetilde{C}}$ describe the external orientation of the world coordinate system to the camera coordinate system and are therefore referred to as the *extrinsic parameters*.

5.1.3 The general calibration matrix K

Up to now we have assumed that the image coordinate frame is Euclidean with equal scales in both axial directions, and its units are the same as those of the world coordinate system. In most cases we might want to measure world coordinates in something like millimetres, but image coordinates in pixels. To account for this change in unit, we may generalize the calibration matrix to

$$\mathsf{K} = \begin{bmatrix} m_x & 0 & 0 \\ 0 & m_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & p_x \\ 0 & f & p_y \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \alpha_x & 0 & x_0 \\ 0 & \alpha_y & y_0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{5.13}$$

where $\alpha_x = f m_x$ and $\alpha_y = f m_y$ represent the focal length of the camera in terms of pixel coordinates in the x and y directions, respectively. Similarly, (x_0, y_0) is the principal point in terms of pixel coordinates with $x_0 = m_x p_x$ and $y_0 = m_y p_y$.

For even more generality, we may use a calibration matrix of the form

$$\mathsf{K} = \begin{bmatrix} \alpha_x & s & x_0 \\ 0 & \alpha_y & y_0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{5.14}$$

where the added parameter s is referred to as the *skew* parameter.

With this calibration matrix K, our most general camera matrix (corresponding to a finite camera) is

$$\mathsf{P} = \mathsf{KR}\big[\mathsf{I} \mid -\underline{\widetilde{C}}\big], \tag{5.15}$$

and the projection of a point $\underline{X}$ in world coordinates to $\underline{x}$ in image coordinates is

$$\underline{x} = \mathsf{P}\underline{X}. \tag{5.16}$$

We note that the pinhole camera model has eleven degrees of freedom: 5 for the calibration matrix K (the elements α_x, α_y, x_0, y_0 and s), 3 for the rotation matrix R and 3 for the camera centre $\underline{\widetilde{C}}$. This is the same number of degrees of freedom for a 3×4 matrix defined up to a scale.

It follows easily that the 3×3 submatrix KR is non-singular ($\det \mathsf{KR} \neq 0$). Conversely, any 3×4 matrix A for which the left hand 3×3 submatrix is non-singular, is the camera matrix of some finite projective camera, because A can then be decomposed as $A = \mathsf{KR}\big[\mathsf{I} \mid -\underline{\widetilde{C}}\big]$, using a variant of the QR matrix factorization (more on this in section 5.2.2).

5.2 Camera calibration

In this section we discuss some aspects of calculating the camera matrix P and its components for a real camera. Firstly, given a number of point correspondences, we use our linear model to derive a set of equations that generate the matrix P. We then provide a means of decomposing a calculated camera matrix P into its components K, R and $\underline{\widetilde{C}}$.

5.2.1 Basic equations

Here we want to calculate the camera matrix P from n given point correspondences between 3D points $\underline{X}_i$ and 2D image points $\underline{x}_i$, i.e. we want to find a 3×4 camera matrix P such that

$$\underline{x}_i = P\underline{X}_i, \quad i = 1, \ldots, n. \tag{5.17}$$

Since P is a homogeneous matrix, defined only up to an arbitrary scale, it has 11 degrees of freedom. We'll shortly find that each point correspondence gives us two equations. Thus at least $5\frac{1}{2}$ point correspondences are needed to calculate P.

Setting up the necessary equations is just a little tricky since we are working in homogeneous coordinates. The equations in (5.17) are not strict equalities; they mean that $\underline{x}_i$ and $P\underline{X}_i$ are equivalent in the homogeneous sense, that is they point in the same directions but need not have the same magnitudes. This relationship can of course be expressed in terms of the vector cross product as $\underline{x}_i \times P\underline{X}_i = \underline{0}$, which enables us to derive a simple linear solution for P.

If we write the camera matrix in terms of its rows as $P = \begin{bmatrix} \underline{r}_1^T & \underline{r}_2^T & \underline{r}_3^T \end{bmatrix}^T$, then

$$P\underline{X}_i = \begin{bmatrix} \underline{r}_1^T \\ \underline{r}_2^T \\ \underline{r}_3^T \end{bmatrix} \underline{X}_i = \begin{bmatrix} \underline{r}_1^T\underline{X}_i \\ \underline{r}_2^T\underline{X}_i \\ \underline{r}_3^T\underline{X}_i \end{bmatrix}. \tag{5.18}$$

Let $\underline{x}_i = \begin{bmatrix} x_i & y_i & 1 \end{bmatrix}^T$. The vector cross product $\underline{x}_i \times P\underline{X}_i = \underline{0}$ becomes

$$\begin{bmatrix} \underline{0}^T & -\underline{X}_i^T & y_i\underline{X}_i^T \\ \underline{X}_i^T & \underline{0}^T & -x_i\underline{X}_i^T \\ -y_i\underline{X}_i^T & x_i\underline{X}_i^T & \underline{0}^T \end{bmatrix} \begin{bmatrix} \underline{r}_1 \\ \underline{r}_2 \\ \underline{r}_3 \end{bmatrix} = \underline{0}. \tag{5.19}$$

This equation has the form $A_i\underline{r} = \underline{0}$, where A_i is a 3×12 matrix, and $\underline{r}$ is a 12-element column vector made up of the entries of the camera matrix P.

The equation $A_i\underline{r} = \underline{0}$ is a linear equation in the unknown $\underline{r}$. Although there are three rows in (5.19), only two of them are linearly independent (check this yourself!), so that (5.19) reduces to

$$\begin{bmatrix} \underline{0}^T & -\underline{X}_i^T & y_i\underline{X}_i^T \\ \underline{X}_i^T & \underline{0}^T & -x_i\underline{X}_i^T \end{bmatrix} \begin{bmatrix} \underline{r}_1 \\ \underline{r}_2 \\ \underline{r}_3 \end{bmatrix} = \underline{0}. \tag{5.20}$$

From a set of n correspondences we obtain $2n$ equations, and a $2n \times 12$ matrix A by stacking up these $2n$ equations. The camera matrix P has 11 degrees of freedom, so that we need at least five and a half point correspondences to solve for $\underline{r}$ in $A\underline{r} = \underline{0}$.

In practice, the rule of thumb for good estimation of a camera matrix is that the number of measurements should exceed the number of unknowns by a factor 5, so that in this case we would use at least 28 point correspondences and the SVD to find the best (least squares) solution.

A quick word on how to calculate the camera matrix in practice. The easiest way is to construct a calibration object with known dimensions. A cube with a chessboard pattern is very common, or build your own wall from Lego bricks as in Figure 5.4 (Lego bricks are manufactured to high accuracies). Choosing the origin and axes of the world coordinate system conveniently on the object, the corners of the bricks are known in world coordinates. It is now a matter of finding the corresponding points in pixel coordinates. It is a good idea to let the calibration object fill the frame and use as many points as possible.

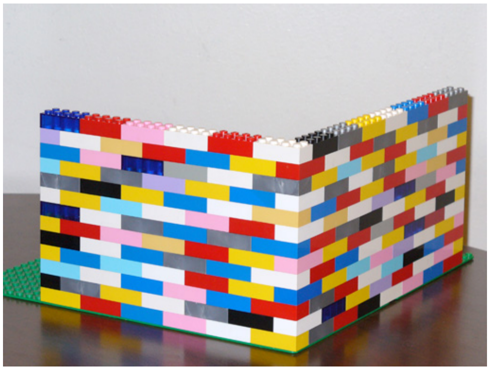


Figure 5.4: A Lego calibration object.

5.2.2 Decomposing the camera matrix

Suppose we've calculated a 3×4 camera matrix P. We know that this matrix is of the form

$$P = KR\left[I \mid -\underline{\widetilde{C}}\right], \tag{5.21}$$

with K the upper-triangular calibration matrix (containing the intrinsic parameters of the camera), and R and $\underline{\widetilde{C}}$ the rotation and translation that relate the camera coordinate frame with the world coordinate frame.

The question we ask now is: how can we extract K, R and $\underline{\widetilde{C}}$ from a given P? It would be useful if we could perform such a decomposition of P. For example, we would then get the intrinsic parameters of the camera, and they will remain constant regardless of subsequent camera motion.

By writing P as

$$P = \left[KR \mid -KR\underline{\widetilde{C}}\right] \tag{5.22}$$

we observe that the first 3 columns of P are in fact KR where K is upper-triangular and R is orthogonal. We immediately think of QR factorization that decomposes a matrix into an orthogonal part and an upper-triangular part. There is a slight problem: QR factorization gives "orthogonal times upper-triangular", but we want "upper-triangular times orthogonal".

Let $P_{1:3}$ be the first 3 columns of P and $\underline{p}_4$ its 4th column. Here is the trick:

1. let $W = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$, and note that $W^{-1} = W^{\mathsf{T}} = W$
2. QR decompose $A = (WP_{1:3})^{\mathsf{T}}$, such that $A = \widehat{Q}\widehat{R}$
3. then let $K = W\widehat{R}^{\mathsf{T}}W$, $R = W\widehat{Q}^{\mathsf{T}}$ and $\underline{\widetilde{C}} = -(P_{1:3})^{-1}\underline{p}_4$.

As an exercise, convince yourself that this procedure is indeed correct. That is to say, show that K is upper-triangular, R is orthogonal, and $KR[I \mid -\underline{\widetilde{C}}] = P$.

We should also keep in mind that QR factorization is unique up to sign. We may fix the signs of the rows and columns of K and R by ensuring that the diagonal entries of K are all positive (as they should be — they represent the focal length of the camera).

5.2.3 Radial distortion

The pinhole camera model assumes that points in world coordinates map linearly (along a straight line) to points on the image plane. This is decidedly untrue for real lenses which introduce nonlinear distortions. The most important nonlinear effect is that of radial distortion. This is where a camera lens loses accuracy towards its edges, which causes straight lines to bend. An example of severe radial distortion is shown in the left hand image of Figure 5.5. Note for example the distortion of the line where the wall meets the ceiling. In the right hand image of the figure the radial distortion has been corrected and the line now appears as being straight.

The idea is that one should first correct an image for radial distortion and then apply the theory derived in the previous sections.

(a) distorted image (b) corrected image

Figure 5.5: Removing radial distortion.

The correction for radial distortion takes place on the image plane. Suppose that the correct, undistorted coordinates are given in pixel coordinates by $\begin{bmatrix}\hat{x} & \hat{y}\end{bmatrix}^T$, and the measured, distorted coordinates by $\begin{bmatrix}x & y\end{bmatrix}^T$. Assuming only radial distortion, we write the relationship between the distorted and undistorted coordinates as

$$\begin{bmatrix}\hat{x} \\ \hat{y}\end{bmatrix} = \begin{bmatrix}x_c \\ y_c\end{bmatrix} + L(r)\begin{bmatrix}x - x_c \\ y - y_c\end{bmatrix}, \tag{5.23}$$

where $r = \sqrt{(x - x_c)^2 + (y - y_c)^2}$ is the distance from the centre $\begin{bmatrix}x_c & y_c\end{bmatrix}^T$ of the radial distortion. Note that the distortion factor $L(r)$ is a function of radius r only.

The distortion factor $L(r)$ is only defined for positive values of r and satisfies $L(0) = 1$. Since the form of $L(r)$ is unknown, we may use a Taylor expansion

$$L(r) = 1 + \kappa_1 r + \kappa_2 r^2 + \kappa_3 r^3 + \cdots,$$

to approximate the distortion factor. An optimization procedure is used to estimate the distortion centre $\begin{bmatrix}x_c & y_c\end{bmatrix}^T$ as well as the Taylor coefficients. The distortion parameters,

$$r_p = \{\kappa_1, \kappa_2, \kappa_3, \ldots, x_c, y_c\}, \tag{5.24}$$

are then considered part of the internal calibration of the camera.

In order to find the distortion parameters (5.24), the common approach is to identify straight lines in the image. Or, rather, lines that are supposed to be straight but are radially distorted in the image. An optimization procedure is used to find suitable distortion parameters that undistort those lines. How many terms in the Taylor expansion one uses depend on the accuracy requirements of the application. Using only κ_1 and setting the rest equal to zero already gives pretty good results.

5.3 More on the camera matrix

Here we take a closer look at the anatomy of P, and ask: what additional information about the camera can we extract from a given camera matrix? Since P is a 3×4 matrix, we can write

$$P = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix} = \begin{bmatrix} \underline{p}_1 & \underline{p}_2 & \underline{p}_3 & \underline{p}_4 \end{bmatrix} = \begin{bmatrix} M \mid \underline{p}_4 \end{bmatrix}, \tag{5.25}$$

where $\underline{p}_i$, $i = 1, 2, 3, 4$, are the columns of P and $M = \begin{bmatrix} \underline{p}_1 & \underline{p}_2 & \underline{p}_3 \end{bmatrix}$.

5.3.1 The null space of P

It follows quite straightforwardly that $\text{rank}(P) = 3$, implying that the dimension of the null space of P equals 1. Since the null space of P is the set of all vectors $\underline{y}$ such that $P\underline{y} = \underline{0}$, and its dimension is 1, it follows that the null space of P consists of a single homogeneous vector. This has special significance since it turns out that this vector is the camera centre. This is easily verified: this, we determine

$$P\underline{C} = KR\begin{bmatrix} I \mid -\underline{\widetilde{C}} \end{bmatrix} \begin{bmatrix} \underline{\widetilde{C}} \\ 1 \end{bmatrix} = KR\left(\underline{\widetilde{C}} - \underline{\widetilde{C}}\right) = \underline{0}. \tag{5.26}$$

Note that in the theoretical case where M (the leftmost 3×3 submatrix of P) is singular, there exists a $\underline{d}$ such that $M\underline{d} = \underline{0}$. This implies that $P\underline{C} = 0$ for $\underline{C} = \begin{bmatrix} \underline{d} & 0 \end{bmatrix}^T$. Therefore, if M is singular, the camera centre is a point at infinity. We call a camera for which M is singular a *camera at infinity*.

5.3.2 The column vectors of P

The column vectors of the camera matrix P reveal information about the orientation of the world coordinate frame.

If $P = \begin{bmatrix} \underline{p}_1 & \underline{p}_2 & \underline{p}_3 & \underline{p}_4 \end{bmatrix}$ then $\underline{p}_1$, $\underline{p}_2$ and $\underline{p}_3$ are the images of the vanishing points of the world X, Y and Z axes, respectively. For example, the homogeneous representation of the vanishing point of the X axis in world coordinates is $\begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}^T$, which maps to

$$P \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \underline{p}_1 & \underline{p}_2 & \underline{p}_3 & \underline{p}_4 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \underline{p}_1. \tag{5.27}$$

Similarly, the vanishing points of the Y and Z axes map to $\underline{p}_2$ and $\underline{p}_3$, respectively. Finally $\underline{p}_4$ is the image of the origin of the world coordinate frame, since

$$P\begin{bmatrix}0\\0\\0\\1\end{bmatrix} = \begin{bmatrix}\underline{p}_1 & \underline{p}_2 & \underline{p}_3 & \underline{p}_4\end{bmatrix}\begin{bmatrix}0\\0\\0\\1\end{bmatrix} = \underline{p}_4. \tag{5.28}$$

There is a lot more to say about the camera matrix, and what it tells us about the camera it is modelling. For example, the row vectors of P are homogeneous representations of certain planes associated with the camera coordinate system. The reader is referred to the excellent text by Hartley and Zisserman[1] for a long and interesting discussion on this topic.

5.4 Back-projection of points to rays

Suppose we are given a point $\underline{x}$ in image coordinates, and we need to determine *all* points $\underline{X}$ that are mapped to $\underline{x}$ with a given camera matrix P. That is, given $\underline{x}$ and P, find all $\underline{X}$ such that

$$\underline{x} = P\underline{X}. \tag{5.29}$$

Before we answer this question, let us first introduce the *pseudo-inverse* of a matrix, which we'll express in terms of the reduced SVD.

The singular value decomposition (SVD) of an $m \times n$ real matrix A is given by

$$A = U\Sigma V^T, \tag{5.30}$$

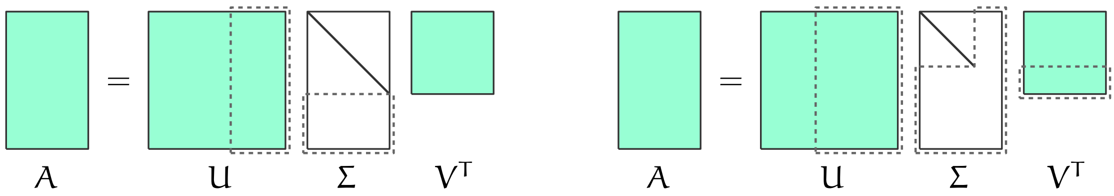
where U and V are orthogonal matrices of sizes $m \times m$ and $n \times n$ respectively. Σ is a diagonal $m \times n$ matrix containing the singular values of A on the main diagonal.

Because of the many zeros in Σ, a number of columns of U or rows of V^T (depending on the shape of A) may be redundant. If for example $m > n$ and A is of full rank, the areas insides the dashed lines in Figure 5.6(a) can always be removed without any loss of information. Let r denote the rank of A. If $r < \min(m, n)$, i.e. A is not of full rank, even more rows and columns can be removed, as illustrated in Figure 5.6(b). This gives the *reduced form* of the SVD.

We can formally state this as

$$A = U_r\Sigma_r V_r^T, \tag{5.31}$$

[1] R. Hartley, A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, 2003.



(a) A has full rank (b) A has less than full rank

Figure 5.6: Removing columns from U and rows from V^T in the reduced form of the SVD.

where Σ_r is an $r \times r$ diagonal matrix containing the first r (i.e. nonzero) singular values of A, U_r is an $m \times r$ matrix consisting of the first r columns of U, and V_r is an $n \times r$ matrix consisting of the first r columns of V.

The reduced SVD can be used to generalize the concept of a matrix inverse, which is defined only for square non-singular matrices. The *generalized inverse* or *pseudo-inverse* of A is denoted by A^+ and can be calculated as

$$A^+ = V_r \Sigma_r^{-1} U_r^T. \tag{5.32}$$

It is straightforward to show that $AA^+ = I$, for any matrix A (except the zero matrix of course). The pseudo-inverse has many applications, including a one-stop solution to linear systems of the form $A\underline{x} = \underline{b}$. The vector $\underline{x} = A^+\underline{b}$ is the exact solution if A is square and non-singular, or the least-squares solution if the system is over-determined and A is square or of full rank. It even offers a natural 'solution' if the system is over-determined and A is rank deficient, or if the system is under-determined.

It turns out that if A has full rank, A^+ can also be calculated as

$$A^+ = A^T(AA^T)^{-1}. \tag{5.33}$$

Let us return to the problem of finding all points that map to a given image point $\underline{x}$. Of course, these are just the points on the line through the camera centre $\underline{C}$ and a version of $\underline{x}$ in world coordinates. It is easily shown that the line

$$\underline{X}(\lambda) = P^+\underline{x} + \lambda\underline{C}, \quad \lambda \in \mathbb{R}, \tag{5.34}$$

projects to $\underline{x}$. To wit,

$$\begin{aligned} P\underline{X}(\lambda) &= P\left(P^+\underline{x} + \lambda\underline{C}\right) \\ &= PP^+\underline{x} + \lambda P\underline{C} \\ &= \underline{x} + \underline{0} \\ &= \underline{x}. \end{aligned}$$

As a side note, one might be tempted to define the line as $\underline{X}(\lambda) = \underline{x} + \lambda\underline{C}$. Why is this wrong?