Camera Calibration

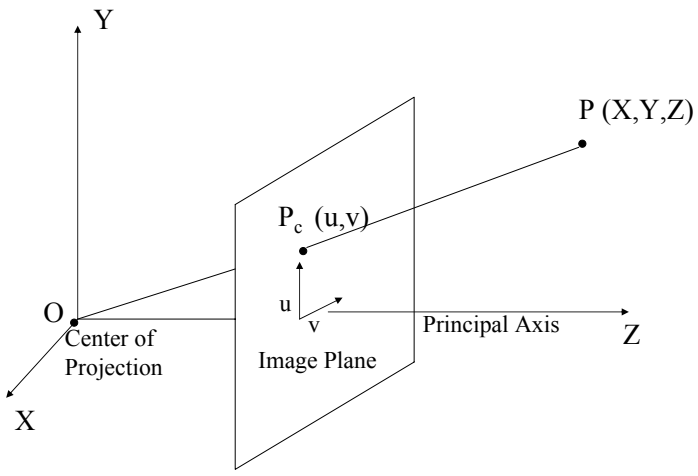

Figure 1: A camera.

Figure 1 shows a camera with center of projection O and the principal axis parallel to Z axis. Image plane is at focus and hence focal length f away from O. A 3D point $P = (X, Y, Z)$ is imaged on the camera's image plane at coordinate $P_c = (u, v)$. We will first find the camera calibration matrix C which maps 3D P to 2D P_c. As we have seen before, we can find P_c using similar triangles as

$$\frac{f}{Z} = \frac{u}{X} = \frac{v}{Y}$$

which gives us

$$u = \frac{fX}{Z}$$

$$v = \frac{fY}{Z}$$

Using homogeneous coordinates for P_c, we can write this as

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

You can verify that this indeed generates the point $P_c = (u, v, w) = (\frac{fX}{Z}, \frac{fY}{Z}, 1)$. Note that P is still not in homogeneous coordinates.

Next, if the origin of the 2D image coordinate system does not coincide with where the Z axis intersects the image plane, we need to translate P_c to the desired origin. Let this translation be defined by (t_u, t_v). Hence, now (u, v) is

$$u = \frac{fX}{Z} + t_u$$

$$v = \frac{fY}{Z} + t_v$$

This can be expressed in a similar form as Equation 1 as

$$\begin{pmatrix} u \\ v \\ w \end{pmatrix} = \begin{pmatrix} f & 0 & t_u \\ 0 & f & t_v \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \tag{2}$$

Now, in Equation 2, P_c is expressed in inches. Since this is a camera image, we need to express it in inches. For this we will need to know the resolution of the camera in pixels/inch. If the pixels are square the resolution will be identical in both u and v directions of the camera image coordinates. However, for a more general case, we assume rectangle pixels with resolution m_u and m_v pixels/inch in u and v direction respectively. Therefore, to measure P_c in pixels, its u and v coordinates should be multiplied by m_u and m_v respectively. Thus

$$u = m_u \frac{fX}{Z} + m_u t_u$$

$$v = m_v \frac{fY}{Z} + m_v t_v$$

This can be expressed in matrix form as

$$\begin{pmatrix} u \\ v \\ w \end{pmatrix} = \begin{pmatrix} m_u f & 0 & m_u t_u \\ 0 & m_v f & m_v t_v \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} \alpha_x & 0 & u_o \\ 0 & \alpha_y & v_o \\ 0 & 0 & 1 \end{pmatrix} P = KP \tag{3}$$

Note that K only depends on the intrinsic camera parameters like its focal length, principal axis and thus defines the intrinsic parameters of the camera. Sometimes K also has a skew parameter s, given by

$$K = \begin{pmatrix} \alpha_x & s & u_o \\ 0 & \alpha_y & v_o \\ 0 & 0 & 1 \end{pmatrix}$$

This usually comes in if the image coordinate axes u and v are not orthogonal to each other. Note that K is an upper triangular 3×3 matrix and P is still not in homogeneous coordinates.

Now if the camera does not have its center of projection at $(0, 0, 0)$ and is oriented in an arbitrary fashion (not necessarily z perpendicular to the image plane), then we need a rotation and translation to make the camera coordinate system coincide with the configuration in Figure 1. Let the camera translation to origin of the XYZ coordinate be given by $-(T_x, T_y, T_z)$ denoted by $-T$ where $T = (T_x, T_y, T_z)$. Let the the rotation applied to coincide the principal axis with Z axis be given by a 3×3 rotation matrix R. Then the matrix formed by first applying the translation followed by the rotation is given by the 4×4 matrix

$$\begin{pmatrix} R & -RT \\ 0 & 1 \end{pmatrix}$$

Thus now, to express the complete transformation, we need to express P in homogeneous coordinates giving

$$P_c = K \begin{pmatrix} R & -RT \\ 0 & 1 \end{pmatrix} P = KR \begin{pmatrix} 1 & -T \\ 0 & 1 \end{pmatrix} P = KR[\, I \quad -T \,]P = CP$$

where $[\, I \quad -T \,]$ is a 4×4 matrix made of the 3×3 identity matrix I and 4×1 vector where T expressed in homogeneous coordinates. Note that the matrix $R[\, I \quad -T \,]$ is a 3×4 matrix that depends solely on the camera's position and orientation. Hence, this defines the extrinsic parameters of the camera. The matrix C obtained by multiplying K with the camera's intrinsic and extrinsic parameters both in a single 3×4 matrix.

Camera Calibration

To fully calibrate a camera, we not only need to know C, but also the breakdown of C to the intrinsic parameters defined by K and the extrinsic parameters defined by R and T. In this section, we will see how to find C and then how to break it up to get the intrinsic and extrinsic parameters. Though C has 12 entries, the entry in the $3rd$ row and $4th$ column is 1. Hence, in effect C has 11 unknown parameters.

Given C, we know that

$$C = [\, KR \;\; -KRT \,]$$

Let $KR = M$. Therefore, the left 3×3 sub matrix of C defines M. WE can use RQ decomposition to break M into two 3×3 one of which is a lower triangular matrix. This lower triangular matrix corresponds to K and the other corresponds to R. The last column of C denoted by c_4 is equivalent to

$$\begin{aligned} -KRT &= c_4 && (4)\\ -MT &= c_4 && (5)\\ T &= -M^{-1}c_4 && (6)\\ & && (7) \end{aligned}$$

Thus, given C, we can find the intrinsic and extrinsic parameters through this process.

The next step is to see how we can find C for any general camera. For this, we need to find *correspondances* between $3D$ points and their images on the camera image. If we know a $3D$ point P_1 corresponding to P_{c_1} on the camera image coordinate, then

$$P_{c_1} = CP_1$$

Or,

$$\begin{pmatrix} u_1 \\ v_1 \\ w_1 \end{pmatrix} = C \begin{pmatrix} X_1 \\ Y_1 \\ Z_1 \\ 1 \end{pmatrix}$$

Note that finding C means we have find all the 11 entries of C. Thus, we are trying to solve for 11 unknowns. Let the rows of C be given by r_i, $i = 1, 2, 3$. Thus

$$C = \begin{pmatrix} r_1 \\ r_2 \\ r_3 \end{pmatrix}$$

Since we know the correspondence P_1 and P_{c_1}, we know

$$u_1 = \frac{r_1.P_1}{r_3.P_1}$$

$$v_1 = \frac{r_2.P_1}{r_3.P_1}$$

which gives us two linear equations

$$u_1(r_3.P_1) - r_1.P_1 = 0$$

$$v_1(r_3.P_1) - r_2.P_1 = 0$$

Note that in these two equations, only the elements of r_1,r_2 and r_3 are the unknowns. So, we find that each $3D$ to $2D$ correspondence generates two linear equations. To solve for 12 unknowns, we will need at least 6 such correspondences. Usually for better accuracy, much more than 6 correspondences are used and the over-determined

system of linear equations thus formed is solved using singular value decomposition methods to generate the 12 entries of C. The correspondences are determined using feducial based image processing methods.

3D Depth Estimation

Now we will see how given the P_c and C, we can find P i.e. using images of 3D world on *calibrated cameras*, how can we estimate the exact location of P. Let us assume that we have a $3D$ point P_1 whose image on a camera defined by the matrix C_1 is given by P_{c_1}. Let the point P be represented in homogeneous coordinates as (X, Y, Z, W).

So, we know

$$P_{c_1} = \begin{pmatrix} u_1 \\ v_1 \\ w_1 \end{pmatrix} = C_1 \begin{pmatrix} X \\ Y \\ Z \\ W \end{pmatrix} \tag{8}$$

The rows of the calibration matrix C_1 are given by $r_i^{C_1}$, $i = 1, 2, 3$. So, from Equation 8, we get two linear equations as

$$u_1(r_3^{C_1}.P) - r_1^{C_1}.P = 0$$

$$v_1(r_3^{C_1}.P) - r_2^{C_1}.P = 0$$

So, from each camera we can generate two linear equations. We have 4 unknowns to be solved given by X, Y, Z, W. Thus, we need at least two camera (with different calibration matrices) and we need to find the point P_{c_2} on this camera's image that corresponds to the same $3D$ point P. Finding the image of the same $3D$ points on two different cameras images is a hard problem. This is the reason that humans need two eyes to resolve depth. Also, note that this mathematics only takes into account the binocular cues like disparity. The reason we humans can still resolve depth to a certain extent with one eye, is because we use several oculomotor and monocular cues. These are not present for a camera and hence depth estimation is not possible with a single camera. Of course, for greater accuracy often more than two cameras are used (called stereo rigs) and singular value decomposition is used to solve the over-determined linear equations that result.

Homography

If two cameras see points lying on a plane, a relationship between them can be easily found without going through explicit camera calibration. This relationship that relates the two cameras is called the *homography*.

Figure 2 illustrates the situation. Let us assume a point P_π that lies on the plane π. Let the plane be defined by the vector $\pi = (\frac{a}{d}\ \frac{b}{d}\ \frac{c}{d}\ 1)$. Let $N = (\frac{a}{d}\ \frac{b}{d}\ \frac{c}{d})$ be the 1×3 row vector defining the normal to the plane π. Thus the plane equation is given by

$$(\ N\ \ 1\ ).P = 0 \tag{9}$$

where P is any point in the $3D$ world.

Let the two cameras be defined by calibration matrices by C_1 and C_2. Let the image of P_π on camera C_1 and C_2 be P_π^1 and P_π^2 respectively. From this, we know that

$$P_\pi^1 = \begin{pmatrix} u_1 \\ v_1 \\ w_1 \end{pmatrix} = C_1.P_\pi$$

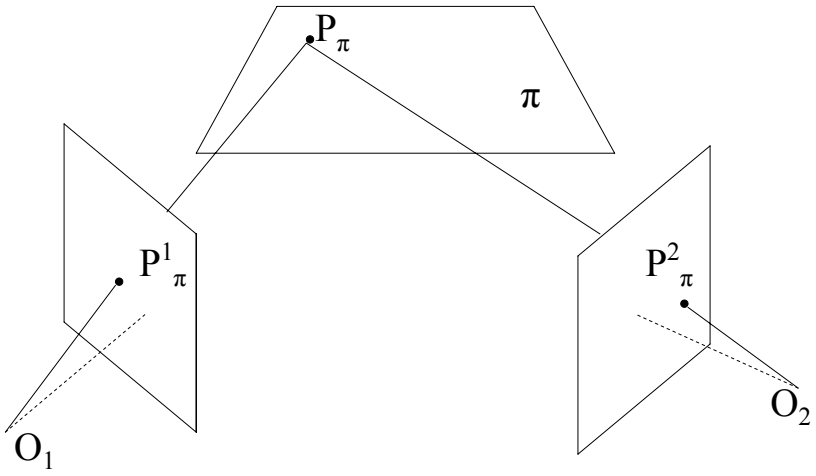

Figure 2: Homography between two cameras seeing a scene.

This means that in $3D$, the point P_π lies on the ray $(u_1, v_1, w_1, 0)^T$. However, the scale factor is unknown. Let this unknown scale factor be denoted by τ. Then we get

$$P_\pi = \begin{pmatrix} u_1 \\ v_1 \\ w_1 \\ \tau \end{pmatrix}$$

Next, since P_π satisfies the plane equation, we get from Equation 9,

$$\tau = -N.P_\pi^1$$

Now, let $C_2 = (\ A_2 \quad a_2\ )$, where A_2 is the 3×3 matrix and a_2 is a 3×1 vector. Then,

$$\begin{aligned} P_\pi^2 &= C_2.P_\pi && (10)\\ &= (\ A_2 \quad a_2\ ) \begin{pmatrix} P_\pi^1 \\ -NP_\pi^1 \end{pmatrix} && (11)\\ &= (\ A_2 \quad a_2\ ) \begin{pmatrix} 1 \\ -N \end{pmatrix} P_\pi^1 && (12)\\ &= (A_2 - a_2N)P_\pi^1 && (13)\\ &= H\ P_\pi^1 && (14)\\ & && (15) \end{aligned}$$

Note that a_2 is a 3×1 matrix and N is a 1×3 matrix. Thus, a_2N would generate a 3×3 matrix that can be subtracted from 3×3 matrix A_2 to generate H. Thus, H is a 3×3 matrix that relates one camera image with another and defines the homography. Using this matrix, the image from one camera can be warped to the view of another camera.

H has 9 parameters. As with C, in H also the element in 3rd row and 3rd column is 1. Hence, in effect H has eight parameters. In order to reconstruct H, we have to see a plane with two cameras. We have to find the image of same point on the plane as seen by two cameras. So, basically we need to know the corresponding points P_π^1 and P_π^2 in the two cameras. From each correspondence, using Equation 14, we can generate two linear equations.

To find the 8 unknowns in H, we need just 4 correspondences. So, now instead of going through a full camera calibration (finding 11 parameters for two cameras leading to 22 parameters), with these 8 homography parameters we can relate one camera with another.