Course Notes 2: Camera Calibration *

1 Camera Calibration

To precisely know the transformation from the real, 3D world into digital images requires prior knowledge of many of the camera's intrinsic parameters. If given an arbitrary camera, we may or may not have access to these parameters. We do, however, have access to the images the camera takes. Therefore, can we find a way to deduce them from images? This problem of estimating the extrinsic and intrinsic camera parameters is known as **camera calibration**.

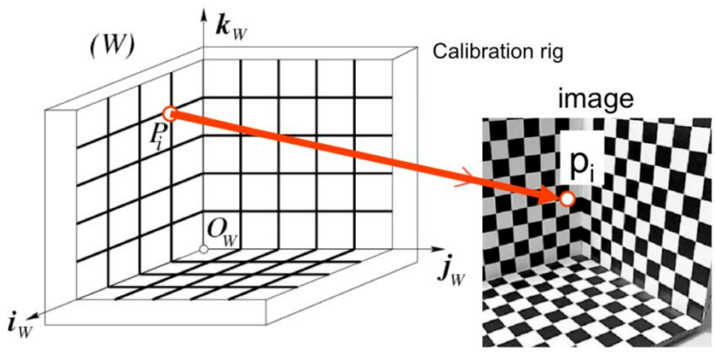

Figure 1: The setup of an example calibration rig.

Specifically, we do this by solving for the intrinsic camera matrix K and the extrinsic parameters R, T from Equation **??**. We can describe this problem in the context of a calibration rig, such as the one show in Figure 1. The rig usually consists of a simple pattern (i.e. checkerboard) with known dimensions. Furthermore, the rig defines our world reference frame with origin O_w and axes i_w, j_w, k_w. From the rig's known pattern, we have known points in the world reference frame $P_1, ..., P_n$. Finding these points in the image we take from the camera gives corresponding points in the image $p_1, ..., p_n$.

We set up a linear system of equations from n correspondences such that for each correspondence P_i, p_i and camera matrix M whose rows are m_1, m_2, m_3:

$$p_i = \begin{bmatrix} u_i \\ v_i \end{bmatrix} = MP_i = \begin{bmatrix} \frac{m_1 P_i}{m_3 P_i} \\ \frac{m_2 P_i}{m_3 P_i} \end{bmatrix} \tag{1}$$

As we see from the above equation, each correspondence gives us two equations and, consequently, two constraints for solving the unknown parameters contained in m. From

*Most contents are from

- R. Hartley and A. Zisserman. Multiple View Geometry in Computer Vision (2nd Edition)
- K. Hata and S. Savarese. Course notes of Stanford CS231A

before, we know that the camera matrix has 11 unknown parameters. This means that we need at least 6 correspondences to solve this. However, in the real world, we often use more, as our measurements are often noisy. To explicitly see this, we can derive a pair of equations that relate u_i and v_i with P_i.

$$\begin{aligned} u_i(m_3P_i) - m_1P_i &= 0 \\ v_i(m_3P_i) - m_2P_i &= 0 \end{aligned}$$

Given n of these corresponding points, the entire linear system of equations becomes

$$\begin{aligned} u_1(m_3P_1) - m_1P_1 &= 0 \\ v_1(m_3P_1) - m_2P_1 &= 0 \\ &\vdots \\ u_n(m_3P_n) - m_1P_n &= 0 \\ v_n(m_3P_n) - m_2P_n &= 0 \end{aligned}$$

This can be formatted as a matrix-vector product shown below:

$$\begin{bmatrix} P_1^T & 0^T & -u_1P_1^T \\ 0^T & P_1^T & -v_1P_1^T \\ & \vdots & \\ P_n^T & 0^T & -u_nP_n^T \\ 0^T & P_n^T & -v_nP_n^T \end{bmatrix} \begin{bmatrix} m_1^T \\ m_2^T \\ m_3^T \end{bmatrix} = \mathbf{P}m = 0 \tag{2}$$

When $2n > 11$, our homogeneous linear system is overdetermined. For such a system $m = 0$ is always a trivial solution. Furthemore, even if there were some other m that were a nonzero solution, then $\forall k \in \mathbb{R}, km$ is also a solution. Therefore, to constrain our solution, we complete the following minimization:

$$\begin{aligned} \underset{m}{\text{minimize}} \quad & \|\mathbf{P}m\|^2 \\ \text{subject to} \quad & \|m\|^2 = 1 \end{aligned} \tag{3}$$

To solve this minimization problem, we simply use singular value decomposition. If we let $P = UDV^T$, then the solution to the above minimization is to set m equal to the last column of V. The derivation for this solution is outside the scope of this class and you may refer to Section 5.3 of Hartley & Zisserman on pages 592-593 for more details.

After reformatting the vector m into the matrix M, we now want to explicitly solve for the extrinsic and intrinsic parameters. We know our SVD-solved M is known up to scale, which means that the true values of the camera matrix are some scalar multiple of M:

$$\rho M = \begin{bmatrix} \alpha r_1^T - \alpha \cot\theta r_2^T + c_x r_3^T & \alpha t_x - \alpha\cot\theta t_y + c_x t_z \\ \frac{\beta}{\sin\theta} r_2^T + c_y r_3^T & \frac{\beta}{\sin\theta} t_y + c_y t_z \\ r_3^T & t_z \end{bmatrix} \tag{4}$$

Here, r_1^T, r_2^T, and r_3^T are the three rows of R. Dividing by the scaling parameter gives

$$M = \frac{1}{\rho}\begin{bmatrix} \alpha r_1^T - \alpha \cot\theta r_2^T + c_x r_3^T & \alpha t_x - \alpha\cot\theta t_y + c_x t_z \\ \frac{\beta}{\sin\theta} r_2^T + c_y r_3^T & \frac{\beta}{\sin\theta} t_y + c_y t_z \\ r_3^T & t_z \end{bmatrix} = \begin{bmatrix} A & b \end{bmatrix} = \begin{bmatrix} a_1^T \\ a_2^T \\ a_3^T \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

Solving for the intrinsics gives

$$\begin{aligned}
\rho &= \pm\frac{1}{\|a_3\|}\\
c_x &= \rho^2(a_1 \cdot a_3)\\
c_y &= \rho^2(a_2 \cdot a_3)\\
\theta &= \cos^{-1}\left(-\frac{(a_1 \times a_3)\cdot(a_2 \times a_3)}{\|a_1 \times a_3\| \cdot \|a_2 \times a_3\|}\right)\\
f_x &= \rho^2\|a_1 \times a_3\| \sin\theta\\
f_y &= \rho^2\|a_2 \times a_3\| \sin\theta
\end{aligned} \tag{5}$$

The extrinsics are

$$\begin{aligned}
r_1 &= \frac{a_2 \times a_3}{\|a_2 \times a_3\|}\\
r_2 &= r_3 \times r_1\\
r_3 &= \rho a_3\\
T &= \rho K^{-1} b
\end{aligned} \tag{6}$$

We leave the derivations as a class exercise or you can refer to Section 1.3.1 of the Forsyth & Ponce textbook.

With the calibration procedure complete, we warn against degenerate cases. Not all sets of n correspondences will work. For example, if the points P_i lie on the same plane, then the system will not be able to be solved. These unsolvable configurations of points are known as **degenerate configurations**. More generally, degenerate configurations have points that lie on the intersection curve of two quadric surfaces. Although this outside the scope of the class, you can find more information in Section 1.3 of the Forsyth & Ponce textbook.

2 Additional reading: Handling Distortion in Camera Calibration

So far, we have been working with ideal lenses which are free from any distortion. However, as seen before, real lenses can deviate from rectilinear projection, which require more advanced methods. This section provides just a brief introduction to handling distortions.

Often, distortions are radially symmetric because of the physical symmetry of the lens. We model the radial distortion with an isotropic transformation:

$$QP_i = \begin{bmatrix} \frac{1}{\lambda} & 0 & 0 \\ 0 & \frac{1}{\lambda} & 0 \\ 0 & 0 & 1 \end{bmatrix} MP_i = \begin{bmatrix} u_i \\ v_i \end{bmatrix} = p_i \tag{7}$$

If we try to rewrite this into a system of equations as before, we get

$$\begin{aligned}
u_i q_3 P_i &= q_1 P_i\\
v_i q_3 P_i &= q_2 P_i
\end{aligned}$$

This system, however, is no longer linear, and we require the use of nonlinear optimization techniques, which are covered in Section 22.2 of Forsyth & Ponce. We can simplify the nonlinear optimization of the calibration problem if we make certain assumptions. In radial distortion, we note that the ratio between two coordinates u_i and v_i is not affected. We can compute this ratio as

$$\frac{u_i}{v_i} = \frac{\frac{m_1 P_i}{m_3 P_i}}{\frac{m_2 P_i}{m_3 P_i}} = \frac{m_1 P_i}{m_2 P_i} \tag{8}$$

Assuming that n correspondences are available, we can set up the system of linear equations:

$$\begin{gathered} v_1(m_1 P_1) - u_1(m_2 P_1) = 0 \\ \vdots \\ v_n(m_1 P_n) - u_n(m_2 P_n) = 0 \end{gathered}$$

Similar to before, this gives a matrix-vector product that we can solve via SVD:

$$Ln = \begin{bmatrix} v_1 P_1^T & -u_1 P_1^T \\ \vdots & \vdots \\ v_n P_n^T & -u_n P_n^T \end{bmatrix} \begin{bmatrix} m_1^T \\ m_2^T \end{bmatrix} \tag{9}$$

Once m_1 and m_2 are estimated, m_3 can be expressed as a nonlinear function of m_1, m_2, and λ. This requires to solve a nonlinear optimization problem whose complexity is much simpler than the original one.