Lecture 04/08

April 8, 2020

1 Estimating P

Recall from the last lecture the problem setup. We want to estimate the camera projection matrix P. To do so, we have a few 3D world points, $\vec{\mathbf{Q}}_1, \ldots, \vec{\mathbf{Q}}_n$, and their corresponding image locations $\vec{\mathbf{q}}_1, \ldots, \vec{\mathbf{q}}_n$. This gives us the following set of equations:

$$\vec{\mathbf{q}}_i \equiv P\vec{\mathbf{Q}}_i \;\; \forall i \tag{1}$$

In addition, because this set of equations still leaves a free scale factor (that is, if P is a solution, so is αP, $\alpha \neq 0$), we added an additional constraint.

$$\|P\|_F = 1 \tag{2}$$

We showed that each of the equivalences in Equation 1 result in *two* linear equations in the entries of P. By writing the vector of unknowns as $\mathbf{p} = \begin{bmatrix} P_{11} & P_{12} & \ldots & P_{34} \end{bmatrix}^T$, we wrote this into the following system:

$$A\mathbf{p} = \mathbf{0} \tag{3}$$
$$\|\mathbf{p}\| = 1 \tag{4}$$

Because the 2D image locations $\vec{\mathbf{q}}_i$ and the corresponding world locations $\vec{\mathbf{Q}}_i$ might have a little bit of noise, the resulting set of equations may not have a solution. We therefore relax this system into the following optimization problem:

$$\min_{\mathbf{p}} \|A\mathbf{p}\|_2^2 \tag{5}$$
$$\text{such that} \tag{6}$$
$$\|\mathbf{p}\| = 1 \tag{7}$$

1.1 Solution using SVD

How do we solve this problem? We can leverage the singular value decomposition. Every matrix has a singular value decomposition: this expresses the matrix as a product of an orthonormal matrix, a diagonal matrix, and another orthonormal matrix. Thus, the matrix A can always be written as:

$$A = U\Sigma V^T \tag{8}$$

where U and V are orthonormal and Σ is diagonal.

We will also use the following fact about orthonormal matrices. If $\mathbf{x}$ is a vector and R is an orthonormal matrix, then:

$$\|R\mathbf{x}\|^2 = (R\mathbf{x})^T(R\mathbf{x}) \tag{9}$$
$$= \mathbf{x}^T R^T R\mathbf{x} \tag{10}$$
$$= \mathbf{x}^T\mathbf{x} \quad \because R^T R = I \tag{11}$$
$$= \|\mathbf{x}\|^2 \tag{12}$$

Thus, multiplication by an orthonormal matrix does not change the norm of a vector.

Using these two facts, we can see that:

$$\|A\mathbf{p}\|^2 = \|U\Sigma V^T\mathbf{p}\|^2 = \|\Sigma V^T\mathbf{p}\|^2 \quad \because U^TU = I \tag{13}$$

So we have:

$$\min_{\mathbf{p}} \|\Sigma V^T\mathbf{p}\|_2^2 \tag{14}$$

$$\text{such that} \tag{15}$$

$$\|\mathbf{p}\| = 1 \tag{16}$$

Now let $\mathbf{y} = V^T\mathbf{p} \Leftrightarrow \mathbf{p} = V\mathbf{y}$. Again, because V is orthonormal, $\|\mathbf{p}\|_2 = \|\mathbf{y}\|_2$. So expressed in terms of $\mathbf{y}$, we get the following optimization problem:

$$\min_{\mathbf{y}} \|\Sigma\mathbf{y}\|_2^2 \tag{17}$$

$$\text{such that} \tag{18}$$

$$\|\mathbf{y}\| = 1 \tag{19}$$

Now Σ is a diagonal matrix, i.e., it takes the form $\begin{bmatrix} \sigma_1 & 0 & \dots & 0 \\ 0 & \sigma_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \sigma_{12} \end{bmatrix}$. Thus $\Sigma\mathbf{y} = \begin{bmatrix} \sigma_1 y_1 \\ \sigma_2 y_2 \\ \vdots \\ \sigma_{12} y_{12} \end{bmatrix}$. Substituting this into the optimization problem and expanding, we have:

$$\min_{\mathbf{y}} \sum_{i=1}^{12} \sigma_i^2 y_i^2 \tag{20}$$

$$\text{such that} \tag{21}$$

$$\sqrt{\sum_{i=1}^{12} y_i^2} = 1 \tag{22}$$

Thus, y_i^2 (all non-negative numbers) should all sum to 1, but when weighted by σ_i^2, they should sum to as low a number as possible. Let us assume that the indices are ordered such that $\sigma_1 \leq \sigma_2 \leq \ldots \leq \sigma_{12}$. The only way to perform this minimization is to put all the mass on the index with the smallest σ. Thus, we want $y_1 = 1, y_i = 0 \forall i \neq 1$.

From this, we can now recover $\mathbf{p}$ and reshape it into our 3×4 camera projection matrix P.

1.2 How many points do we need?

The procedure outlined above and the previous class require a set of world points $\vec{\mathbf{Q}}_i$ and corresponding image locations $\vec{\mathbf{q}}_i$. How many such points do we need?

To answer this, first observe that P is a 3×4 matrix, so it has 12 entries. Setting $\|P\|_F = 1$ already gives us one equation, so we need 11 more. Each point gives us two linear equations (one in the x coordinate and the other from the y coordinate). This means that we need **at least 6 points**. However, there is a caveat: for the equations to be linearly independent, we need to ensure that these points are in general position, i.e., no five of them are coplanar.

What does this look like in practice? Figure **??** shows an example: the cyan points might be the ones we use for camera calibration.

2 Recovering K, R and $\mathbf{t}$

We know that $P = K\begin{bmatrix} R & \mathbf{t} \end{bmatrix}$. Now that we know P, we want to recover K, R and $\mathbf{t}$. How do we do this?

Figure 1: Input for camera calibration. For the cyan points, we also know corresponding world coordinate locations.

The first thing to realize is that we cannot let K be an arbitrary 3×3 matrix. If K can be arbitrary, then there are multiple possible solutions. In particular, if $K, R, \mathbf{t}$ represent one solution, and R' is another rotation matrix, then $KR'^T, R'R, R'\mathbf{t}$ is also a solution:

$$KR'^T \begin{bmatrix} R'R & R'\mathbf{t} \end{bmatrix} = K \begin{bmatrix} R'^T R'R & R'^T R'\mathbf{t} \end{bmatrix} = K \begin{bmatrix} R & \mathbf{t} \end{bmatrix} \tag{23}$$

However, when we derived the expression for camera projection, we actually had a specific operation in mind. In particular, K was only supposed to (a) translate the origin of the image coordinate system, and (b) scale to change units. A few lectures ago, K was defined to look something like this:

$$K = \begin{bmatrix} f & 0 & u \\ 0 & f & v \\ 0 & 0 & 1 \end{bmatrix}. \tag{24}$$

Observe that this is an upper triangular matrix. We can therefore use this as a constraint.

Thus, our task is to decompose P as $K \begin{bmatrix} R & \mathbf{t} \end{bmatrix}$, where K is an *upper triangular* matrix, and R is an orthonormal matrix. Observe that the first 3×3 sub-matrix of P must be KR, and the last column must be $K\mathbf{t}$.

Here, yet another matrix decomposition comes to our rescue: the RQ decomposition. This allows us to decompose any matrix (in our case, the first 3×3 sub-matrix of P) into the product of an upper-triangular matrix and an orthonormal matrix. We can use this out of the box to get K and R. Given these, it is then trivial to derive $\mathbf{t}$ by using the fact that the last column of P must be $K\mathbf{t}$.

3 Nuances and caveats

This approach to estimating the camera parameters is called the *Direct Linear Transform*, because it converts everything into a linear system. This is not the best way of estimating these parameters, but it is the simplest: it draws on fairly well known and standard techniques in linear algebra.