Administrivia

I am willing to meet **Wednesday afternoon** between 2:00 and 5:00 regarding projects (send me E-mail).

Syllabus Notes

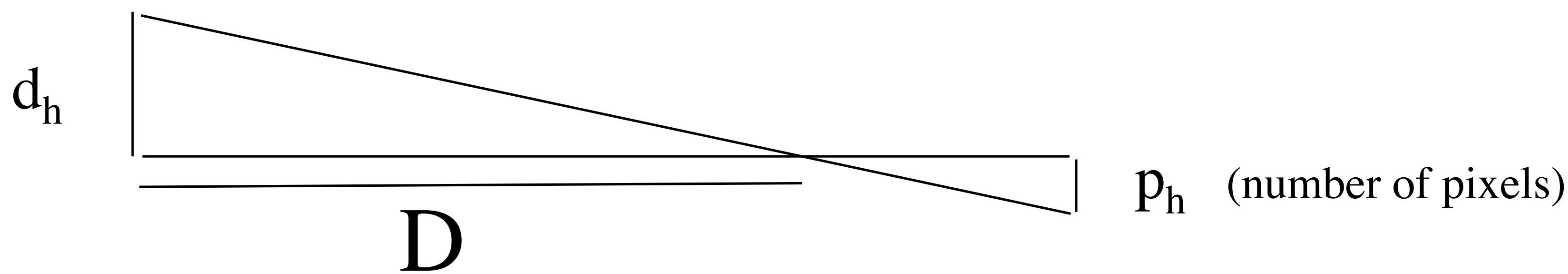
As mentioned via E-mail, we are doing camera calibration, and learning two forms of least squares on route. (§2.2.1, §2.2.2, §3.1.1, §3.2).

We will be using an eigenvalue decomposition next lecture. The details will be optional. However, you may find a quick review of that part of your linear algebra book will make the argument a little less mysterious.

Camera parameters (§2.2)

- Extrinsic parameters
  - position of the camera (3)
  - and orientation of the camera (3)
- Intrinsic parameters
  - focal length (1)
  - principal point (intersection of viewing direction with camera plane) (2)
  - aspect ratio (ratio of size of horizontal pixel sizes to vertical ones (1)
  - angle between axes of image plane---usually very close to 90 degrees (1)

Common to rework focal length and aspect ration into two other parameters giving the horizontal and vertical magnifications □ and □ (easy to measure)---which is basically the focal length for the vertical or horizontal directions measured in units of the width of a pixel.



$$\square = D \frac{p_h}{d_h}$$

$$\begin{pmatrix} U \\ V \\ W \end{pmatrix} = \underbrace{\begin{pmatrix} \text{Transformation} \\ \text{representing} \\ \text{intrinsic parameters} \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} \text{Transformation} \\ \text{representing} \\ \text{extrinsic parameters} \end{pmatrix}}_{\text{Camera matrix, } M} \begin{pmatrix} X \\ Y \\ Z \\ T \end{pmatrix}$$

Camera matrix, *M*

Goal one: find M from image of calibration object

Goal two: given M, find the two matrices (argued last time that under favorable conditions, this should be possible because M is 3 by 4, and there are 11 parameters)

Typical setup for calibration

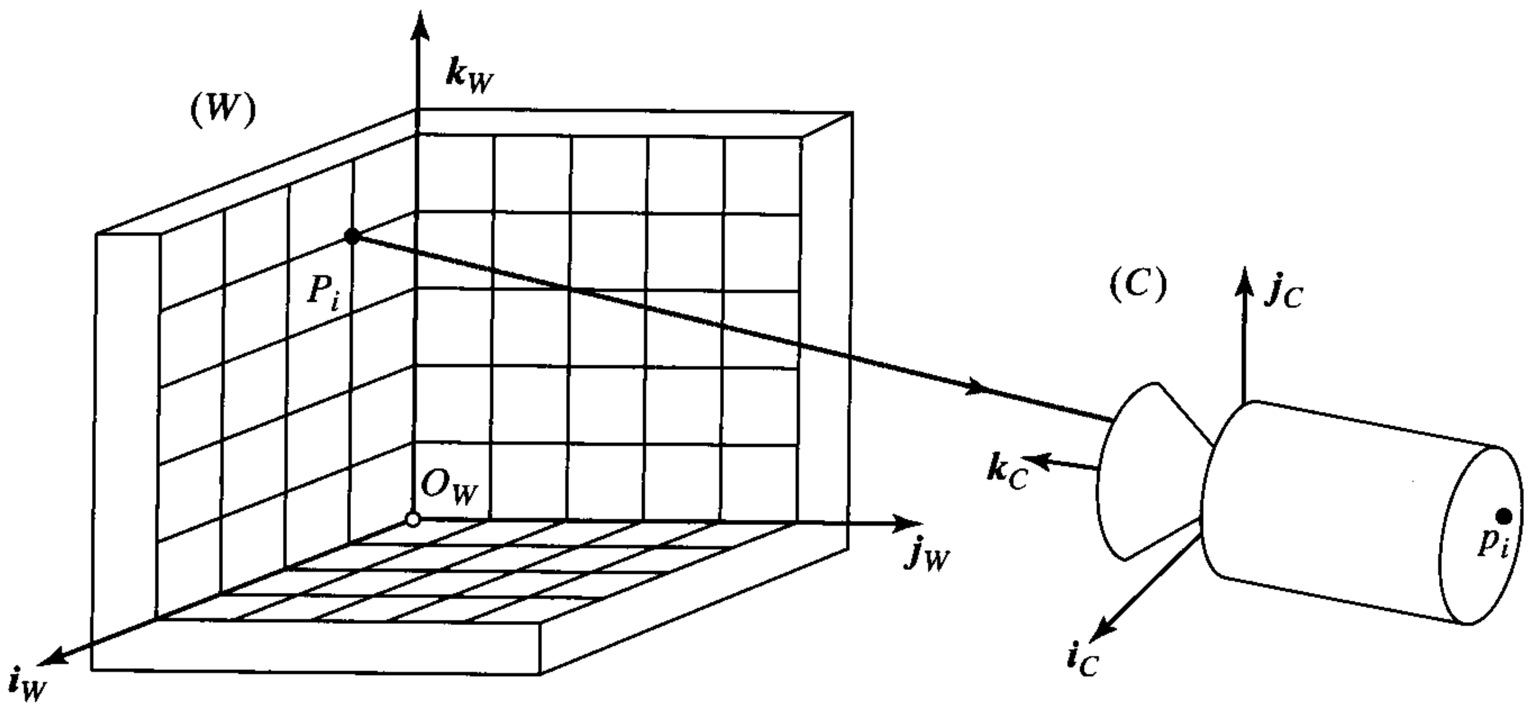


Figure 3.1 Camera calibration setup: In this example, the calibration rig is formed by three grids drawn in orthogonal planes. Other patterns could be used as well, and they may involve lines or other geometric figures.

Goal one: Find M from image of calibration object. The equation relating world coordinates to image coordinates is:

$$\begin{bmatrix} U \\ V \\ W \end{bmatrix} = M \begin{bmatrix} X \\ Y \\ Z \\ T \end{bmatrix} = MP$$

If we identify enough non-degenerate points whose *world coordinates are known* then we can estimate M from their *location in the image*.

The above is the equation in terms of homogeneous coordinates. So we have to work in terms of the observed image coordinates, u=U/W and v=V/W

(§2.2.2, §3.2.1)

Write $M = \begin{bmatrix} \mathbf{m}_1^T \\ \mathbf{m}_2^T \\ \mathbf{m}_3^T \end{bmatrix}$

Note that the matrix is the unknown so we want to make it a vector in some matrix equation (where something else is going to be the matrix)---standard thing to do.

We have

$$U = \mathbf{m_1} \cdot \mathbf{P}$$
$$V = \mathbf{m_2} \cdot \mathbf{P}$$
$$W = \mathbf{m_3} \cdot \mathbf{P}$$

So each point, i, gives two equations (§2.2.2, §3.2.1)

$$u_i = \frac{\mathbf{m}_1 \cdot \mathbf{P}_i}{\mathbf{m_3} \cdot \mathbf{P}_i} \qquad v_i = \frac{\mathbf{m}_2 \cdot \mathbf{P}_i}{\mathbf{m_3} \cdot \mathbf{P}_i}$$

Which become

$$(\mathbf{m_1} - u_i \mathbf{m_3}) \cdot \mathbf{P}_i = 0$$
$$(\mathbf{m}_2 - v_i \mathbf{m_3}) \cdot \mathbf{P}_i = 0$$

(§2.2.2, §3.2.1)

We have linear equations for the components of M

The components of the matrix M are the *variables* in linear equations

Represent M by a vector $\mathbf{m} = \begin{bmatrix} \mathbf{m_1} \\ \mathbf{m}_2 \\ \mathbf{m_3} \end{bmatrix}$

Now rewrite $(\mathbf{m_1} - u_i \mathbf{m_3}) \cdot \mathbf{P}_i = 0$ as $\begin{pmatrix} \mathbf{P}_i^T & 0 & -u_i \mathbf{P}_i^T \end{pmatrix} \mathbf{m} = 0$

$(\mathbf{m}_2 - v_i \mathbf{m_3}) \cdot \mathbf{P}_i = 0$ as $\begin{pmatrix} 0 & \mathbf{P}_i^T & -v_i \mathbf{P}_i^T \end{pmatrix} \mathbf{m} = 0$

We have linear equations for the components of M

The components of the matrix M are the *variables* in linear equations

Represent M by a vector $\mathbf{m} = \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \mathbf{m}_3 \end{bmatrix}$

Now rewrite $(\mathbf{m}_1 - u_i \mathbf{m}_3) \cdot \mathbf{P}_i = 0$ as $\begin{pmatrix} \mathbf{P}_i^T & 0 & -u_i \mathbf{P}_i^T \end{pmatrix} \mathbf{m} = 0$

$(\mathbf{m}_2 - v_i \mathbf{m}_3) \cdot \mathbf{P}_i = 0$ as $\begin{pmatrix} 0 & \mathbf{P}_i^T & -v_i \mathbf{P}_i^T \end{pmatrix} \mathbf{m} = 0$

Thus every point leads to two rows of a matrix P.

(§2.2.2, §3.2.1)

From previous slide, each point gives two rows of a matrix P

$$\begin{pmatrix} \mathbf{P}_i^T & 0 & -u_i\mathbf{P}_i^T \end{pmatrix} \mathbf{m} = 0$$

$$\begin{pmatrix} 0 & \mathbf{P}_i^T & -v_i\mathbf{P}_i^T \end{pmatrix} \mathbf{m} = 0$$

So, in general, the 2n by 12 matrix P is:

$$\begin{bmatrix} \mathbf{P}_1^T & & -u_1\mathbf{P}_1^T \\ & \mathbf{P}_1^T & -v_1\mathbf{P}_1^T \\ \cdots & \cdots & \cdots \\ \mathbf{P}_i^T & & -u_i\mathbf{P}_i^T \\ & \mathbf{P}_i^T & -v_i\mathbf{P}_i^T \\ \cdots & \cdots & \cdots \\ \mathbf{P}_n^T & & -u_n\mathbf{P}_n^T \\ & \mathbf{P}_n^T & -v_n\mathbf{P}_n^T \end{bmatrix}$$

(§2.2.2, §3.2.1)

So, we want to solve P**m**=0 for **m**, where P is 2n by 12

Can this be done?

(§2.2.2, §3.2.1)

So, we want to solve P**m=0** for **m**, where P is 2n by 12

This problem is a bit tricky

Clearly **m=0** is a solution (degenerate solution)

There must be another solution (if we believe our imaging model)

If **m** is a solution, then a scalar multiple of **m** is also (homogeneity)

So, we solve P**m=0** under the constraint that |**m**|=1

If n>6, then this typically will not have a solution due to error (over-constrained)

To simultaneously deal with this problem, AND to use the information from multiple points, we find a "best" solution, using more than 6 points.

Linear Least Squares (§3.1)

- Very common problem in vision: solve an over-constrained system of linear equations
- More equations allows multiple measurements to be used
- Least squares means that you minimize squared error (the difference between your model and your data)
- Least squares minimization is (relatively) easy
- Not very robust to outliers (assumes error is Gaussian)
- Used, overused, and abused.

Linear Least Squares (§3.1)

We will look at two problems

First, $U\mathbf{x} = \mathbf{y}$ where U has more rows than needed

Second, $U\mathbf{x} = \mathbf{0}$ subject to $|\mathbf{x}| = 1$ were U has more rows than needed

Hopefully you will recognize the **second** problem as our camera calibration problem.

We will start with the first (a bit easier and also useful)

Linear Least Squares (§3.1)

Problem one $U\mathbf{x} = \mathbf{y}$ where U has more rows than needed

U is not square, so inverting it does not work

(Following solution sketch is optional, but try to get main idea)

Define $\mathbf{e} = U\mathbf{x} - \mathbf{y}$ and $E = |\mathbf{e}|^2 = \mathbf{e}^T\mathbf{e}$

Linear Least Squares (§3.1)

$$\frac{\partial E}{\partial x_i} = 2\frac{\partial \mathbf{e}^T}{\partial x_i}\mathbf{e} = 0 \quad \text{(for minimum)}$$

$$\frac{\partial \mathbf{e}^T}{\partial x_i} = \mathbf{c}_i$$ where $\mathbf{c}_i$ is column i of U (from $\mathbf{e} = U\mathbf{x} - \mathbf{y}$)

So, $$\frac{\partial \mathbf{e}^T}{\partial x_i}\mathbf{e} = U^T\mathbf{e} = U^T(U\mathbf{x} - \mathbf{y}) = 0$$

Linear Least Squares (§3.1)

From the previous slide our condition is $U^T(U\mathbf{x} - \mathbf{y}) = 0$

Or $U^T U\mathbf{x} = U^T\mathbf{y}$

So $\mathbf{x} = (U^T U)^{-1} U^T \mathbf{y}$ Note that $(U^T U)^{-1}$ can easily be shown to exist

Thus

$\mathbf{x} = U^{\dagger}\mathbf{y}$ where $U^{\dagger} = (U^T U)^{-1} U^T$ is the pseudoinverse of U