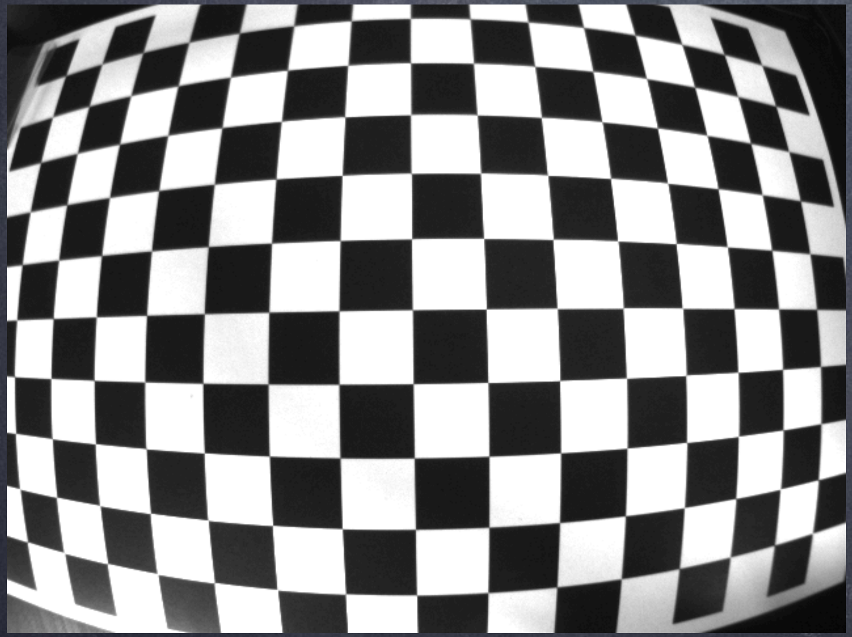
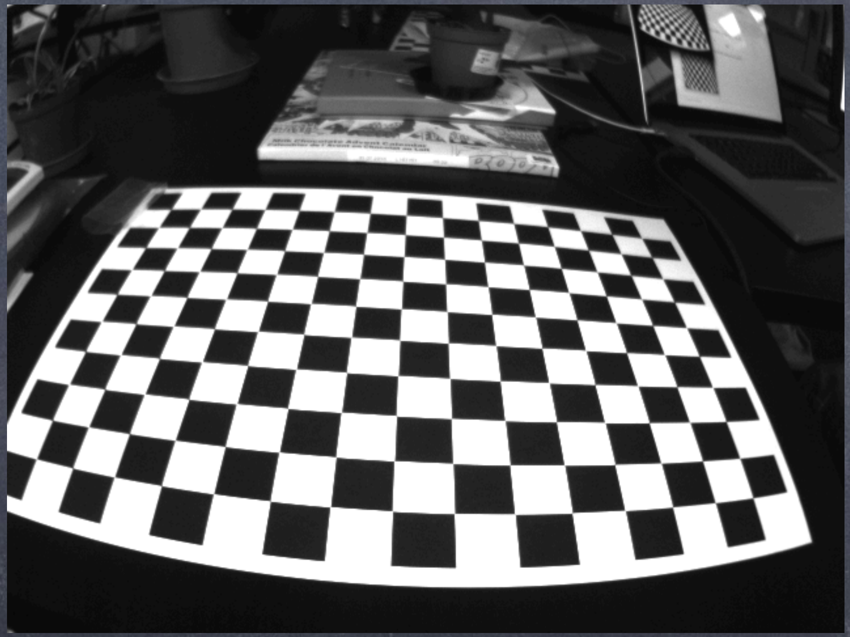
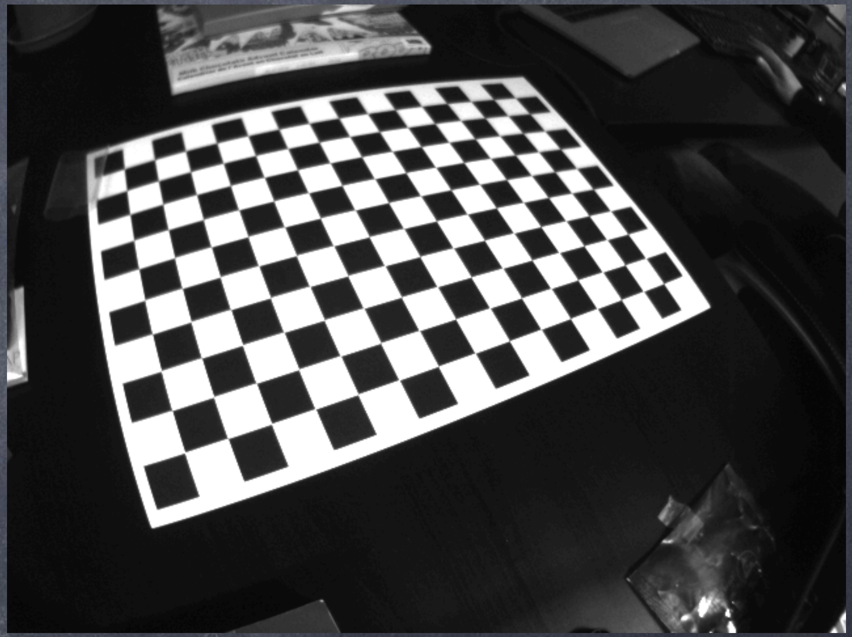
Computer Vision

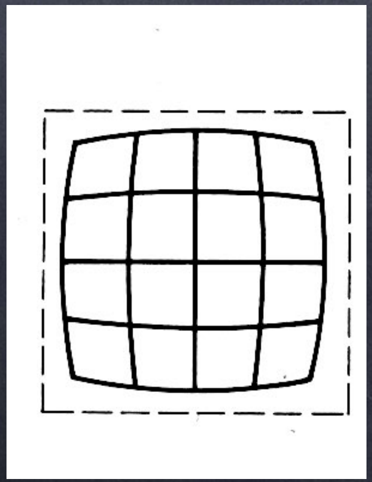
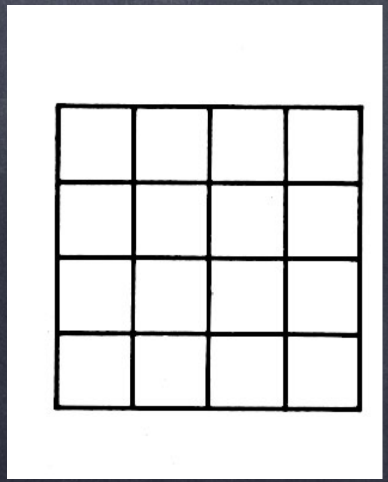
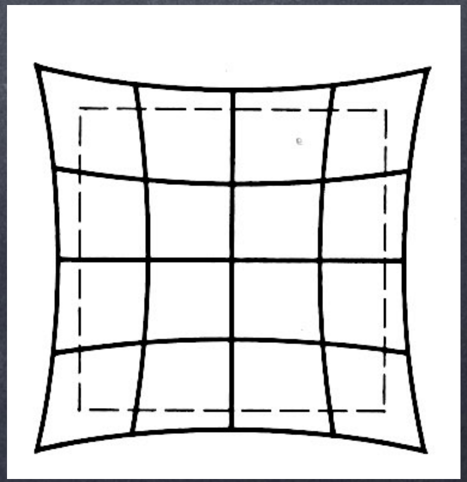
Coming back to Camera Calibration









Modeling distortion

Distortion-Free:

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

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

With Distortion:

1. Project (X, Y, Z) to "normalized" image coordinates

$$x_n = \frac{X}{Z}$$

$$y_n = \frac{Y}{Z}$$

2. Apply radial distortion

$$r^2 = x_n^2 + y_n^2$$

$$x_d = x_n\left(1 + \kappa_1 r^2 + \kappa_2 r^4\right)$$

$$y_d = y_n\left(1 + \kappa_1 r^2 + \kappa_2 r^4\right)$$

3. Apply focal length translate image center

$$x = f x_d + x_c$$

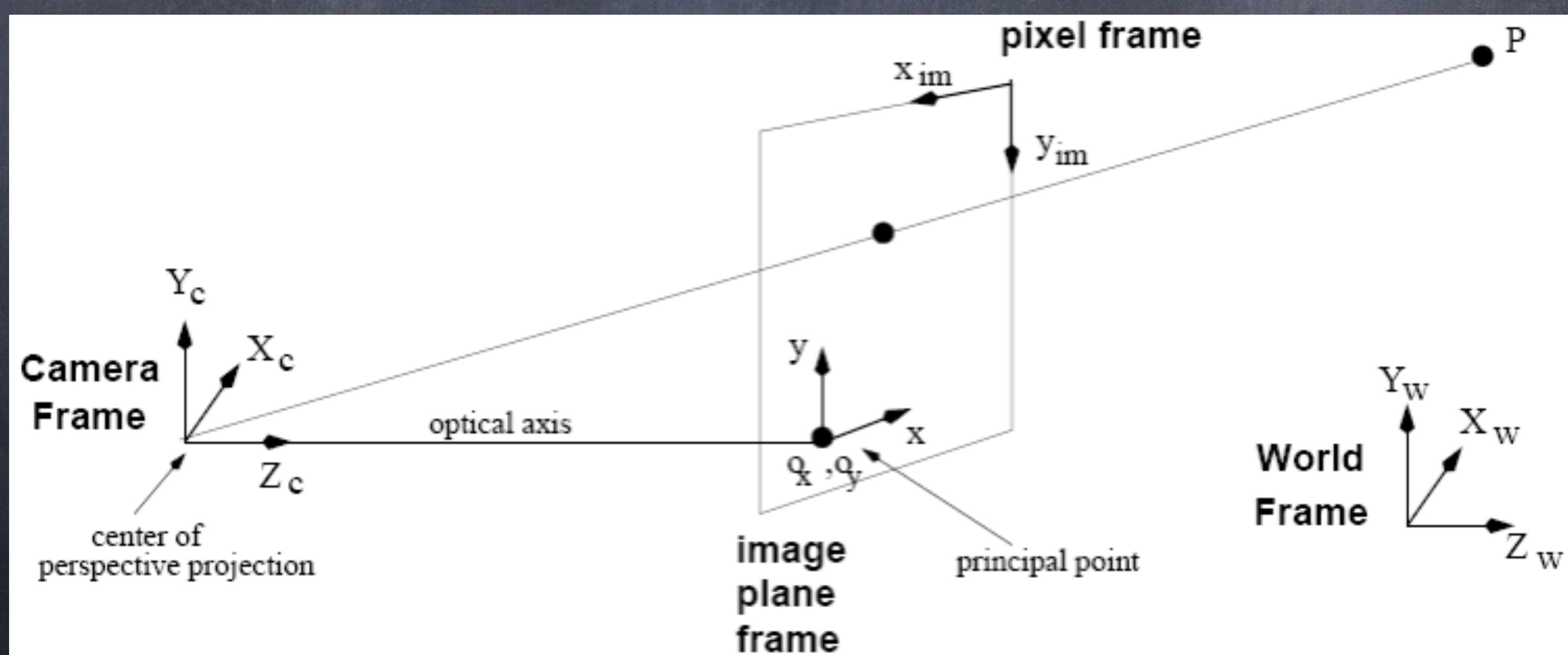
$$y = f y_d + y_c$$

- To model lens distortion
  - Use above projection operation instead of standard projection matrix multiplication

Camera Calibration - Goal

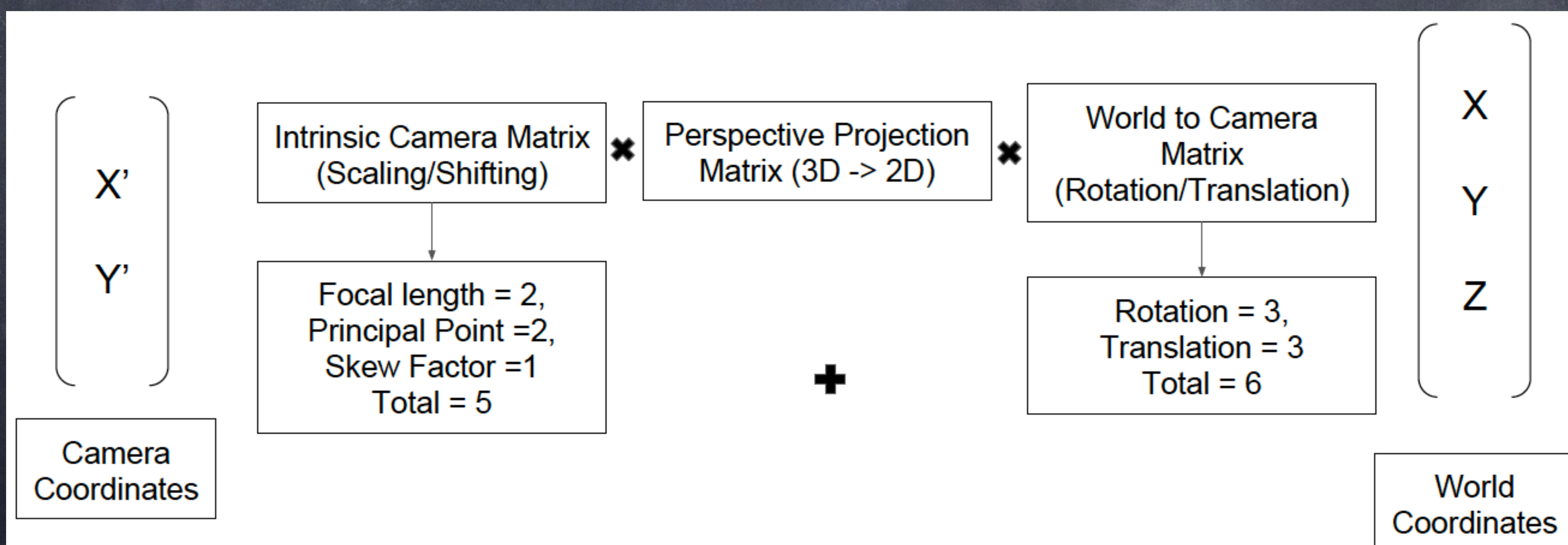
- Estimate the extrinsic and intrinsic camera parameters.

$$x_{im} = -f/s_x \, \frac{R_1^T(P_w - T)}{R_3^T(P_w - T)} + o_x, \qquad y_{im} = -f/s_y \, \frac{R_2^T(P_w - T)}{R_3^T(P_w - T)} + o_y$$



Camera Calibration

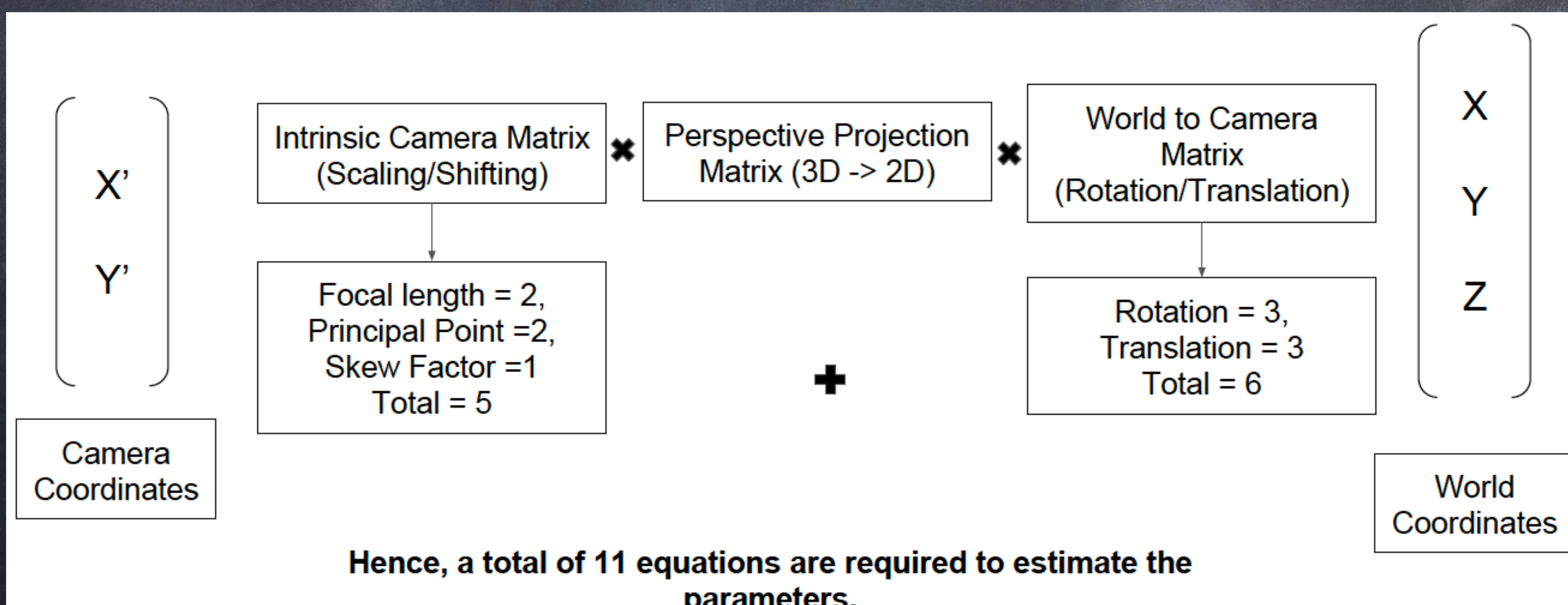
- Estimate the extrinsic and intrinsic camera parameters.



Hence, a total of 11 equations are required to estimate the parameters.

Camera Calibration

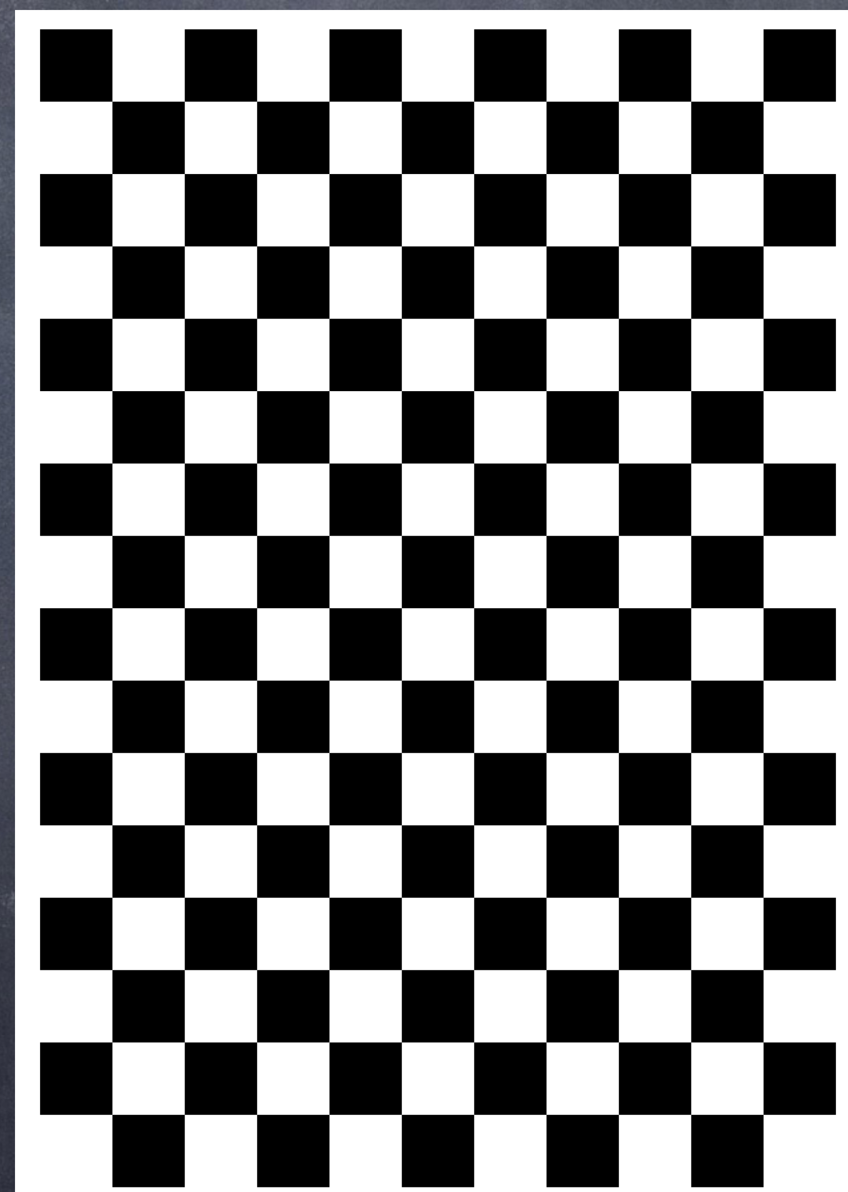
- Using a set of known correspondences between point features in the world (X_w, Y_w, Z_w) and their projections on the image (x_{im}, y_{im})

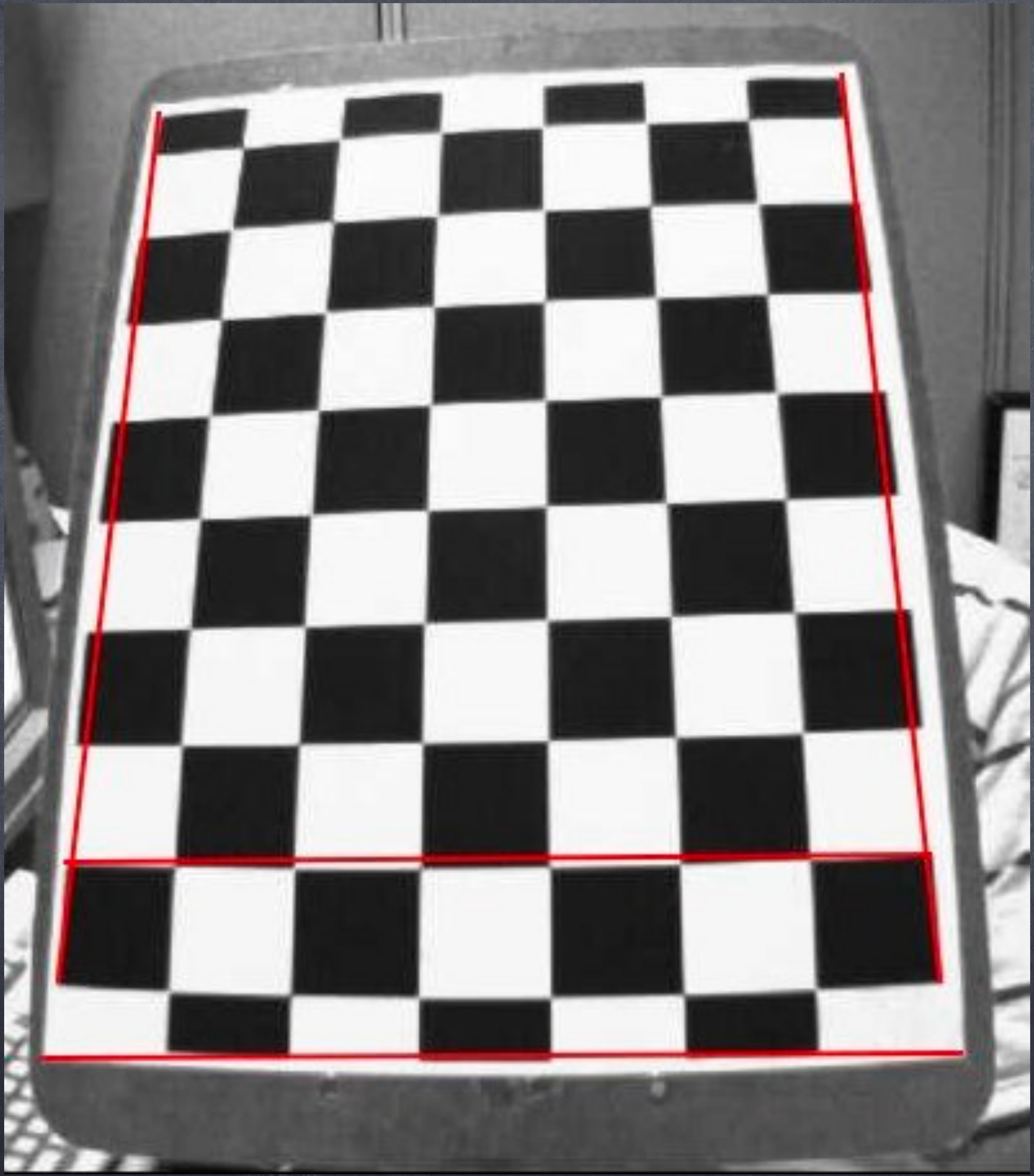


Camera Calibration

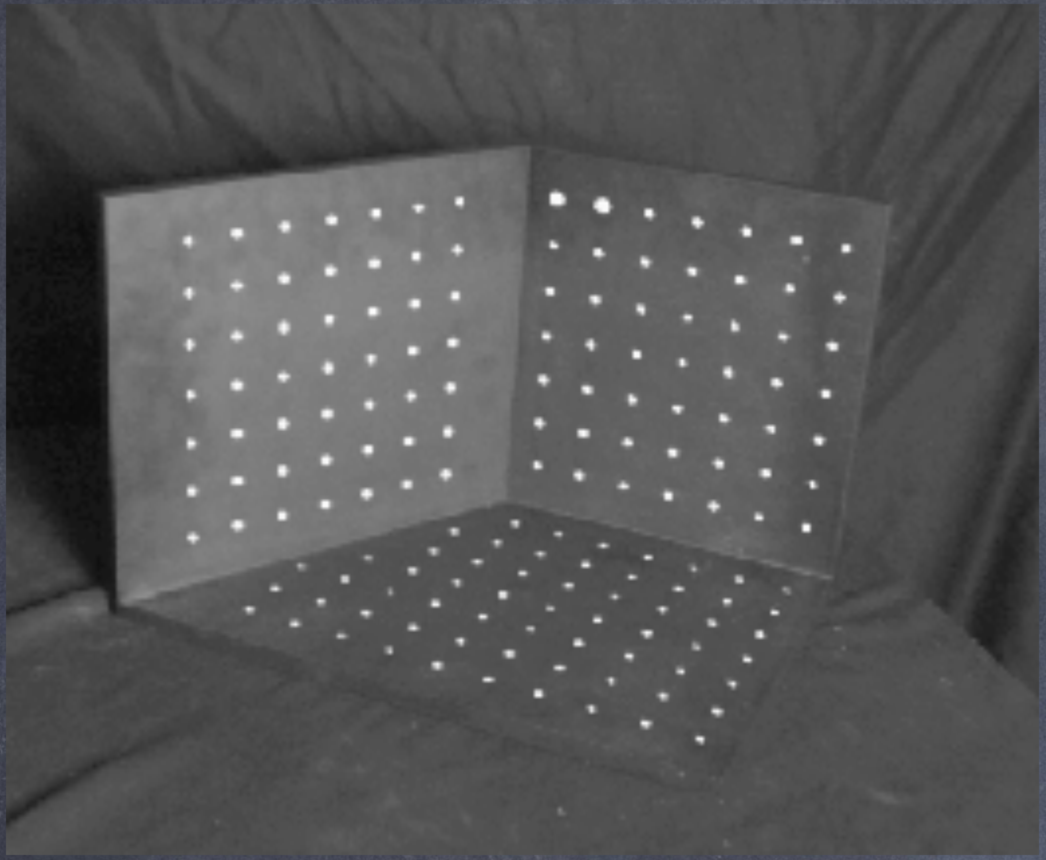
- Camera calibration requires two things: a physical calibration pattern and an algorithm which estimates the parameters

Pattern





Camera calibration



- Directly estimate 11 unknowns in the **M** matrix using known 3D points (*Xi*,*Yi*,*Zi*) and measured feature positions (*ui*,*vi*)

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \sim \begin{bmatrix} m_{00} & m_{01} & m_{02} & m_{03} \\ m_{10} & m_{11} & m_{12} & m_{13} \\ m_{20} & m_{21} & m_{22} & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

Parameter Estimation

Indirect camera calibration

- Estimate the elements of the projection matrix.
- Compute the intrinsic/extrinsic camera parameters from the entries of the projection matrix.

$$M = M_{in}\ M_{ex} = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{bmatrix}$$

$$\Downarrow$$

$$M = \begin{bmatrix} -f_x r_{11} + o_x r_{31} & -f_x r_{12} + o_x r_{32} & -f_x r_{13} + o_x r_{33} & -f_x T_x + o_x T_z \\ -f_y r_{21} + o_y r_{31} & -f_y r_{22} + o_y r_{32} & -f_y r_{23} + o_y r_{33} & -f_y T_y + o_y T_z \\ r_{31} & r_{32} & r_{33} & T_z \end{bmatrix}$$

Step 1: solve for m_{ij}'s

- M has 11 independent entries.
  - e.g., divide every entry by m_{11}

$$M = M_{in}\ M_{ex} = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{bmatrix}$$

- Need at least 11 equations for computing M.
- Need at least 6 world-image point correspondences.

How we can solve?

Direct Linear Calibration

$$\begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} \sim \begin{bmatrix} m_{00} & m_{01} & m_{02} & m_{03} \\ m_{10} & m_{11} & m_{12} & m_{13} \\ m_{20} & m_{21} & m_{22} & 1 \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix}$$

$$u_i = \frac{m_{00}X_i + m_{01}Y_i + m_{02}Z_i + m_{03}}{m_{20}X_i + m_{21}Y_i + m_{22}Z_i + 1}$$

$$v_i = \frac{m_{10}X_i + m_{11}Y_i + m_{12}Z_i + m_{13}}{m_{20}X_i + m_{21}Y_i + m_{22}Z_i + 1}$$

$$u_i(m_{20}X_i + m_{21}Y_i + m_{22}Z_i + 1) = m_{00}X_i + m_{01}Y_i + m_{02}Z_i + m_{03}$$
$$v_i(m_{20}X_i + m_{21}Y_i + m_{22}Z_i + 1) = m_{10}X_i + m_{11}Y_i + m_{12}Z_i + m_{13}$$

$$\begin{bmatrix} X_i & Y_i & Z_i & 1 & 0 & 0 & 0 & 0 & -u_iX_i & -u_iY_i & -u_iZ_i \\ 0 & 0 & 0 & 0 & X_i & Y_i & Z_i & 1 & -v_iX_i & -v_iY_i & -v_iZ_i \end{bmatrix} \begin{bmatrix} m_{00} \\ m_{01} \\ m_{02} \\ m_{03} \\ m_{10} \\ m_{11} \\ m_{12} \\ m_{13} \\ m_{20} \\ m_{21} \\ m_{22} \end{bmatrix} = \begin{bmatrix} u_i \\ v_i \end{bmatrix}$$

Direct Linear Calibration

$$\begin{bmatrix} X_1 & Y_1 & Z_1 & 1 & 0 & 0 & 0 & 0 & u_iX_1 & u_iY_1 & u_iZ_i \\ 0 & 0 & 0 & 0 & X_1 & Y_1 & Z_1 & 1 & v_1X_1 & v_1Y_1 & v_1Z_1 \\ & & & & & & \vdots & & & & \\ X_n & Y_n & Z_n & 1 & 0 & 0 & 0 & 0 & u_nX_n & u_nY_n & u_nZ_n \\ 0 & 0 & 0 & 0 & X_n & Y_n & Z_n & 1 & v_nX_n & v_nY_n & v_nZ_n \end{bmatrix} \begin{bmatrix} m_{00} \\ m_{01} \\ m_{02} \\ m_{03} \\ m_{10} \\ m_{11} \\ m_{12} \\ m_{13} \\ m_{20} \\ m_{21} \\ m_{22} \end{bmatrix} = \begin{bmatrix} u_1 \\ v_1 \\ \vdots \\ u_n \\ v_n \end{bmatrix}$$

Can solve for m_{ij} by linear least squares

$$minimize \left\| \begin{bmatrix} X_1 & Y_1 & Z_1 & 1 & 0 & 0 & 0 & 0 & u_iX_1 & u_iY_1 & u_iZ_i \\ 0 & 0 & 0 & 0 & X_1 & Y_1 & Z_1 & 1 & v_1X_1 & v_1Y_1 & v_1Z_1 \\ & & & & & & \vdots & & & & \\ X_n & Y_n & Z_n & 1 & 0 & 0 & 0 & 0 & u_nX_n & u_nY_n & u_nZ_n \\ 0 & 0 & 0 & 0 & X_n & Y_n & Z_n & 1 & v_nX_n & v_nY_n & v_nZ_n \end{bmatrix} \begin{bmatrix} m_{00} \\ m_{01} \\ m_{02} \\ m_{03} \\ m_{10} \\ m_{11} \\ m_{12} \\ m_{13} \\ m_{20} \\ m_{21} \\ m_{22} \end{bmatrix} - \begin{bmatrix} u_1 \\ v_1 \\ \vdots \\ u_n \\ v_n \end{bmatrix} \right\|$$

What error function are we minimizing?

Nonlinear estimation

Feature measurement equations

$$u_i = \frac{m_{00}X_i + m_{01}Y_i + m_{02}Z_i + m_{03}}{m_{20}X_i + m_{21}Y_i + m_{22}Z_i + 1}$$

$$v_i = \frac{m_{10}X_i + m_{11}Y_i + m_{12}Z_i + m_{13}}{m_{20}X_i + m_{21}Y_i + m_{22}Z_i + 1}$$

Minimize "image-space error"

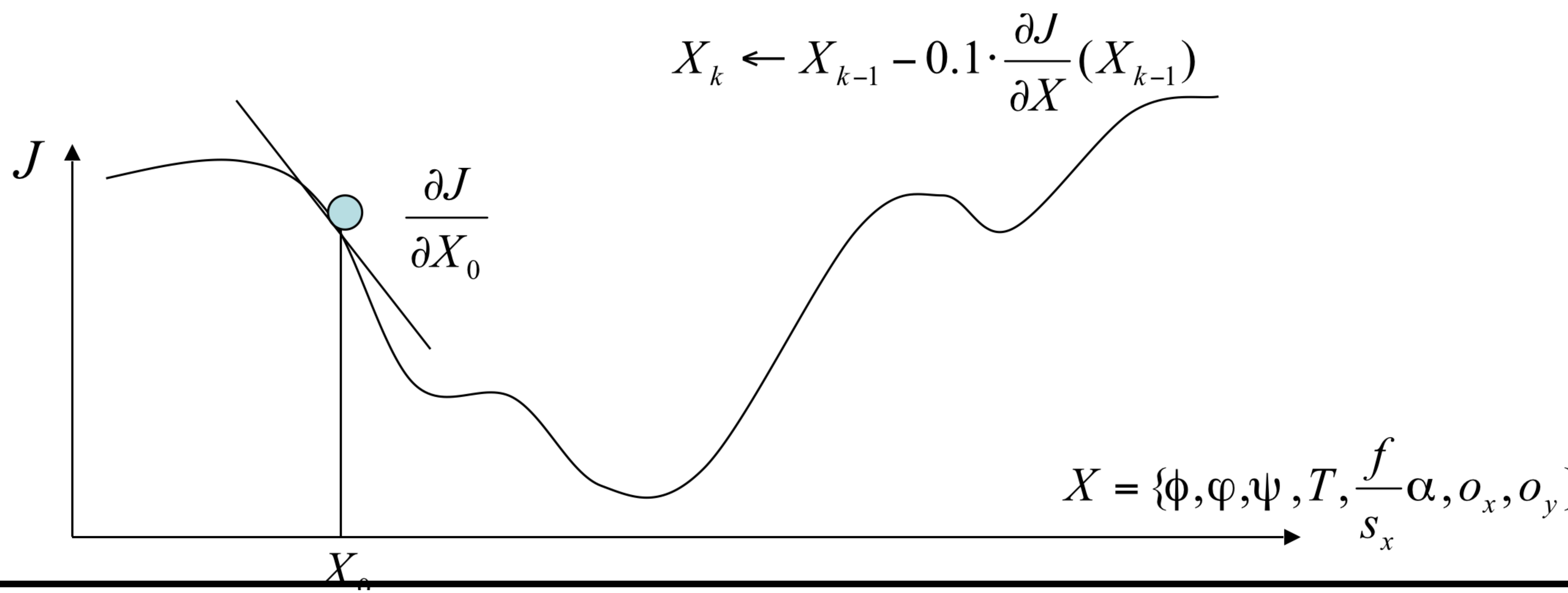
$$e(\mathrm{M}) = \sum_i [(u_i - \frac{m_{00}X_i + m_{01}Y_i + m_{02}Z_i + m_{03}}{m_{20}X_i + m_{21}Y_i + m_{22}Z_i + 1})^2 + (v_i - \frac{m_{10}X_i + m_{11}Y_i + m_{12}Z_i + m_{13}}{m_{20}X_i + m_{21}Y_i + m_{22}Z_i + 1})^2]$$

How to minimize e(**M**)?

- Non-linear regression (least squares),

Calibration by nonlinear Least Squares

- Gradient descent:



Statistical estimation

Feature measurement equations

$$\begin{aligned} u_i &= f(\mathrm{M}, \mathbf{x}_i) + n_i = \widehat{u}_i + n_i, \quad n_i \sim N(0, \sigma) \\ v_i &= g(\mathrm{M}, \mathbf{x}_i) + m_i = \widehat{v}_i + m_i, \quad m_i \sim N(0, \sigma) \end{aligned}$$

Likelihood of measurements given **M**

$$\begin{aligned} L &= \prod_i p(u_i | \widehat{u}_i) p(v_i | \widehat{v}_i) \\ &= \prod_i e^{-(u_i - \widehat{u}_i)^2/\sigma^2} e^{-(v_i - \widehat{v}_i)^2/\sigma^2} \end{aligned}$$

Negative Log likelihood

$$C(\mathrm{M}) = -\log L = \sum_i (u_i - \widehat{u}_i)^2/\sigma_i^2 + (v_i - \widehat{v}_i)^2/\sigma_i^2$$

Minimize C wrt. **M**

- gives maximum likelihood estimate (MLE)

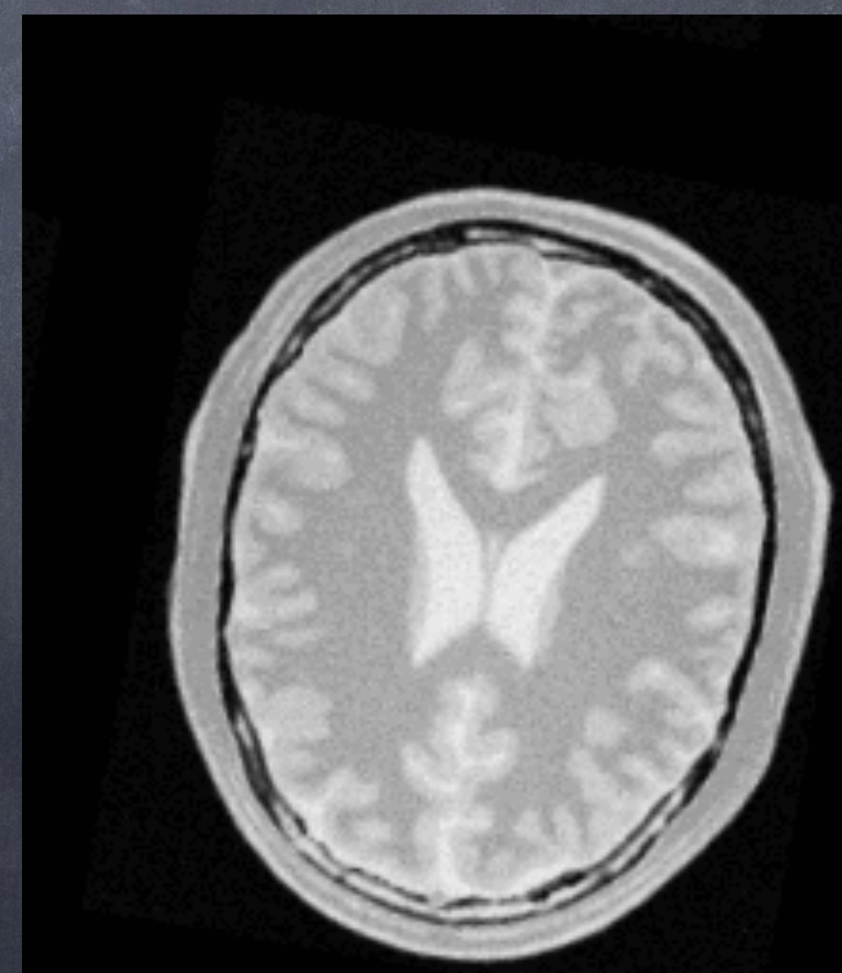
Next Topic

- Image registration/alignment

Image Registration

- How we can register two images with respect to each other?

I_m



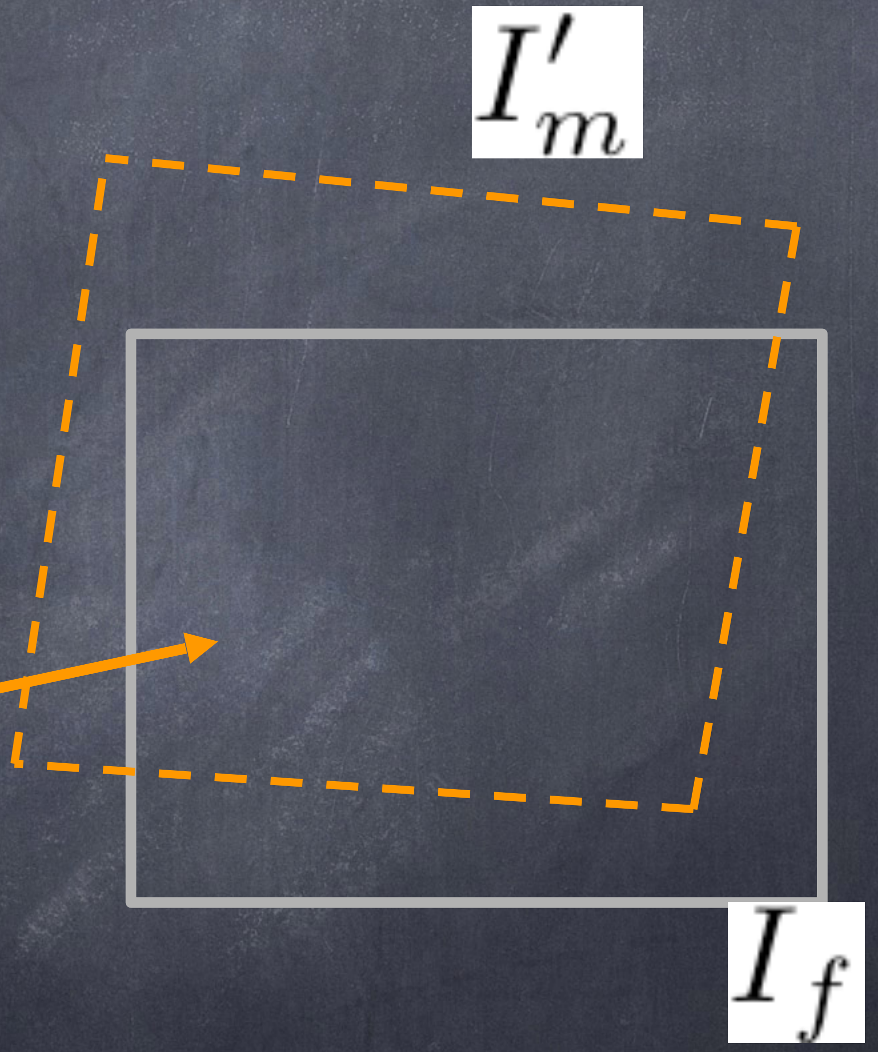
I_f

Intensity based Registration

- How we can do that?

Example Error Measure: SSD

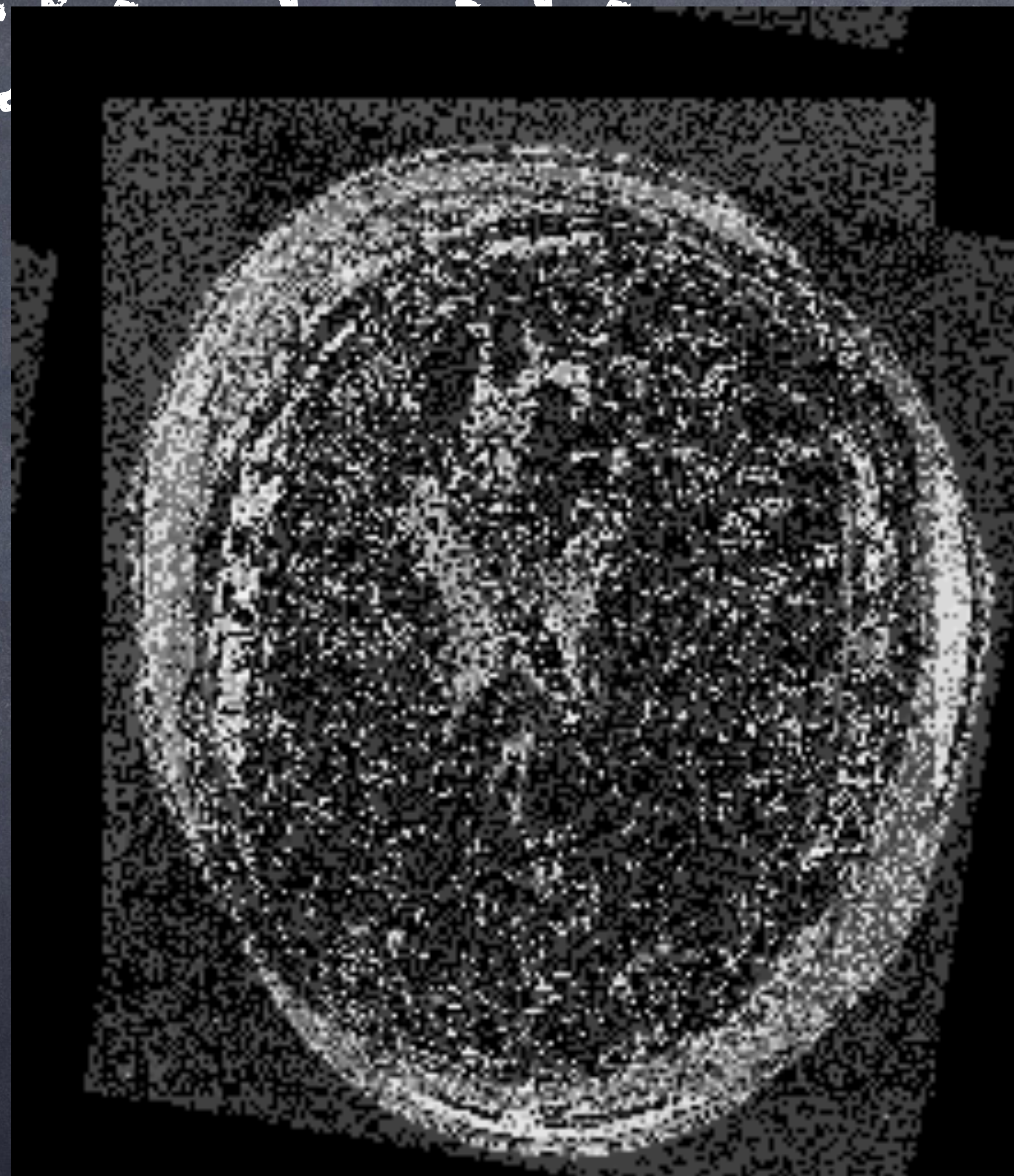
$$\sum_{\mathbf{p}\in\Omega}\left[I_f(\mathbf{p}) - I'_m(\mathbf{p})\right]^2$$



Ω Region of intersection between images

$\mathbf{p}$ Pixel location within region

SSD Example:
Ini ent



- More in next lecture