



Geometric Camera Calibration

Chapter 2

Guido Gerig
CS 6320 Spring 2015

Slides modified from Marc Pollefeys, UNC Chapel Hill, Comp256,
Other slides and illustrations from J. Ponce, addendum to course book,
and Trevor Darrell, Berkeley, C280 Computer Vision Course.

Equation: World coordinates to image pixels



pixel coordinates

$$\vec{p} = \frac{1}{z} M$$

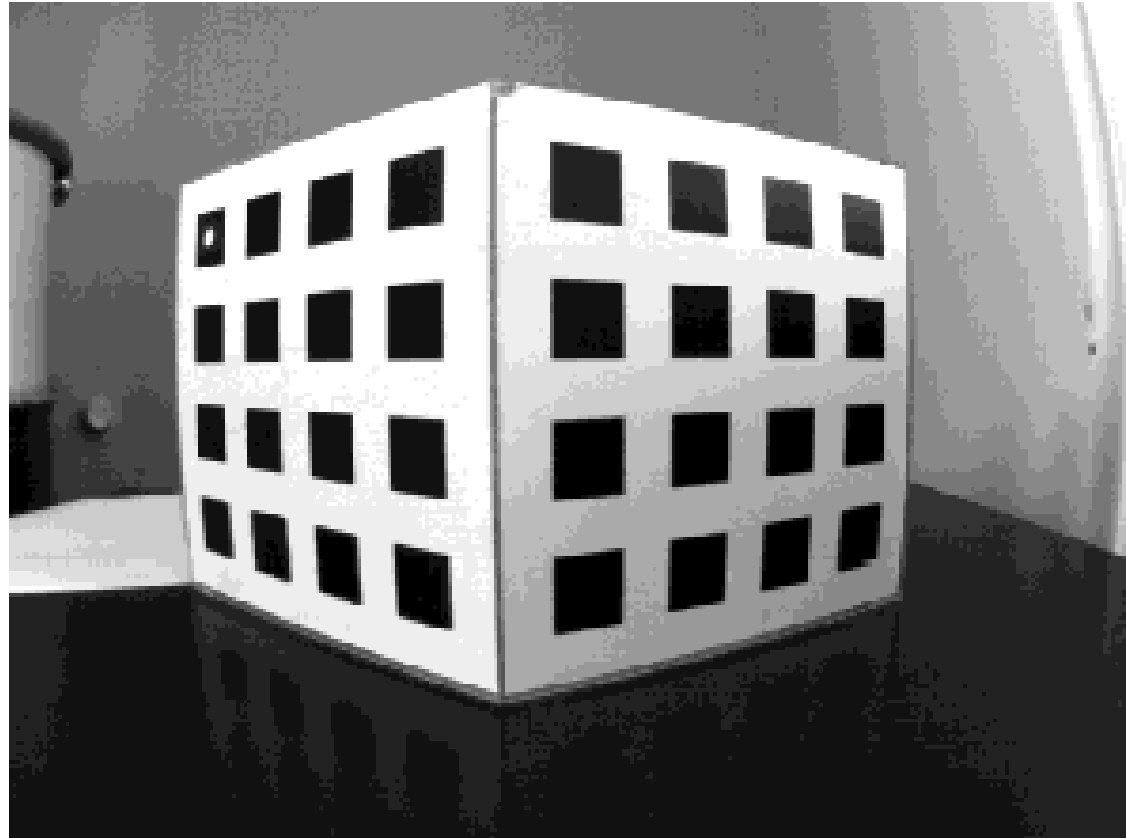
world coordinates

$${}^w \vec{p}$$

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \frac{1}{z} \begin{pmatrix} \cdot & m_1^T & \cdot & \cdot \\ \cdot & m_2^T & \cdot & \cdot \\ \cdot & m_3^T & \cdot & \cdot \end{pmatrix} \begin{pmatrix} {}^w p_x \\ {}^w p_y \\ {}^w p_z \\ 1 \end{pmatrix} \left\{ \begin{array}{l} u = \frac{m_1 \cdot \vec{P}}{m_3 \cdot \vec{P}} \\ v = \frac{m_2 \cdot \vec{P}}{m_3 \cdot \vec{P}} \end{array} \right.$$

Conversion back from homogeneous coordinates leads to:

Calibration target



The Opti-CAL Calibration Target Image

Find the position, u_i and v_i , in pixels,
of each calibration object feature point.

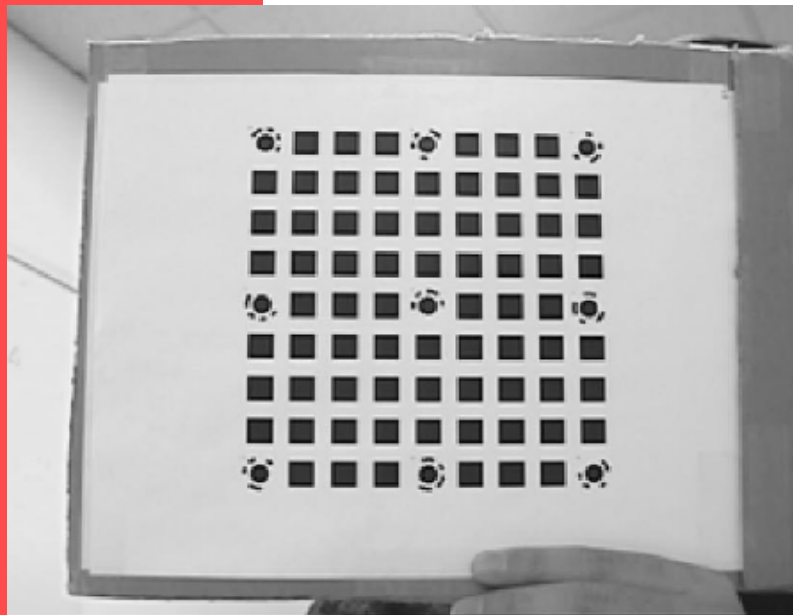
<http://www.kinetic.bc.ca/CompVision/opti-CAL.html>

Camera calibration



From before, we had these equations relating image positions, u, v , to points at 3-d positions P (in homogeneous coordinates):

$$u = \frac{m_1 \cdot \vec{P}}{m_3 \cdot \vec{P}}$$
$$v = \frac{m_2 \cdot \vec{P}}{m_3 \cdot \vec{P}}$$



So for each feature point, i , we have:

$$(m_1 - u_i m_3) \cdot \vec{P}_i = 0$$

$$(m_2 - v_i m_3) \cdot \vec{P}_i = 0$$



Camera calibration

Stack all these measurements of $i=1\dots n$ points

$$(m_1 - u_i m_3) \cdot \vec{P}_i = 0$$

$$(m_2 - v_i m_3) \cdot \vec{P}_i = 0$$

into a big matrix:

$$\begin{pmatrix} P_1^T & 0^T & -u_1 P_1^T \\ 0^T & P_1^T & -v_1 P_1^T \\ \dots & \dots & \dots \\ P_n^T & 0^T & -u_n P_n^T \\ 0^T & P_n^T & -v_n P_n^T \end{pmatrix} \begin{pmatrix} m_1 \\ m_2 \\ m_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix}$$



Camera calibration

In vector form:

$$\begin{pmatrix} P_1^T & 0^T & -u_1 P_1^T \\ 0^T & P_1^T & -v_1 P_1^T \\ \dots & \dots & \dots \\ P_n^T & 0^T & -u_n P_n^T \\ 0^T & P_n^T & -v_n P_n^T \end{pmatrix} \begin{pmatrix} m_1 \\ m_2 \\ m_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix}$$

Showing all the elements:

$$\begin{pmatrix} P_{1x} & P_{1y} & P_{1z} & 1 & 0 & 0 & 0 & 0 & -u_1 P_{1x} & -u_1 P_{1y} & -u_1 P_{1z} & -u_1 \\ 0 & 0 & 0 & 0 & P_{1x} & P_{1y} & P_{1z} & 1 & -v_1 P_{1x} & -v_1 P_{1y} & -v_1 P_{1z} & -v_1 \\ & & & \dots & \dots & \dots & & & & & & \\ P_{nx} & P_{ny} & P_{nz} & 1 & 0 & 0 & 0 & 0 & -u_n P_{nx} & -u_n P_{ny} & -u_n P_{nz} & -u_n \\ 0 & 0 & 0 & 0 & P_{nx} & P_{ny} & P_{nz} & 1 & -v_n P_{nx} & -v_n P_{ny} & -v_n P_{nz} & -v_n \end{pmatrix} \begin{pmatrix} m_{11} \\ m_{12} \\ m_{13} \\ m_{14} \\ m_{21} \\ m_{22} \\ m_{23} \\ m_{24} \\ m_{31} \\ m_{32} \\ m_{33} \\ m_{34} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$



Camera calibration

$$\begin{pmatrix} P_{1x} & P_{1y} & P_{1z} & 1 & 0 & 0 & 0 & 0 & -u_1 P_{1x} & -u_1 P_{1y} & -u_1 P_{1z} & -u_1 \\ 0 & 0 & 0 & 0 & P_{1x} & P_{1y} & P_{1z} & 1 & -v_1 P_{1x} & -v_1 P_{1y} & -v_1 P_{1z} & -v_1 \\ & & & & \dots & \dots & \dots & & & & & \\ P_{nx} & P_{ny} & P_{nz} & 1 & 0 & 0 & 0 & 0 & -u_n P_{nx} & -u_n P_{ny} & -u_n P_{nz} & -u_n \\ 0 & 0 & 0 & 0 & P_{nx} & P_{ny} & P_{nz} & 1 & -v_n P_{nx} & -v_n P_{ny} & -v_n P_{nz} & -v_n \end{pmatrix} \begin{pmatrix} m_{11} \\ m_{12} \\ m_{13} \\ m_{14} \\ m_{21} \\ m_{22} \\ m_{23} \\ m_{24} \\ m_{31} \\ m_{32} \\ m_{33} \\ m_{34} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

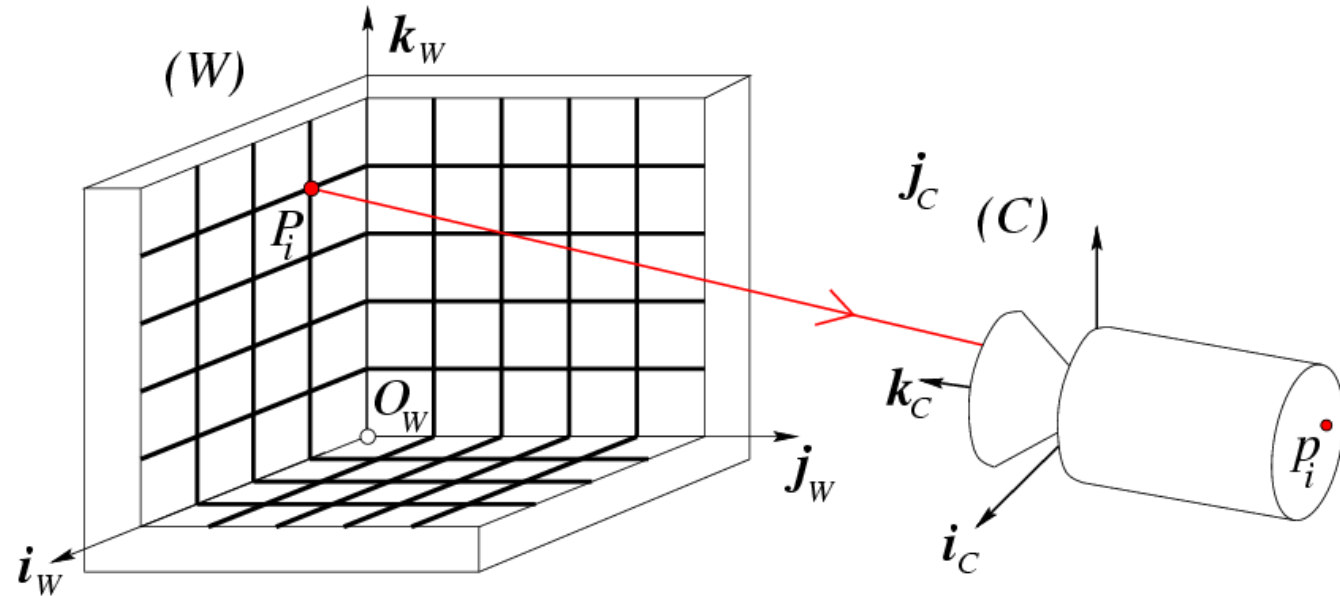
Q
 $m = 0$

We want to solve for the unit vector m (the stacked one) that minimizes $|Qm|^2$

The eigenvector assoc. to the minimum eigenvalue of the matrix $Q^T Q$ gives us that because it is the unit vector x that minimizes $x^T Q^T Q x$.



Calibration Problem



Given n points P_1, \dots, P_n with *known* positions and their images p_1, \dots, p_n

Find \mathbf{i} and \mathbf{e} such that

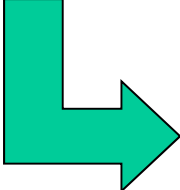
$$\left\{ \begin{array}{l} u_i = \frac{\mathbf{m}_1(\mathbf{i}, \mathbf{e}) \cdot \mathbf{P}_i}{\mathbf{m}_3(\mathbf{i}, \mathbf{e}) \cdot \mathbf{P}_i} \\ v_i = \frac{\mathbf{m}_2(\mathbf{i}, \mathbf{e}) \cdot \mathbf{P}_i}{\mathbf{m}_3(\mathbf{i}, \mathbf{e}) \cdot \mathbf{P}_i} \end{array} \right. \quad \sum_{i=1}^n \left[\left(u_i - \frac{\mathbf{m}_1(\mathbf{i}, \mathbf{e}) \cdot \mathbf{P}_i}{\mathbf{m}_3(\mathbf{i}, \mathbf{e}) \cdot \mathbf{P}_i} \right)^2 + \left(v_i - \frac{\mathbf{m}_2(\mathbf{i}, \mathbf{e}) \cdot \mathbf{P}_i}{\mathbf{m}_3(\mathbf{i}, \mathbf{e}) \cdot \mathbf{P}_i} \right)^2 \right] \text{ is minimized}$$



Analytical Photogrammetry

Given n points P_1, \dots, P_n with *known* positions and their images p_1, \dots, p_n

Find \mathbf{i} and \mathbf{e} such that


$$\sum_{i=1}^n \left[\left(u_i - \frac{\mathbf{m}_1(\mathbf{i}, \mathbf{e}) \cdot \mathbf{P}_i}{\mathbf{m}_3(\mathbf{i}, \mathbf{e}) \cdot \mathbf{P}_i} \right)^2 + \left(v_i - \frac{\mathbf{m}_2(\mathbf{i}, \mathbf{e}) \cdot \mathbf{P}_i}{\mathbf{m}_3(\mathbf{i}, \mathbf{e}) \cdot \mathbf{P}_i} \right)^2 \right] \text{ is minimized}$$

Non-Linear Least-Squares Methods

- Newton
- Gauss-Newton
- Levenberg-Marquardt

Iterative, quadratically convergent in favorable situations



Homogeneous Linear Systems

$$\boxed{A} \quad \boxed{x} = \boxed{0}$$

Square system:

- unique solution: 0
- unless $\text{Det}(A)=0$

$$\begin{array}{|c|} \hline \\ \hline \boxed{A} \\ \hline \\ \hline \end{array} \quad \boxed{x} = \begin{array}{|c|} \hline \\ \hline \boxed{0} \\ \hline \\ \hline \end{array}$$

Rectangular system ??

- 0 is always a solution

Minimize $|Ax|^2$
under the constraint $|x|^2=1$



How do you solve overconstrained homogeneous linear equations ??

$$E = |\mathcal{U}\mathbf{x}|^2 = \mathbf{x}^T (\mathcal{U}^T \mathcal{U}) \mathbf{x}$$

- Orthonormal basis of eigenvectors: $\mathbf{e}_1, \dots, \mathbf{e}_q$.
- Associated eigenvalues: $0 \leq \lambda_1 \leq \dots \leq \lambda_q$.
- Any vector can be written as

$$\mathbf{x} = \mu_1 \mathbf{e}_1 + \dots + \mu_q \mathbf{e}_q$$

for some μ_i ($i = 1, \dots, q$) such that $\mu_1^2 + \dots + \mu_q^2 = 1$.

$$\begin{aligned} E(\mathbf{x}) - E(\mathbf{e}_1) &= \mathbf{x}^T (\mathcal{U}^T \mathcal{U}) \mathbf{x} - \mathbf{e}_1^T (\mathcal{U}^T \mathcal{U}) \mathbf{e}_1 \\ &= \lambda_1^2 \mu_1^2 + \dots + \lambda_q^2 \mu_q^2 - \lambda_1^2 \\ &\geq \lambda_1^2 (\mu_1^2 + \dots + \mu_q^2 - 1) = 0 \end{aligned}$$

The solution is \mathbf{e}_1 .

remember: $\text{EIG}(\mathcal{U}^T \mathcal{U}) = \text{SVD}(\mathcal{U})$, i.e. solution is \mathbf{V}_n



Matlab Solution

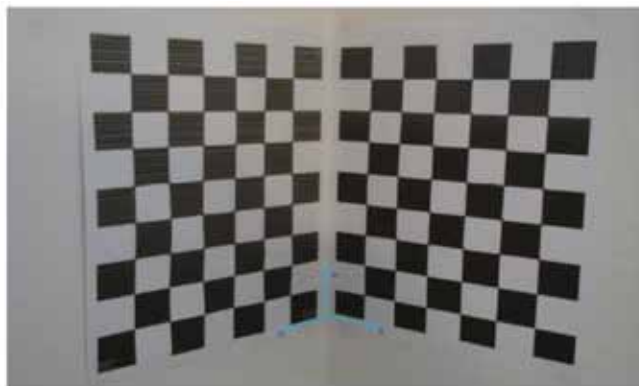


Figure 1: Checkerboard pattern on the wall corner and the world frame coordinate axes

Example:
60 point pairs

Least squares method is used to estimate the calibration matrix. There are 120 homogeneous linear equations in twelve variables, which are the coefficients of the calibration matrix \mathcal{M} . Lets denote this system of linear equations as

$$\mathcal{P}\mathbf{m} = 0, \quad \mathbf{m} := [\mathbf{m}_1 \quad \mathbf{m}_2 \quad \mathbf{m}_3]^T, \quad (1)$$

where, $\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3$ are first, second and third rows of the matrix \mathcal{M} respectively. \mathbf{m} is a 12×1 vector, and \mathcal{P} is a 120×12 matrix. The problem of least square estimation of \mathcal{P} is defined as

$$\min \|\mathcal{P}\mathbf{m}\|^2, \quad \text{subject to } \|\mathbf{m}\|^2 = 1. \quad (2)$$

As it turns out, the solution of above problem is given by the eigenvector of matrix $\mathcal{P}^T\mathcal{P}$ having the least eigenvalue. The eigenvectors of matrix $\mathcal{P}^T\mathcal{P}$ can also be computed by performing the singular value decomposition (SVD) of \mathcal{P} . The 12 right singular vectors of \mathcal{P} are also the eigenvectors of $\mathcal{P}^T\mathcal{P}$.

```
%Perform SVD of P
[U S V] = svd(P);
[min_val, min_index] = min(diag(S(1:12,1:12)));

%m is given by right singular vector of min. singular value
m = V(1:12, min_index);
```



Degenerate Point Configurations

Are there other solutions besides M ??

$$0 = \mathcal{P}l = \begin{pmatrix} P_1^T & 0^T & -u_1 P_1^T \\ 0^T & P_1^T & -v_1 P_1^T \\ \dots & \dots & \dots \\ P_n^T & 0^T & -u_n P_n^T \\ 0^T & P_n^T & -v_n P_n^T \end{pmatrix} \begin{pmatrix} \lambda \\ \mu \\ \nu \end{pmatrix} = \begin{pmatrix} P_1^T \lambda - u_1 P_1^T \nu \\ P_1^T \mu - v_1 P_1^T \nu \\ \dots \\ P_n^T \lambda - u_n P_n^T \nu \\ P_n^T \mu - v_n P_n^T \nu \end{pmatrix}$$



$$\begin{cases} P_i^T \lambda - \frac{m_1^T P_i}{m_3^T P_i} P_i^T \nu = 0 \\ P_i^T \mu - \frac{m_2^T P_i}{m_3^T P_i} P_i^T \nu = 0 \end{cases} \longrightarrow \begin{cases} P_i^T (\lambda m_3^T - m_1 \nu^T) P_i = 0 \\ P_i^T (\mu m_3^T - m_2 \nu^T) P_i = 0 \end{cases}$$

- Coplanar points: $(\lambda, \mu, \nu) = (\Pi, 0, 0)$ or $(0, \Pi, 0)$ or $(0, 0, \Pi)$
- Points lying on the intersection curve of two quadric surfaces = straight line + twisted cubic

Does **not** happen for 6 or more random points!

Camera calibration



Once you have the M matrix, can recover the intrinsic and extrinsic parameters.

Estimation of the Intrinsic and Extrinsic Parameters, see pdf slides [S.M. Abdallah](#).

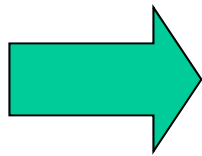
$$\mathcal{M} = \begin{pmatrix} \alpha \mathbf{r}_1^T - \alpha \cot \theta \mathbf{r}_2^T + u_0 \mathbf{r}_3^T & \alpha t_x - \alpha \cot \theta t_y + u_0 t_z \\ \frac{\beta}{\sin \theta} \mathbf{r}_2^T + v_0 \mathbf{r}_3^T & \frac{\beta}{\sin \theta} t_y + v_0 t_z \\ \mathbf{r}_3^T & t_z \end{pmatrix}$$



Once M is known, you still got to recover the intrinsic and extrinsic parameters !!!

This is a decomposition problem, **not** an estimation problem.

$$\boxed{\rho} \mathcal{M} = \begin{pmatrix} \alpha \mathbf{r}_1^T - \alpha \cot \theta \mathbf{r}_2^T + u_0 \mathbf{r}_3^T & \alpha t_x - \alpha \cot \theta t_y + u_0 t_z \\ \frac{\beta}{\sin \theta} \mathbf{r}_2^T + v_0 \mathbf{r}_3^T & \frac{\beta}{\sin \theta} t_y + v_0 t_z \\ \mathbf{r}_3^T & t_z \end{pmatrix}$$



- Intrinsic parameters
- Extrinsic parameters

Slide Samer M Abdallah, Beirut

Estimation of the intrinsic and extrinsic parameters

Write $M = (A, b)$, therefore

$$\rho(A \ b) = \mathcal{K}(\mathcal{R} \ t) \iff \rho \begin{pmatrix} a_1^T \\ a_2^T \\ a_3^T \end{pmatrix} = \begin{pmatrix} \alpha r_1^T - \alpha \cot \theta r_2^T + u_0 r_3^T \\ \frac{\beta}{\sin \theta} r_2^T + v_0 r_3^T \\ r_3^T \end{pmatrix}$$

Using the fact that the rows of a rotation matrix have unit length and are perpendicular to each other yields

$$\begin{cases} \rho = \varepsilon / |a_3|, \\ r_3 = \rho a_3, \\ u_0 = \rho^2 (a_1 \cdot a_3), \\ v_0 = \rho^2 (a_2 \cdot a_3), \end{cases} \quad \text{where } \varepsilon = \mp 1.$$

Since θ is always in the neighborhood of $\pi/2$ with a positive sine, we have

$$\begin{cases} \rho^2 (a_1 \times a_3) = -\alpha r_2 - \alpha \cot \theta r_1, \\ \rho^2 (a_2 \times a_3) = \frac{\beta}{\sin \theta} r_1, \end{cases} \quad \text{and} \quad \begin{cases} \rho^2 |a_1 \times a_3| = \frac{|\alpha|}{\sin \theta}, \\ \rho^2 |a_2 \times a_3| = \frac{|\beta|}{\sin \theta}. \end{cases}$$

Thus,

$$\begin{cases} \cos \theta = -\frac{(a_1 \times a_3) \cdot (a_2 \times a_3)}{|a_1 \times a_3| |a_2 \times a_3|}, \\ \alpha = \rho^2 |a_1 \times a_3| \sin \theta, \\ \beta = \rho^2 |a_2 \times a_3| \sin \theta, \end{cases} \quad \text{and} \quad \begin{cases} r_1 = \frac{\rho^2 \sin \theta}{\beta} (a_2 \times a_3) = \frac{1}{|a_2 \times a_3|} (a_2 \times a_3), \\ r_2 = r_3 \times r_1. \end{cases}$$

Note that there are two possible choices for the matrix \mathcal{R} depending on the value of ε .



Slide Samer M Abdallah, Beirut

Estimation of the intrinsic and extrinsic parameters

The translation parameters can now be recovered by writing $\mathcal{K}t = \bar{\rho}b$, and hence $t = \rho\mathcal{K}^{-1}b$. In practical situations, the sign of t_z is often known in advance (this corresponds to knowing whether the origin of the world coordinate system is in front or behind the camera), which allows the choice of a unique solution for the calibration parameters.



Other Slides following Forsyth&Ponce

Just for additional information on previous slides



Linear Systems

$$\boxed{A} \quad \boxed{x} = \boxed{b}$$

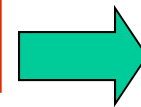
Square system:

- unique solution
- Gaussian elimination

$$\begin{array}{|c|} \hline \\ \hline A \\ \hline \\ \hline \end{array} \quad \boxed{x} = \begin{array}{|c|} \hline \\ \hline b \\ \hline \\ \hline \end{array}$$

Rectangular system ??

- underconstrained:
infinity of solutions
- overconstrained:
no solution



Minimize $|Ax-b|^2$



How do you solve overconstrained linear equations ??

- Define $E = |\mathbf{e}|^2 = \mathbf{e} \cdot \mathbf{e}$ with

$$\begin{aligned}\mathbf{e} &= A\mathbf{x} - \mathbf{b} = \begin{bmatrix} \mathbf{c}_1 & | & \mathbf{c}_2 & | & \dots & | & \mathbf{c}_n \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} - \mathbf{b} \\ &= x_1\mathbf{c}_1 + x_2\mathbf{c}_2 + \dots + x_n\mathbf{c}_n - \mathbf{b}\end{aligned}$$

- At a minimum,

$$\begin{aligned}\frac{\partial E}{\partial x_i} &= \frac{\partial \mathbf{e}}{\partial x_i} \cdot \mathbf{e} + \mathbf{e} \cdot \frac{\partial \mathbf{e}}{\partial x_i} = 2 \frac{\partial \mathbf{e}}{\partial x_i} \cdot \mathbf{e} \\ &= 2 \frac{\partial}{\partial x_i} (x_1\mathbf{c}_1 + \dots + x_n\mathbf{c}_n - \mathbf{b}) \cdot \mathbf{e} = 2\mathbf{c}_i \cdot \mathbf{e} \\ &= 2\mathbf{c}_i^T (A\mathbf{x} - \mathbf{b}) = 0\end{aligned}$$

- or

$$0 = \begin{bmatrix} \mathbf{c}_i^T \\ \vdots \\ \mathbf{c}_n^T \end{bmatrix} (A\mathbf{x} - \mathbf{b}) = A^T (A\mathbf{x} - \mathbf{b}) \Rightarrow A^T A\mathbf{x} = A^T \mathbf{b},$$

where $\mathbf{x} = A^\dagger \mathbf{b}$ and $A^\dagger = (A^T A)^{-1} A^T$ is the *pseudoinverse* of A !



Homogeneous Linear Systems

$$\boxed{A} \quad \boxed{x} = \boxed{0}$$

Square system:

- unique solution: 0
- unless $\text{Det}(A)=0$

$$\begin{array}{|c|} \hline \\ \hline \boxed{A} \\ \hline \\ \hline \end{array} \quad \boxed{x} = \begin{array}{|c|} \hline \\ \hline \boxed{0} \\ \hline \\ \hline \end{array}$$

Rectangular system ??

- 0 is always a solution

Minimize $|Ax|^2$
under the constraint $|x|^2=1$



How do you solve overconstrained homogeneous linear equations ??

$$E = |\mathcal{U}\mathbf{x}|^2 = \mathbf{x}^T (\mathcal{U}^T \mathcal{U}) \mathbf{x}$$

- Orthonormal basis of eigenvectors: $\mathbf{e}_1, \dots, \mathbf{e}_q$.
- Associated eigenvalues: $0 \leq \lambda_1 \leq \dots \leq \lambda_q$.
- Any vector can be written as

$$\mathbf{x} = \mu_1 \mathbf{e}_1 + \dots + \mu_q \mathbf{e}_q$$

for some μ_i ($i = 1, \dots, q$) such that $\mu_1^2 + \dots + \mu_q^2 = 1$.

$$\begin{aligned} E(\mathbf{x}) - E(\mathbf{e}_1) &= \mathbf{x}^T (\mathcal{U}^T \mathcal{U}) \mathbf{x} - \mathbf{e}_1^T (\mathcal{U}^T \mathcal{U}) \mathbf{e}_1 \\ &= \lambda_1^2 \mu_1^2 + \dots + \lambda_q^2 \mu_q^2 - \lambda_1^2 \\ &\geq \lambda_1^2 (\mu_1^2 + \dots + \mu_q^2 - 1) = 0 \end{aligned}$$

The solution is \mathbf{e}_1 .

remember: $\text{EIG}(\mathcal{U}^T \mathcal{U}) = \text{SVD}(\mathcal{U})$, i.e. solution is \mathbf{V}_n



Linear Camera Calibration

Given n points P_1, \dots, P_n with *known* positions and their images p_1, \dots, p_n

$$\begin{matrix} \rightarrow \\ \rightarrow \end{matrix} \begin{pmatrix} u_i \\ v_i \end{pmatrix} = \begin{pmatrix} \frac{\mathbf{m}_1 \cdot \mathbf{P}_i}{\mathbf{m}_3 \cdot \mathbf{P}_i} \\ \frac{\mathbf{m}_2 \cdot \mathbf{P}_i}{\mathbf{m}_3 \cdot \mathbf{P}_i} \end{pmatrix} \iff \begin{pmatrix} \mathbf{m}_1 - u_i \mathbf{m}_3 \\ \mathbf{m}_2 - v_i \mathbf{m}_3 \end{pmatrix} \mathbf{P}_i = 0$$

$$\begin{matrix} \rightarrow \\ \rightarrow \end{matrix} \mathcal{P} \mathbf{m} = 0 \text{ with } \mathcal{P} \stackrel{\text{def}}{=} \begin{pmatrix} \mathbf{P}_1^T & \mathbf{0}^T & -u_1 \mathbf{P}_1^T \\ \mathbf{0}^T & \mathbf{P}_1^T & -v_1 \mathbf{P}_1^T \\ \dots & \dots & \dots \\ \mathbf{P}_n^T & \mathbf{0}^T & -u_n \mathbf{P}_n^T \\ \mathbf{0}^T & \mathbf{P}_n^T & -v_n \mathbf{P}_n^T \end{pmatrix} \text{ and } \mathbf{m} \stackrel{\text{def}}{=} \begin{pmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \mathbf{m}_3 \end{pmatrix} = 0$$



Useful Links

Demo calibration (some links broken):

- <http://mitpress.mit.edu/e-journals/Videre/001/articles/Zhang/CalibEnv/CalibEnv.html>

Bouget camera calibration SW:

- http://www.vision.caltech.edu/bouguetj/calib_doc/

CVonline: Monocular Camera calibration:

- <http://homepages.inf.ed.ac.uk/cgi/rbf/CVONLINE/entries.pl?TAG250>