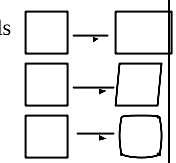
Camera Calibration

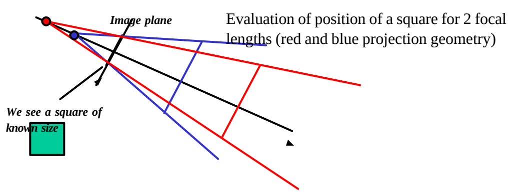
What is Camera Calibration?

- Primarily, finding the quantities internal to the camera that affect the imaging process
  - Position of image center in the image
    - It is typically not at (width/2, height/2) of image
  - Focal length
  - Different scaling factors for row pixels and column pixels
  - Skew factor
  - Lens distortion (pin-cushion effect)



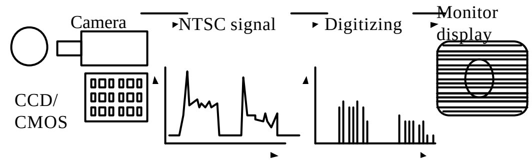
Motivation

- Good calibration is important when we need to
  - Reconstruct a world model: Virtual L.A. project
  - Interact with the world
    - Robot, hand-eye coordination



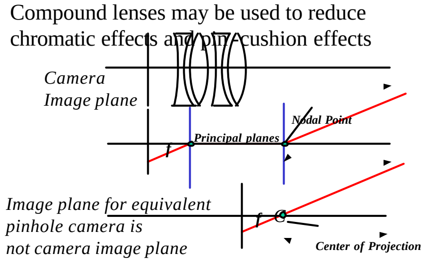
Scaling of Rows and Columns in Image

- Camera pixels are not necessarily square
- Camera output may be analog (NTSC)
- Image may be obtained by digitizing card
  - A/D converter samples NTSC signal



Compound Lens Imaging

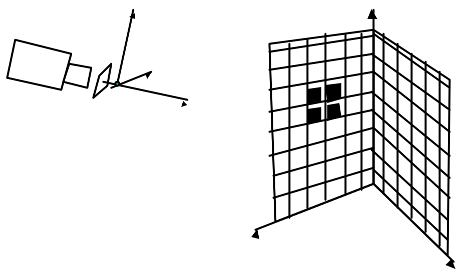
- Inexpensive single lens systems distort image at its periphery
- Compound lenses may be used to reduce chromatic effects and pin-cushion effects



Variety of Techniques

- VERY large literature on the subject
- Work of Roger Tsai influential
- Linear algebra method described here
  - Can be used as initialization for iterative non linear methods.
- Some interesting methods use vanishing points

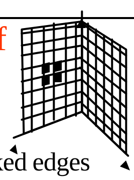
Camera and Calibration Target



Calibration Procedure

- Calibration target: 2 planes at right angle with checkerboard patterns (Tsai grid)
  - We know positions of pattern corners only with respect to a coordinate system of the target
  - We position camera in front of target and find images of corners
  - We obtain equations that describe imaging and contain internal parameters of camera
    - As a side benefit, we find position and orientation of camera with respect to target (camera ***pose***)

Image Processing of Image of Target



- Canny edge detection
- Straight line fitting to detected linked edges
- Intersecting the lines to obtain the image corners
- Matching image corners and 3D target checkerboard corners
  - By counting if whole target is visible in image
- We get pairs (image point)--(world point)

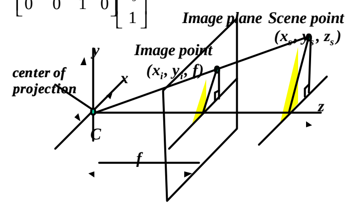
$$(x_i, y_i) \rightarrow (X_i, Y_i, Z_i)$$

Central Projection

If world and image points are represented by homogeneous vectors, central projection is a linear transformation:

$$x_i = f\,\frac{x_s}{z_s} \qquad y_i = f\,\frac{y_s}{z_s}$$

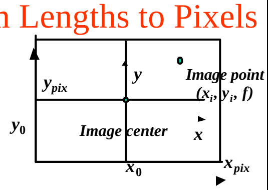
$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_s \\ y_s \\ z_s \\ 1 \end{bmatrix} \qquad x_i = u/w, \;\; y_i = v/w$$



Transformation From Lengths to Pixels

Transformation uses:

- image center (x_0, y_0)
- scaling factors k_x and k_y



$$x_i = f\,\frac{x_s}{z_s} \qquad x_{pix} = k_x x_i + x_0 = f\,k_x \frac{x_s + z_s x_0}{z_s}$$

$$y_i = f\,\frac{y_s}{z_s} \qquad y_{pix} = k_y y_i + y_0 = f\,k_y \frac{y_s + z_s y_0}{z_s}$$

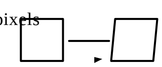
$$\begin{bmatrix} u' \\ v' \\ w' \end{bmatrix} = \begin{bmatrix} a_x & 0 & x_0 & 0 \\ 0 & a_y & y_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_s \\ y_s \\ z_s \\ 1 \end{bmatrix} \text{ with } \begin{array}{l} a_x = f\,k_x \\ a_y = f\,k_y \end{array} \text{ then } \begin{array}{l} x_{pix} = u'/w' \\ y_{pix} = v'/w' \end{array}$$

Internal Camera Parameters

$$\begin{bmatrix} u' \\ v' \\ w' \end{bmatrix} = \begin{bmatrix} a_x & s & x_0 & 0 \\ 0 & a_y & y_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_s \\ y_s \\ z_s \\ 1 \end{bmatrix} \text{ with } \begin{array}{l} a_x = f\,k_x \\ a_y = -f\,k_y \end{array} \qquad \begin{array}{l} x_{pix} = u'/w' \\ y_{pix} = v'/w' \end{array}$$

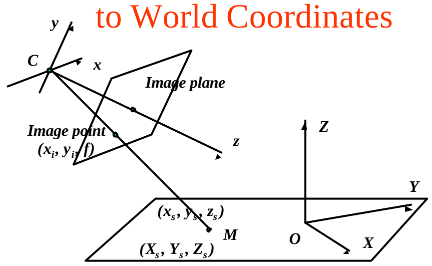
$$\begin{bmatrix} a_x & s & x_0 & 0 \\ 0 & a_y & y_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} a_x & s & x_0 \\ 0 & a_y & y_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} = \mathbf{K}\,[\mathbf{I}_3 \;\; | \;\; \mathbf{0}_3]$$

- a_x and a_y "focal lengths" in pixels
- x_0 and y_0 coordinates of image center in pixels

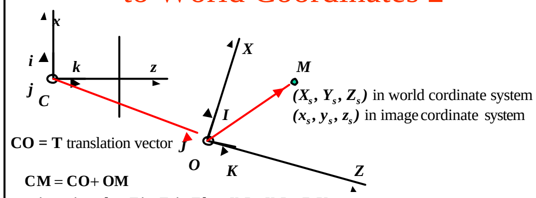


- Added parameter S is skew parameter
- **K** is called *calibration matrix*. **Five degrees of freedom.**
  - **K** is a 3x3 upper triangular matrix

From Camera Coordinates to World Coordinates



From Camera Coordinates to World Coordinates 2



$\mathbf{CM} = \mathbf{CO} + \mathbf{OM}$

$x_s\mathbf{i} + y_s\mathbf{j} + z_s\mathbf{k} = T_x\mathbf{i} + T_y\mathbf{j} + T_z\mathbf{k} + X_s\mathbf{I} + Y_s\mathbf{J} + Z_s\mathbf{K}$

$x_s = T_x + X_s\,\mathbf{I.i} + Y_s\,\mathbf{J.i} + Z_s\,\mathbf{K.i}$

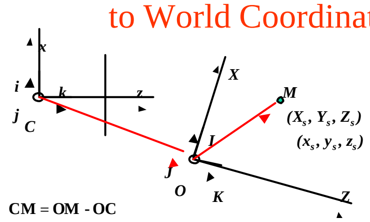
$$\begin{bmatrix} x_s \\ y_s \\ z_s \end{bmatrix} = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} + \begin{bmatrix} \mathbf{I.i} & \mathbf{J.i} & \mathbf{K.i} \\ \mathbf{I.j} & \mathbf{J.j} & \mathbf{K.j} \\ \mathbf{I.k} & \mathbf{J.k} & \mathbf{K.k} \end{bmatrix} \begin{bmatrix} X_s \\ Y_s \\ Z_s \end{bmatrix}$$

Homogeneous Coordinates

$$\begin{bmatrix} x_s \\ y_s \\ z_s \end{bmatrix} = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} + \begin{bmatrix} \mathbf{I.i} & \mathbf{J.i} & \mathbf{K.i} \\ \mathbf{I.j} & \mathbf{J.j} & \mathbf{K.j} \\ \mathbf{I.k} & \mathbf{J.k} & \mathbf{K.k} \end{bmatrix} \begin{bmatrix} X_s \\ Y_s \\ Z_s \end{bmatrix}$$

$$\begin{bmatrix} x_s \\ y_s \\ z_s \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{I.i} & \mathbf{J.i} & \mathbf{K.i} & T_x \\ \mathbf{I.j} & \mathbf{J.j} & \mathbf{K.j} & T_y \\ \mathbf{I.k} & \mathbf{J.k} & \mathbf{K.k} & T_z \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} X_s \\ Y_s \\ Z_s \\ 1 \end{bmatrix} \qquad \begin{bmatrix} x_s \\ y_s \\ z_s \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R} & \mathbf{T} \\ \mathbf{0}_3^{\mathbf{T}} & 1 \end{bmatrix} \begin{bmatrix} X_s \\ Y_s \\ Z_s \\ 1 \end{bmatrix}$$

From Camera Coordinates to World Coordinates 3



$\mathbf{CM} = \mathbf{OM} - \mathbf{OC}$

$x_s\mathbf{i} + y_s\mathbf{j} + z_s\mathbf{k} = (X_s - X_C)\,\mathbf{I} + (Y_s - Y_C)\,\mathbf{J} + (Z_s - Z_C)\,\mathbf{K}$

$x_s = (X_s - X_C)\,\mathbf{I.i} + (Y_s - Y_C)\,\mathbf{J.i} + (Z_s - Z_C)\,\mathbf{K.i}$

$\mathbf{x}_{cam} = \mathbf{R}(\mathbf{X} - \tilde{\mathbf{C}}) \qquad (\mathbf{T} = -\mathbf{R}\,\tilde{\mathbf{C}})$

$\tilde{\mathrm{C}}$ is vector OC expressed in world coordinate system

Homogeneous Coordinates 2

- Here we use $-\mathbf{R}\tilde{\mathbf{C}}$ instead of $\mathbf{T}$

$$\begin{bmatrix} x_s \\ y_s \\ z_s \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R} & -\mathbf{R}\tilde{\mathbf{C}} \\ \mathbf{0}_3^{\mathbf{T}} & 1 \end{bmatrix} \begin{bmatrix} X_s \\ Y_s \\ Z_s \\ 1 \end{bmatrix}$$

Linear Transformation from World Coordinates to Pixels

- Combine camera projection and coordinate transformation matrices into a single matrix **P**

$$\begin{bmatrix} u' \\ v' \\ w' \end{bmatrix} = \mathbf{K}[\mathbf{I}_3 \mid \mathbf{0}_3] \begin{bmatrix} x_s \\ y_s \\ z_s \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} x_s \\ y_s \\ z_s \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R} & -\mathbf{R}\tilde{\mathbf{C}} \\ \mathbf{0}_3^{\mathbf{T}} & 1 \end{bmatrix} \begin{bmatrix} X_s \\ Y_s \\ Z_s \\ 1 \end{bmatrix}$$

$$\Rightarrow \begin{bmatrix} u' \\ v' \\ w' \end{bmatrix} = \mathbf{K}[\mathbf{I}_3 \mid \mathbf{0}_3] \begin{bmatrix} \mathbf{R} & -\mathbf{R}\tilde{\mathbf{C}} \\ \mathbf{0}_3^{\mathbf{T}} & 1 \end{bmatrix} \begin{bmatrix} X_s \\ Y_s \\ Z_s \\ 1 \end{bmatrix}$$

$$\Rightarrow \begin{bmatrix} u' \\ v' \\ w' \end{bmatrix} = \mathbf{P} \begin{bmatrix} X_s \\ Y_s \\ Z_s \\ 1 \end{bmatrix} \qquad \mathbf{x} = \mathbf{P}\mathbf{X}$$

Properties of Matrix **P**

- Further simplification of **P**:

$$\mathbf{x} = \mathbf{K}[\mathbf{I}_3 \;|\; \mathbf{0}_3]\begin{bmatrix} \mathbf{R} & -\mathbf{R}\tilde{\mathbf{C}} \\ \mathbf{0}_3^T & 1 \end{bmatrix}\mathbf{X}$$

$$[\mathbf{I}_3 \;|\; \mathbf{0}_3]\begin{bmatrix} \mathbf{R} & -\mathbf{R}\tilde{\mathbf{C}} \\ \mathbf{0}_3^T & 1 \end{bmatrix} = [\mathbf{R} \;\; -\mathbf{R}\tilde{\mathbf{C}}] = \mathbf{R}[\mathbf{I}_3 \;|\; -\tilde{\mathbf{C}}]$$

$$\mathbf{x} = \mathbf{K}\mathbf{R}[\mathbf{I}_3 \;|\; -\tilde{\mathbf{C}}]\mathbf{X}$$

$$\boxed{\mathbf{P} = \mathbf{K}\mathbf{R}[\mathbf{I}_3 \;|\; -\tilde{\mathbf{C}}]}$$

- **P** has 11 degrees of freedom:
  - 5 from triangular calibration matrix **K**, 3 from **R** and 3 from $\tilde{\mathbf{C}}$
- **P** is a fairly general 3 x 4 matrix
  - left 3x3 submatrix **KR** is non-singular

Calibration

- 1. Estimate matrix **P** using scene points and their images
- 2. Estimate the interior parameters and the exterior parameters

$$\boxed{\mathbf{P} = \mathbf{K}\mathbf{R}[\mathbf{I}_3 \;|\; -\tilde{\mathbf{C}}]}$$

  - Left 3x3 submatrix of **P** is product of upper-triangular matrix and orthogonal matrix

Finding Camera Translation

- Find homogeneous coordinates of *C* in the scene
- **C** is the null vector of matrix **P**
  - **P C** = 0:

$$\mathbf{P} = \mathbf{K}\mathbf{R}[\mathbf{I}_3 \;|\; -\tilde{\mathbf{C}}] \qquad \begin{bmatrix} 1 & 0 & 0 & X_c \\ 0 & 1 & 0 & Y_c \\ 0 & 0 & 1 & Z_c \end{bmatrix}\begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

- Find null vector **C** of **P** using SVD
  - **C** is the unit singular vector of **P** corresponding to the smallest singular value (the last column of **V**, where **P** = **U D V**$^{\mathbf{T}}$ is the SVD of **P**)

Finding Camera Orientation and Internal Parameters

- Left 3x3 submatrix **M** of **P** is of form **M=K R**
  - **K** is an upper triangular matrix
  - **R** is an orthogonal matrix
- Any non-singular square matrix **M** can be decomposed into the product of an upper-triangular matrix **K** and an orthogonal matrix **R** using the RQ factorization
  - Similar to QR factorization but order of 2 matrices is reversed

RQ Factorization of **M**

$$\mathbf{R}_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c & -s \\ 0 & s & c \end{bmatrix}, \quad \mathbf{R}_y = \begin{bmatrix} c' & 0 & s' \\ 0 & 1 & 0 \\ -s' & 0 & c' \end{bmatrix}, \quad \mathbf{R}_z = \begin{bmatrix} c'' & -s'' & 0 \\ s'' & c'' & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

- Compute $c = -\dfrac{m_{33}}{(m_{32}^2 + m_{33}^2)^{1/2}}, \quad s = \dfrac{m_{32}}{(m_{32}^2 + m_{33}^2)^{1/2}}$
- Multiply **M** by $\mathbf{R}_x$. The resulting term at (3,2) is zero because of the values selected for *c* and *s*
- Multiply the resulting matrix by $\mathbf{R}_y$, after selecting *c'* and *s'* so that the resulting term at position (3,1) is set to zero
- Multiply the resulting matrix by $\mathbf{R}_z$, after selecting *c''* and *s''* so that the resulting term at position (2,1) is set to zero

$$\mathbf{M}\mathbf{R}_x\mathbf{R}_y\mathbf{R}_z = \mathbf{K} \Rightarrow \mathbf{M} = \mathbf{K}\mathbf{R}_z^T\mathbf{R}_y^T\mathbf{R}_x^T = \mathbf{K}\mathbf{R}$$

Computing Matrix **P**

- Use corresponding image and scene points
  - 3D points $\mathbf{X}_i$ in world coordinate system
  - Images $\mathbf{x}_i$ of $\mathbf{X}_i$ in image
- Write $\mathbf{x}_i = \mathbf{P}\,\mathbf{X}_i$ for all *i*
- Similar problem to finding projectivity matrix **H** (i.e. homography) in homework

Improved Computation of P

- $\mathbf{x_i} = \mathbf{P}\ \mathbf{X_i}$ involves homogeneous coordinates, thus $\mathbf{x_i}$ and $\mathbf{P}\ \mathbf{X_i}$ just have to be proportional: $\mathbf{x}_i \times \mathbf{P}\mathbf{X}_i = 0$
- Let $\mathbf{p}_1^T$, $\mathbf{p}_2^T$, $\mathbf{p}_3^T$ be the 3 row vectors of $\mathbf{P}$

$$\mathbf{P}\ \mathbf{X}_i = \begin{bmatrix} \mathbf{p}_1^T \mathbf{X}_i \\ \mathbf{p}_2^T \mathbf{X}_i \\ \mathbf{p}_3^T \mathbf{X}_i \end{bmatrix} \qquad \mathbf{x}_i \times \mathbf{P}\mathbf{X}_i = \begin{bmatrix} v'_i\,\mathbf{p}_3^T\mathbf{X}_i - w'_i\,\mathbf{p}_2^T\mathbf{X}_i \\ w'_i\,\mathbf{p}_1^T\mathbf{X}_i - u'_i\,\mathbf{p}_3^T\mathbf{X}_i \\ u'_i\,\mathbf{p}_2^T\mathbf{X}_i - v'_i\,\mathbf{p}_1^T\mathbf{X}_i \end{bmatrix}$$

$$\Rightarrow \begin{bmatrix} \mathbf{0}_4^T & -w'_i\,\mathbf{X}_i^T & v'_i\,\mathbf{X}_i^T \\ w'_i\,\mathbf{X}_i^T & \mathbf{0}_4^T & -u'_i\,\mathbf{X}_i^T \\ -v'_i\,\mathbf{X}_i^T & u'_i\,\mathbf{X}_i^T & \mathbf{0}_4^T \end{bmatrix} \begin{bmatrix} \mathbf{p}_1 \\ \mathbf{p}_2 \\ \mathbf{p}_3 \end{bmatrix} = 0 \qquad \begin{bmatrix} \mathrm{p}_1 \\ \mathrm{p}_2 \\ \mathrm{p}_3 \end{bmatrix} \text{ is a } 12\times 1 \text{ vector}$$

Improved Computation of P, cont’d

- Third row can be obtained from sum of u'_i times first row - v'_i times second row

$$\begin{bmatrix} \mathbf{0}_4^T & -w'_i\,\mathbf{X}_i^T & v'_i\,\mathbf{X}_i^T \\ w'_i\,\mathbf{X}_i^T & \mathbf{0}_4^T & -u'_i\,\mathbf{X}_i^T \\ -v'_i\,\mathbf{X}_i^T & u'_i\,\mathbf{X}_i^T & \mathbf{0}_4^T \end{bmatrix} \begin{bmatrix} \mathbf{p}_1 \\ \mathbf{p}_2 \\ \mathbf{p}_3 \end{bmatrix} = 0$$

- So we get 2 independent equations in 11 unknowns (ignoring scale)
- With 6 point correspondences, we get enough equations to compute matrix $\mathbf{P}$

$$\mathbf{A}\ \mathbf{p} = 0$$

Solving $\mathbf{A}\ \mathbf{p} = 0$

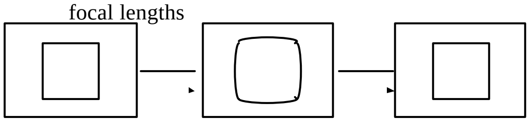
- Linear system $\mathbf{A}\ \mathbf{p} = 0$
- When possible, have at least 5 times as many equations as unknowns (28 points)
- Minimize $\|\ \mathbf{A}\ \mathbf{p}\ \|$ with the constraint $\|\ \mathbf{p}\ \| = 1$
  - P is the unit singular vector of **A** corresponding to the smallest singular value (the last column of **V**, where $\mathbf{A} = \mathbf{U}\ \mathbf{D}\ \mathbf{V}^T$ is the SVD of **A**)
- Called Direct Linear Transformation (DLT)

Improving **P** Solution with Nonlinear Minimization

- Find **p** using DLT
- Use as initialization for nonlinear minimization of $\sum_i d(\mathbf{x}_i, \mathbf{P}\mathbf{X}_i)^2$
  - Use Levenberg-Marquardt iterative minimization

Radial Distortion

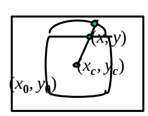
- We have assumed that lines are imaged as lines
- Not quite true for real lenses
  - Significant error for cheap optics and for short focal lengths



Scene Image Corrected image

Radial Distortion Modeling

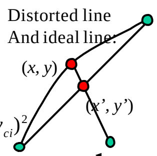
- In pixel cordinates the correction is written



$$x_c - x_0 = L(r)(x - x_0)$$
$$y_c - y_0 = L(r)(y - y_0)$$

with

$$r^2 = (x - x_0)^2 + (y - y_0)^2$$
$$L(r) = 1 + k_1 r + k_2 r^2 + \ldots$$



- Minimize $f(k_1, k_2) = \sum_i (x'_i - x_{ci})^2 + (y'_i - y_{ci})^2$ using lines known to be straight
  (x’,y’) is radial projection of (x,y) on straight line

References

- Multiple View Geometry in Computer Vision, R. Hartley and A. Zisserman, Cambridge University Press, 2000, pp. 138-183
- Three-Dimensional Computer Vision: A Geometric Approach, O. Faugeras, MIT Press, 1996, pp. 33-68
- "A Versatile Camera Calibration Technique for 3D Machine Vision", R. Y. Tsai, IEEE J. Robotics & Automation, RA-3, No. 4, August 1987, pp. 323-344