{ CSE 252B Lecture 04 }

OWEN JOW | JANUARY 16, 2019

2D (Planar) Transformations

Euclidean (rotation + translation)

INHOMOGENEOUS

$$\underset{}{\tilde{\underline{x}}'} = \underset{2\times2}{\underline{R}}\,\underset{2\times1}{\tilde{\underline{x}}} + \underset{2\times1}{\underline{t}}$$

3 DOF (one for $\underline{R}$, two for $\underline{t}$)
$\underline{R} \in SO(2)$

HOMOGENEOUS

$$\tilde{\underline{x}}' = \underbrace{\left[\underset{2\times2}{\underline{R}} \mid \underset{2\times1}{\underline{t}}\right]}_{2\times3} \underbrace{\begin{bmatrix} \tilde{\underline{x}} \\ 1 \end{bmatrix}}_{3\times1} \rightarrow \underline{x}' = \underbrace{\begin{bmatrix} \underline{R} & \underline{t} \\ \underline{0}^T & 1 \end{bmatrix}}_{3\times3} \underbrace{\underline{x}}_{3\times1} \rightarrow \underline{x}' = \underline{H}_E\,\underline{x}$$

Similarity (rotation + translation + scale)

INHOMOGENEOUS

$$\tilde{\underline{x}}' = s\,\underline{R}\,\tilde{\underline{x}} + \underline{t}$$

4 DOF (one for $\underline{R}$, two for $\underline{t}$, one for s)

HOMOGENEOUS

$$\tilde{\underline{x}}' = \left[s\underline{R} \mid \underline{t}\right] \begin{bmatrix} \tilde{\underline{x}} \\ 1 \end{bmatrix} \rightarrow \underline{x}' = \begin{bmatrix} s\underline{R} & \underline{t} \\ \underline{0}^T & 1 \end{bmatrix} \underline{x} \rightarrow \underline{x}' = \underline{H}_S\,\underline{x}$$

Affine (general linear transformation + translation)

INHOMOGENEOUS

$$\tilde{\underline{x}}' = \underline{A}\,\tilde{\underline{x}} + \underline{t}$$

6 DOF (four for $\underline{A}$, two for $\underline{t}$)
introduce non-uniform scaling / skew, preserve parallelism

HOMOGENEOUS

$$\tilde{\underline{x}}' = \left[\underline{A} \mid \underline{t}\right] \begin{bmatrix} \tilde{\underline{x}} \\ 1 \end{bmatrix} \rightarrow \underline{x}' = \begin{bmatrix} \underline{A} & \underline{t} \\ \underline{0}^T & 1 \end{bmatrix} \underline{x} \rightarrow \underline{x}' = \underline{H}_A\,\underline{x}$$

Projective (general 3×3 homogeneous matrix)

HOMOGENEOUS

$$\underline{x}' = \begin{bmatrix} \underline{A} & \underline{t} \\ \underline{v}^T & v \end{bmatrix} \underline{x} \rightarrow \underline{x}' = \underline{H}_P\,\underline{x}$$

8 DOF (defined up to scale)

3D Transformations ($\underline{H}_{4\times4}$)

	Degrees of freedom	Notes
$\underline{R} \in SO(3)$	3	$\det(\underline{R}) = +1$ (if in special orthogonal group)
Euclidean	6	
Similarity	7	uniform scaling adds 1 DOF
Affine	12	3×3 inhomog. matrix + 3D translation
Projective	15	

Transformations

2D

$\underline{x}' = \underline{H}\underline{x}$	point
$\underline{\ell}' = \underline{H}^{-T}\underline{\ell}$	line
$\underline{C}' = \underline{H}^{-T}\underline{C}\,\underline{H}^{-1}$	conic
$\underline{C}^{*\prime} = \underline{H}\underline{C}^{*}\underline{H}^{T}$	dual conic

3D

$\underline{X}' = \underline{H}\underline{X}$	point
$\underline{\pi}' = \underline{H}^{-T}\underline{\pi}$	plane
$\underline{Q}' = \underline{H}^{-T}\underline{Q}\,\underline{H}^{-1}$	quadric
$\underline{Q}^{*\prime} = \underline{H}\underline{Q}^{*}\underline{H}^{T}$	dual quadric
$\underline{L}' = \underline{H}\underline{L}\underline{H}^{T}$	line
$\underline{L}^{*\prime} = \underline{H}^{-T}\underline{L}^{*}\underline{H}^{-1}$	dual line

Projections

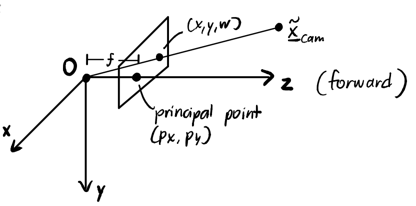
Imaging geometry : mostly concerned with 3D world ⟷ 2D plane

Projective (Pinhole) Camera

models real-world cameras closely enough that we use this for everything

3D ($\underline{X}_{cam}$) to 2D ($\underline{x}$) Projection

camera coordinate frame



camera calibration matrix; canonical projection matrix

$$\underbrace{\begin{bmatrix} x \\ y \\ w \end{bmatrix}}_{3\times1} = \underbrace{\begin{bmatrix} f & 0 & p_x \\ 0 & f & p_y \\ 0 & 0 & 1 \end{bmatrix}}_{3\times3} \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}}_{3\times4} \begin{bmatrix} \tilde{x}_{cam} \\ \tilde{y}_{cam} \\ \tilde{z}_{cam} \\ 1 \end{bmatrix}$$

$$\underline{x} = \underline{K}\,[\underline{I} \mid \underline{0}]\,\underline{X}_{cam}$$

3D ($\underline{X}$) to 3D ($\underline{X}_{cam}$) Camera Frame Transformation

$$\underline{x} = \underline{K}\,[\underline{I} \mid \underline{0}]\,\underline{H}_E\,\underline{X}$$

; $\underline{X}_{cam} = \underline{H}_E\underline{X}$, for $\underline{H}_E = \begin{bmatrix} \underline{R} & \underline{t} \\ \underline{0}^T & 1 \end{bmatrix}$ extrinsic matrix ($\underline{H}_E : \underline{X} \longmapsto \underline{X}_{cam}$)

$$\underline{x} = \underline{K}\,[\underline{I} \mid \underline{0}] \begin{bmatrix} \underline{R} & \underline{t} \\ \underline{0}^T & 1 \end{bmatrix} \underline{X}$$

$$\underline{x} = \underset{3\times3}{\underline{K}}\;\underset{3\times4}{[\underline{R}\;\underline{t}]}\;\underline{X}$$

$$\underline{x} = \underline{P}\underline{X}$$

; where $\underline{P} = \underbrace{\underline{K}}_{5\text{ DOF}}\,[\underbrace{\underline{R}}_{3\text{ DOF}} \mid \underbrace{\underline{t}}_{3\text{ DOF}}]$

(homogeneous camera projection matrix)

Some Descriptions

$\mathbf{R}, \mathbf{t}$ from world coord frame to cam coord frame
$\mathbf{K}$ camera calibration matrix
$\mathbf{X}$ world point (3D)
$\mathbf{X}_{cam}$ world point in camera coordinates (3D)
$\mathbf{x}$ image point (2D)

Alternative $\mathbf{K}$ Parameterization

$$\mathbf{K} = \begin{bmatrix} \alpha_x & 0 & x_0 \\ 0 & \alpha_y & y_0 \\ 0 & 0 & 1 \end{bmatrix}$$

where $\alpha_x = f\mu_x$ is focal length in pixel dimensions

$$\alpha_y = f\mu_y$$
$$x_0 = \mu_x p_x$$
$$y_0 = \mu_y p_y$$

μ_x and μ_y are the number of pixels per unit distance in x and y directions

World and Camera Coordinate Frames

Let $\tilde{\mathbf{X}}_{cam} = \mathbf{R}\tilde{\mathbf{X}} + \mathbf{t}$ be the transf from world to cam coord frame

Let $\tilde{\mathbf{C}}$ be the camera center in world coord frame

Let $\tilde{\mathbf{C}}_{cam} = \mathbf{0}$ be the camera center in cam coord frame

Then

$$\tilde{\mathbf{C}}_{cam} = \mathbf{R}\tilde{\mathbf{C}} + \mathbf{t}$$
$$\mathbf{0} = \mathbf{R}\tilde{\mathbf{C}} + \mathbf{t}$$

$$\mathbf{t} = -\mathbf{R}\tilde{\mathbf{C}}$$

$$\mathbf{R}\tilde{\mathbf{C}} = -\mathbf{t}$$
$$\tilde{\mathbf{C}} = -\mathbf{R}^T\mathbf{t}$$

Substituting back into the definition of the camera projection matrix,

$$\mathbf{P} = \mathbf{K}[\mathbf{R} \mid \mathbf{t}]$$
$$\mathbf{P} = \mathbf{K}[\mathbf{R} \mid -\mathbf{R}\tilde{\mathbf{C}}]$$
$$\mathbf{P} = \mathbf{K}\mathbf{R}[\mathbf{I} \mid -\tilde{\mathbf{C}}]$$

Calibrated vs. Uncalibrated Cases

$$\mathbf{x} = \mathbf{K}[\mathbf{R} \mid \mathbf{t}]\mathbf{X}$$

$\mathbf{x} = \mathbf{P}\mathbf{X}$ where $\mathbf{P} = \mathbf{K}[\mathbf{R} \mid \mathbf{t}]$ (camera projection matrix)

When we have a calibrated camera (know $\mathbf{K}$), we can do

$$\mathbf{K}^{-1}\mathbf{x} = [\mathbf{R} \mid \mathbf{t}]\mathbf{X}$$
$$\hat{\mathbf{x}} = \hat{\mathbf{P}}\mathbf{X}$$

where $\hat{\mathbf{x}} = \mathbf{K}^{-1}\mathbf{x}$ normalized coordinates

$\hat{\mathbf{P}} = [\mathbf{R} \mid \mathbf{t}]$ normalized cam projection matrix

When the camera is calibrated, we'll typically try to use normalized coordinates.
When the camera is uncalibrated, we'll do things as before (just use image coordinates, i.e. $\underline{x}$, and the cam proj matrix, i.e. $\underline{P}$).

Conversion Between Regular and Normalized Forms

$$\underline{P} = \underline{K}[\underline{R} \mid \underline{t}] = \underline{K}\hat{\underline{P}} \qquad \underline{x} = \underline{K}[\underline{R} \mid \underline{t}]\underline{X} = \underline{K}\hat{\underline{x}}$$

$$\hat{\underline{P}} = [\underline{R} \mid \underline{t}] = \underline{K}^{-1}\underline{P} \qquad \hat{\underline{x}} = [\underline{R} \mid \underline{t}]\underline{X} = \underline{K}^{-1}\underline{x}$$