Lecture 2 - Homogeneous Coordinates

Motivations for Leaving the Cartesian Plane

- Cameras generate projected img
- Euclidian geometry is suboptimal to describe that
  ↳ the math can get complicated
- Projective geometry is an alternative algebraic representation of geometric objects & transformations
  ↳ math becomes simpler
  ↳ doesn't change geometric relations

Homogeneous Coordinates

- H.C. are a system of coordinates used in projective geometry
- Formulas involving H.C. are much simpler
  1) Points at infinity can be represented using finite coordinates
  2) A single mtx can represent affine transformations & projective transformations
     ↳ like shear & rotate, etc. (by a matrix multiplication)

Definition

- The representation x of a geometric object is homogeneous if x and λx represent the same object for $\lambda \neq 0$

e.g. $\underbrace{x = \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} wx \\ wy \\ w \end{bmatrix} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}}_{\text{homogeneous}} \quad \therefore \underbrace{\begin{bmatrix} u/w \\ v/w \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix}}_{\text{Euclidian}}$

→ it's sufficient to just add a new dimension & assign it 1 to make it homogeneous

Center of Coordinate Systems;

$$O_2 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \qquad O_3 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

Infinitively Distant Objects

$$X_\infty = \begin{bmatrix} u \\ v \\ 0 \end{bmatrix}$$

→ great tool when working with bearing-only sensors such as cameras

3D Points

homogeneous — Euclidean

$$X = \begin{bmatrix} u \\ v \\ w \\ t \end{bmatrix} = \begin{bmatrix} u/t \\ v/t \\ w/t \\ 1 \end{bmatrix} \longrightarrow \begin{bmatrix} u/t \\ v/t \\ w/t \end{bmatrix}$$

Transformations

- a projective transformation is an invertible linear mapping

$$x' = MX$$

↳ with this mtx "M" we can express transformations that can't be done w/ a single mtx transform. in Cartesian space

Important Transformations ($\mathbb{P}^3$)

- General Projective mappings:

$$\underset{(4\times4)}{x' = MX}$$

- Translation (3 Params):

$$M = \lambda \begin{bmatrix} I & t \\ 0^T & 1 \end{bmatrix} \text{ where; } I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \ t = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}, \ 0 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

↳ instead of adding, we can transform by matrix mult.

↳ can easily perform large chains of transformations lect 02 - 23:34 for list of $\mathbb{P}^2$ transformations

- Rotation (3 Params);

note that the transformation vector is 0

$$M = \lambda \left[\begin{array}{c|c} R & 0 \\ \hline 0^T & 1 \end{array}\right]$$ where R = rotation mtx

Recap - Rotation Matrices

$$R^{2D}(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

$$R_x^{3D}(w) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(w) & -\sin(w) \\ 0 & \sin(w) & \cos(w) \end{bmatrix} \qquad R_y^{3D}(\theta) = \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & \\ -\sin(\theta) & 0 & \cos(\theta) \end{bmatrix}$$

$$R_z^{3D}(k) = \begin{bmatrix} \cos(k) & -\sin(k) & 0 \\ \sin(k) & \cos(k) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\boxed{R^{3D}(w, \theta, k) = R_z^{3D}(k) R_y^{3D}(\theta) R_x^{3D}(w)}$$

↳ a standard way (not only way)

- Rigid Body Transformation / Motion Transformation (6 Params)

↳ 3 translation + 3 Rotation

$$M = \lambda \left[\begin{array}{c|c} R & t \\ \hline 0^T & 1 \end{array}\right]$$ a.k.a $\boxed{\text{motions}}$

↳ using extensively in course

- Similarity Transformations (7 Params; 3 trans, 3 Rot, 1 scale)

$$M = \lambda \left[\begin{array}{c|c} mR & t \\ \hline 0^T & 1 \end{array}\right]$$

- Affine Transformations (12 Params; 3 trans, 3 rot, 3 Scale, 3 Shear)

$$M = \lambda \left[\begin{array}{c|c} A & t \\ \hline 0^T & 1 \end{array}\right]$$

Transformations

- Inverting an OP ; $x' = Mx$ → this works cuz M is constructed from R & t and such are always invertible

$$x = M^{-1}x'$$

- Note ; chaining via mtx mult is not commutative

$$\left.\begin{aligned} x' &= M_1 M_2 x \\ &\neq M_2 M_1 x \end{aligned}\right\} \text{ eg. order of rotations}$$