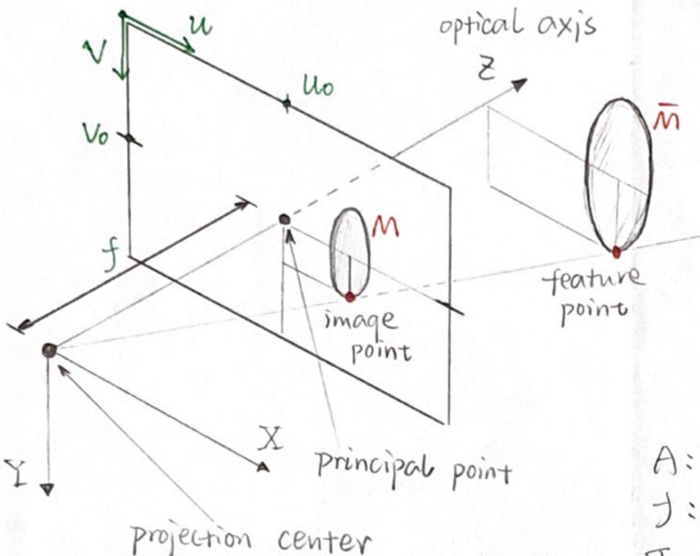
Lecture 1: Sensors

Internal State Sensors
External State Sensors

Lecture 2: Vision



Given pixel coordinate p.
recover normalized coordinate m:

$$m = A^{-1} p$$

Cannot to recover $\bar{m}$ without depth information.

① $\bar{m}$ in Euclidean coordinate in camera frame $\mathcal{F}$

② projected to image plane (m)

③ mapped to pixel coordinate by calibration matrix A

3D feature point $\bar{m} = [x, y, z]^T$

image point $m = \frac{\bar{m}}{z} = [\frac{x}{z}, \frac{y}{z}, 1]$

⇓ discretized

$p = [u, v, 1]^T$

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f\sigma_x & -f\sigma_x \tan\alpha & u_0 \\ 0 & f\sigma_y \sec\alpha & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{x}{z} \\ \frac{y}{z} \\ 1 \end{bmatrix}$$

$p = A m$

A: intrinsic calibration matrix

f: camera focal length

σ_x, σ_y: size of pixels in focal length

$[u_0, v_0]$: principle point in pixel

α: skew angle (≈ 0)

Consider a point $m_w = [x_w, y_w, z_w]^T$ in World frame $\mathcal{F}_w$:

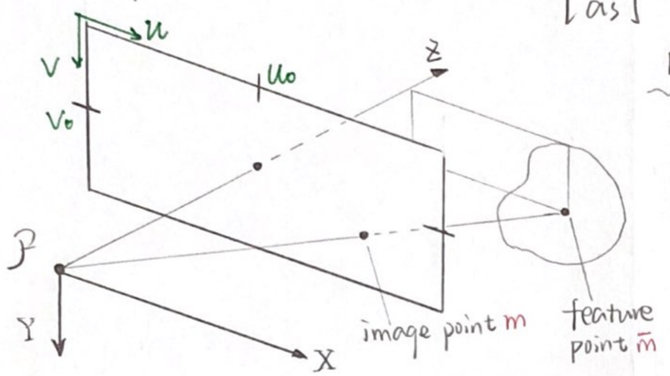
$$\underbrace{\begin{bmatrix} u \\ v \\ 1 \end{bmatrix}}_{p} = \underbrace{\begin{bmatrix} f\sigma_x & -f\sigma_x \tan\alpha & u_0 \\ 0 & f\sigma_y \sec\alpha & v_0 \\ 0 & 0 & 1 \end{bmatrix}}_{\text{intrinsic calibration matrix } A} \underbrace{\underbrace{\begin{bmatrix} \frac{1}{z} & 0 & 0 & 0 \\ 0 & \frac{1}{z} & 0 & 0 \\ 0 & 0 & \frac{1}{z} & 0 \end{bmatrix}}_{\text{projection matrix}} \underbrace{\left[\begin{array}{c|c} R & T \\ \hline 0\ 0\ 0 & 1 \end{array}\right]}_{\text{extrinsic calibration matrix}} \underbrace{\begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix}}_{m_w}}_{m}$$

Lecture 3: Structure & Pose Estimation

SVD: $\underset{m\times n}{M} = \underset{m\times m}{U}\ \underset{m\times n}{\Sigma}\ \underset{n\times n}{V}$ $\Sigma = \mathrm{diag}\{\sigma_1, \cdots, \sigma_m\}$

Skew-symmetric matrix: $a = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} \Rightarrow [a]_\times = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}$

$[a]_\times b = a \times b$



$\bar{m} = [x, y, z]^T$.

$m = [\frac{x}{z}, \frac{y}{z}, 1]^T = [m_x, m_y, 1]^T$.

⇓ map to pixel coordinate

$p = Am$

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f\sigma_x & -f\sigma_x \tan\alpha & u_0 \\ 0 & f\sigma_y \sec\alpha & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} m_x \\ m_y \\ 1 \end{bmatrix}.$$

Pose Reconstruction:

two images with matched feature points

⇒ reconstruct relative position and orientation

N feature points are matched:

frame $\mathcal{F}$: $\bar{m}_j = [x_j, y_j, z_j]^T$.

frame $\mathcal{F}^*$: $\bar{m}_j^* = [x_j^*, y_j^*, z_j^*]^T$.

related in camera frame:

$\bar{m}_j = R\bar{m}_j^* + X$ $(\mathcal{F}^* \to \mathcal{F})$

8-point algorithm:

$E = \begin{bmatrix} e_1 & e_2 & e_3 \\ e_4 & e_5 & e_6 \\ e_7 & e_8 & e_9 \end{bmatrix} \Rightarrow e = [e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8, e_9]^T$,

$M_j = [m_x^* m_x, m_x^* m_y, m_x^*, m_y^* m_x, m_y^* m_y, m_y^*, m_x, m_y, 1]$.

epipolar constraint: $M_j e = 0$ (for one feature point)

$Me = 0$ (for 8 feature points)

Essential Matrix & 8-point Algorithm:

至少8个点 matched，任意4点不共面 (coplanar).

$m_j(t) = \left[\frac{x_j(t)}{z_j(t)}, \frac{y_j(t)}{z_j(t)}, 1\right]^T$.

$m_j^* = \left[\frac{x_j^*}{z_j^*}, \frac{y_j^*}{z_j^*}, 1\right]^T$, $j = 1, 2, \cdots, 8$

⇒ $z_j m_j = z_j^* R m_j^* + X$

同乘：$[X]_\times m = X \times m$

$z_j [X]_\times m_j = z_j^* [X]_\times R m_j^* + \underbrace{[X]_\times X}_{0}$

同乘 m_j^T：

$z_j \underbrace{m_j^T [X]_\times m_j}_{0} = z_j^* m_j^T [X]_\times R m_j^*$

⇒ $z_j^* m_j^T [X]_\times R m_j^* = 0$, $m_j^T E m_j^*$

⇒ Essential matrix $E = [X]_\times R$

8-point algorithm

$$E = [X]_\times R = \begin{bmatrix} e_1 & e_2 & e_3 \\ e_4 & e_5 & e_6 \\ e_7 & e_8 & e_9 \end{bmatrix}.$$

$e = [e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8, e_9]^T$.

$M_j = [m_x^* m_x, m_x^* m_y, m_x^*, m_y^* m_x, m_y^* m_y, m_y^*, m_x, m_y, 1]$.

$M_j e = 0$. $Me = 0$

$E = U \Sigma V^T$, $\Sigma = \mathrm{diag}(\sigma_1, \sigma_2, \sigma_3)$.

Replace Σ with $\Sigma' = \mathrm{diag}(1, 1, 0)$

Define a rotation matrix:

$$R_z(\pm \tfrac{\pi}{2}) = \begin{bmatrix} 0 & \mp 1 & 0 \\ \pm 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \text{ SVD: } E = U \Sigma V^T$$

$[x']_\times = U R_z(\pm \tfrac{\pi}{2}) \Sigma U^T$.

$R = U R_z(\pm \tfrac{\pi}{2}) V^T$.

two solutions

$[x']_\times = \lambda [x]_\times$ with scale factor λ.

Repeat with $\Sigma' = \mathrm{diag}(-1, -1, 0)$

another two solutions

---

3 solutions is physically impossible.

test eqn: $\lambda m_j = \lambda^* R m_j^* + x'$

for 3 solutions, $\lambda < 0$ and/or $\lambda^* < 0$.

$\Rightarrow$ recovered R, $x' = \lambda' x$

With known R, x', relative depth can be recovered as:

$\lambda m_j = \lambda^* R m_j^* + x'$,

$\lambda^* [m_j]_\times R m_j^* + [m_j]_\times x' = 0$.

$$\begin{bmatrix} [m_j]_\times R m_j^* & [m_j]_\times x' \end{bmatrix} \begin{bmatrix} \lambda_j^* \\ 1 \end{bmatrix} = 0.$$

$\boxed{Me = 0}$

$\Downarrow$

$\boxed{e}$

$\Downarrow$ wrap

$\boxed{E}$

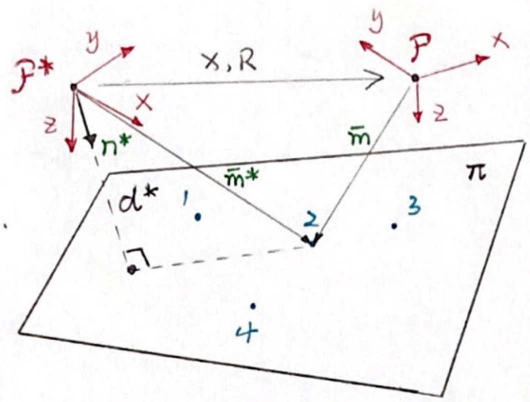
$\downarrow$ correction

$\boxed{\text{4 solutions}}$

$\downarrow$ $\lambda, \lambda^*, \frac{\lambda}{\lambda^*}$ positive

$\boxed{\text{find solution}}$

4-point algorithm



Homography Matrix:

mapping between points in two images.

至少4个点且不共线

$m_j^* = [\frac{x_j^*}{z_j^*}, \frac{y_j^*}{z_j^*}, 1]^T$

$m_j = [\frac{x_j(t)}{z_j(t)}, \frac{y_j(t)}{z_j(t)}, 1]^T$

relationship: $\bar{m}_j(t) = R(t)\,\bar{m}_j^* + X(t)$

$z_j m_j = z_j^* R\, m_j^* + X$

$$H_n = \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & 1 \end{bmatrix},$$

$h = [h_1, h_2, h_3, h_4, h_5, h_6, h_7, h_8]^T$

$$M_j = \begin{bmatrix} m_x^*, & m_y^*, & 1, & 0, & 0, & 0, & -m_x^* m_x, & -m_y^* m_y \\ 0, & 0, & 0, & m_x^*, & m_y^*, & 1, & -m_x^* m_y, & -m_y^* m_y \end{bmatrix}$$

$M_j h = m_j \Rightarrow Mh = m$

$d^* = n^{*T} \bar{m}_j^*$ (distance)

$$m_j = \frac{\bar{m}_j}{z_j} = \frac{R\bar{m}_j^* + X}{z_j}$$

$$= \frac{R z_j^* m_j^* + X\frac{d^*}{d^*}}{z_j}$$

$$= \frac{R z_j^* m_j^* + \frac{X}{d^*} n^{*T} \bar{m}_j^*}{z_j}$$

$$= \frac{R z_j^* m_j^* + \frac{X}{d^*} z_j^* n^{*T} m_j^*}{z_j}$$

$$= \frac{z_j^*}{z_j}\left(R + \frac{X}{d^*} n^{*T}\right) m_j^*$$

$$= \alpha_j \cdot H \cdot m_j^*$$

- Use standard linear algebra to find vector h that satisifties $Mh = m$.
- Wrap h back into 3×3 matrix to get estimation of H_n.
- SVD: $H_n = U \Sigma V^T$, which maps H_n to an orthogonal space such that $\Sigma = R' + x' n'^T$.
- Recover: $R = U R' V^T$, $\frac{X}{d^*} = U x'$, $n^* = V n'$.