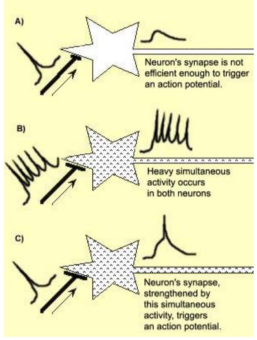
Early History



1943 McCulloch & Pitts Neuron

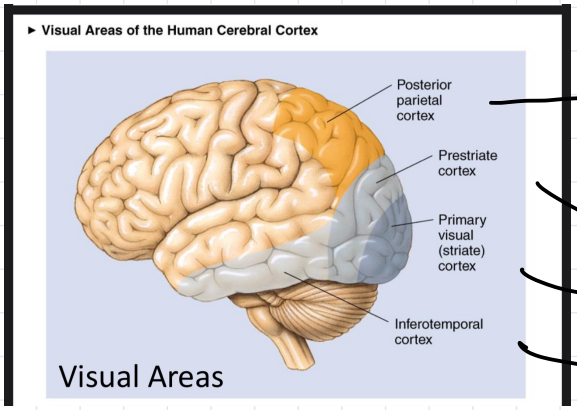
1949 D. Hebb: Synaptic Learning

1950 Turing's suggestion: produce a programme to simulate child

~1960 McCarthy, Minsky, Newell: Represent & reason in first-order logic

Rosenblatt, Widrow: Pattern Recognition, learning

Bellman, Kalman: Estimation & Control (Markov Decision ...)



→ Spatial awareness; Movement planning; Sensory info Integration

② V2 → detect complex visual features (edge, contour, texture)

① V1 → extract fundamental visual features for subsequent steps

→ visual object recognition

1962 Hubel & Wiesel: orientation sensitive neurons in V1

1980 Fukushima: Neural Network model for pattern recognition
∧ Lack the effectiveness of backpropagation ⇒ x scale output abstract repres...
(unsupervised) i.e. learned feature activations

1989 Yan LeCun Backpropgation to train weights for handwritten digits
& with GPUs, ImageNet

2012 Krizhevsky, Sutskever & Hinton ++ CV obj detection benchmark

3R : Reorganization, Recognition, Reconstruction

6 lessons from baby for embodied intelligence.

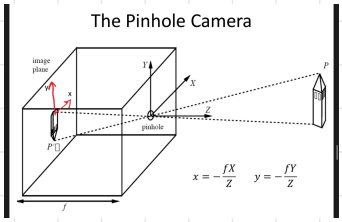
- multi-modal
- be incremental
- be physical
- Explore
- Be social
- language

Image Formation

Image $I(x,y)$ measures how much light is captured at pixel (x,y)

① where does a point (X,Y,Z) in the world get imaged

② what's the brightness at the resulting point (x,y)



Projection model

projective transformation

① perspective projection: mapping points from [depth variation] 3D space to rays through proj center

- Parallel lines converge to a vanishing point

$$\begin{bmatrix}x\\y\\z\end{bmatrix} = \begin{bmatrix}Ax\\Ay\\Az\end{bmatrix} + \lambda\begin{bmatrix}Dx\\Dy\\Dz\end{bmatrix} \quad \lambda\to\infty \Rightarrow x = \frac{fX}{Z} = \frac{Ax+\lambda Dx}{\lambda Dz} = \frac{Dx}{Dz}$$

Lines perpendicular to the camera optic axis will not converge

parallel to the image plane

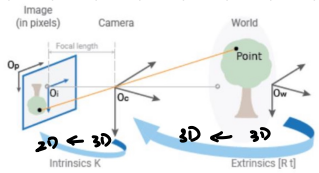
- Farther, smaller. $1/z$
- Tilted with respect to line of sight, smaller. cos Foreshortened

② Orthogonal projection: depth vary not much affine transformation

Camera Calibration

Pinhole camera to real cameras imaging 3D points

↓ silicon-based sensor convert light into electrical voltage {CCD ↑power sci / CMOS ↑noise life



[Rt] Extrinsic param: location of the camera in the 3-D scene

K Intrinsic param: optical center and the focal length of the camera

$$w\begin{bmatrix}x\\y\\1\end{bmatrix} = P\begin{bmatrix}X\\Y\\Z\\1\end{bmatrix} \qquad P = K[R\,t]$$

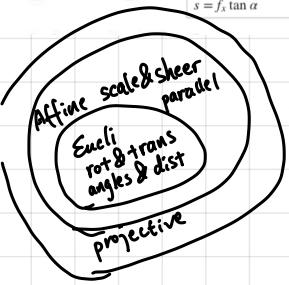
Scale factor · Image points · World points · Camera matrix · Intrinsics matrix · Extrinsics · Rotation and Translation

$$\begin{bmatrix}f_x & s & c_x\\0 & f_y & c_y\\0 & 0 & 1\end{bmatrix}$$

$[c_x \;\; c_y]$ – Optical center (the principal point), in pixels.

$(f_x \;\; f_y)$ – Focal length in pixels.
$f_x = F/p_x$
$f_y = F/p_y$
F – Focal length in world units, typically expressed in millimeters.
$(p_x \;\; p_y)$ – Size of the pixel in world units.

s – Skew coefficient, which is non-zero if the image axes are not perpendicular.
$s = f_x \tan\alpha$



Eucli: $\varphi(a) = Aa + t$, $A^TA = I$

Affine: $\varphi(a) = Aa + t$, $\det(A) \neq 0$

Homo:

Projective line: $\begin{bmatrix}x\\1\end{bmatrix}, \begin{bmatrix}2x\\2\end{bmatrix}, \begin{bmatrix}b\cdot x\\b\cdot 2\end{bmatrix}$ .. finite. $\begin{bmatrix}x\\0\end{bmatrix}$ infinite

Projective plane: $\begin{bmatrix}\lambda x\\\lambda y\\\lambda\end{bmatrix}$ $\begin{bmatrix}x\\y\\0\end{bmatrix}$

$a_1x_1 + a_2x_2 + a_3x_3 = 0$ $\quad \begin{bmatrix}x\\y\end{bmatrix} \leftrightarrow \begin{bmatrix}x_1\\x_2\\x_3\end{bmatrix}$

point on line: $\begin{bmatrix}a_1\\a_2\\a_3\end{bmatrix}\begin{bmatrix}x_1\\x_2\\x_3\end{bmatrix} = 0$

intersect: $\begin{bmatrix}a_1\\a_2\\a_3\end{bmatrix} \times \begin{bmatrix}b_1\\b_2\\b_3\end{bmatrix} = \begin{bmatrix}x\\x\\0\end{bmatrix} \rightarrow$ parallel

Affine $\begin{bmatrix}x'\\y'\\w'\end{bmatrix} = \begin{bmatrix}a_{11} & a_{12} & t_x\\a_{21} & a_{22} & t_y\\0 & 0 & 1\end{bmatrix}\begin{bmatrix}x\\y\\1\end{bmatrix}$

Perspective proj $\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1/f&0\end{bmatrix}\begin{bmatrix}x\\y\\z\\1\end{bmatrix} = \begin{bmatrix}x\\y\\z/f\end{bmatrix}$

Calibration method.

① Tsai's method: $n \geq 6$ control points 3D position + 2D coordinates in image

$[X_i\ Y_i\ Z_i]$ $i\geq 6$ and $[u_i\ v_i]$ $i\geq 6$ → Direct Linear Transform → $P = \begin{bmatrix}P_{11} & P_{12} & P_{13} & P_{14}\\P_{21} & \cdots & \cdots & \cdots\\P_{31} & \cdots & \cdots & \cdots\end{bmatrix}_{3\times 4}$

$\Rightarrow \lambda\begin{bmatrix}u\\v\\1\end{bmatrix} = P\begin{bmatrix}X\\Y\\Z\\1\end{bmatrix}$

Q $\quad QP = 0$ $\quad P$

$U,S,V = \text{svd}(Q)$

$P = V(:, 12)$

$K, RT = qr(P)$

Metric: reprojection error $= \|p^i - \pi(p_w^i, K, R, T)\|$

Refine by minimizing K, R, T, len distor $= \text{argmin}_{K,R,T,\text{len}} \|p^i - \pi(p_w^i, K, R, T)\|^2$

Levenberg-Marquardt

② Zhang's method: multi-views of a planar grid. $Z=0$ reduce complexity

2D-2D Homography

$$\lambda\begin{bmatrix}u\\v\\1\end{bmatrix} = \begin{bmatrix}h_{11} & h_{12} & h_{13}\\h_{21} & h_{22} & h_{23}\\h_{31} & h_{32} & h_{33}\end{bmatrix}\begin{bmatrix}X\\Y\\1\end{bmatrix} \qquad QH = 0 \quad \geq 4 \text{ non-colinear}$$

20-50 views for KRT $\quad H = \text{SVD}$

After calibration, camera localization PnP

determine camera pose (R,T) relative to World Frame.

3 (P3P) +1 for disambiguation.

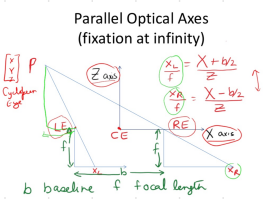
Calibrated: DLT or PnP $\quad$ Uncalibrated: DLT

$[X_w^i\ Y_w^i\ Z_w^i]$ $i\geq 4$ for non-collinear → DLT/PnP → R, T

$[u^i\ v^i]$ → DLT ($i\geq 6$ for non-coplanar) → K, R, T

R, T and K, R, T → non-linear optimization → ↓ reprojection error.

Triangulation



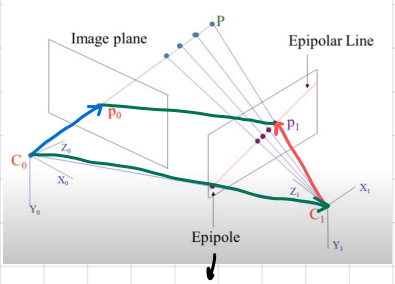
$\text{disparity} = x_L - x_R = \frac{bf}{z}$

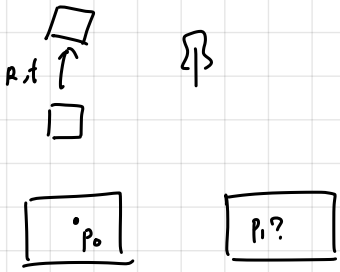
$z = \frac{bf}{x_L - x_R}$

Multi-view geometry ⇒ reconstruct internet images

{ X camera X 3D shape
✓ N correspondences ⇒ solve for camera & depth of pts

① re { camera points → epipolar geo structure from motions





estimate the motion

① in the image, forward & backward

② outside the image, sideways

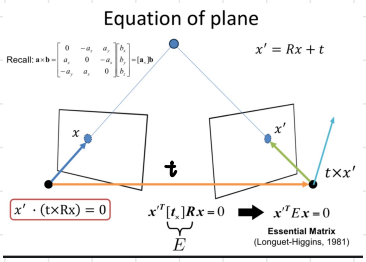
③ image center, pure translation, no rot

$\overrightarrow{C_0p_0}$ $\overrightarrow{C_0C_1}$ $\overrightarrow{C_1p_1}$ are coplanar vectors

Epipolar geo: { two camera centers / two image projections of any 3D point } ⇒ on the same plane.

⇒ reduce search space

② get camera from points using epipolar geo



$\dot{x} \cdot (t \times \dot{x}) = 0$

$\dot{x} \cdot (t \times (Rx + t)) = 0$

$\dot{x} \cdot (t \times Rx + t \times t) = 0$

$\dot{x}(t \times Rx) = 0$

$\dot{x}^T \underbrace{\hat{T} R}_{E} x = 0$

skew-symmetric orthogonal

$$\begin{bmatrix} x' & y' & z' \end{bmatrix} \underbrace{\begin{bmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & t_x & 0 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = 0$$

$$\begin{bmatrix} x' & y' & z' \end{bmatrix} \begin{bmatrix} e_{11} & e_{12} & e_{13} \\ e_{21} & e_{22} & e_{23} \\ e_{31} & e_{32} & e_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = 0$$

SVD

$E \rightarrow T, R$ ✓