Model fitting - I

Tasks: 1) Grouping 2) fitting

Hough transform
detect lines by casting vote.
image $\rightarrow$ Parameter space
$y = ax + b$ $\quad b = y - ax$
multiple points on line in img corresponds to single pt in param sp.
given (x, y) scan a & compute b.

line parameters; each pt in the image defines a line in param. space

Issues: how to represent vertical lines & range of a.
therefore Implicit line eqn
polar eqn.
$x\cos\theta + y\sin\theta - d = 0$ — ①
$Ax + By + C = 0$
for each θ_i vote $d_i = x\cos\theta_i + y\sin\theta_i$

Practical issues.
1) Parameter space size
2) Bin size — BST more efficient but provided less localisation.
3) Peak detection.

Hough with edge elements.
$d = x\cos(\theta + \alpha\Delta\theta) + y\sin(\theta + \alpha\Delta\theta)$
where $\alpha \in [-1, 1]$

Hough for circle.
Params a, b, r
$(x-a)^2 + (y-b)^2 = r^2$
or $x = a + r\cos\theta$, $y = b + r\sin\theta$
Given (x, y) vote in each r plane using: [where $\theta \in (0, 360)$]
$a = x - r\cos\theta$, $b = y - r\sin\theta$

Circle Hough Vote.


each point (x, y) casts a circle vote in each r plane.

Correlation Matrix of points
$$S = \sum P_i P_i^T = \begin{bmatrix} \sum x_i^2 & \sum x_i y_i & \sum x_i \\ \sum x_i y_i & \sum y_i^2 & \sum y_i \\ \sum x_i & \sum y_i & n \end{bmatrix} = D^T D \quad D = \begin{bmatrix} x & y & 1 \\ \vdots & \vdots & \vdots \\ x_m & y_m & 1 \end{bmatrix}$$
(3×1, 1×3, 3×3)

Review ① $(n_x, n_y) = (1, 2)$, $d = 2$
$ax + by - d = 0$
$\frac{x}{\sqrt5} + \frac{2y}{\sqrt5} - \frac{2}{\sqrt5} = 0$

② Given pts: $(x_1, y_1) = (0, 1)$, $(x_2, y_2) = (1, 3)$, $(x_3, y_3) = (2, 6)$
$$D = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 3 & 1 \\ 2 & 6 & 1 \end{bmatrix} \Rightarrow S = \begin{bmatrix} 5 & 15 & 3 \\ 15 & 46 & 10 \\ 3 & 10 & 3 \end{bmatrix}$$
find S.

③ Given gradient vectors $(1, 2), (3, 4)$
$A = \begin{bmatrix} 1+3^2 & \\ & \end{bmatrix}$

Line fitting
★ Given a grp of points (eg. neighbouring a line detected using Hough) find a more accurate model:
$$E(a, b) = \sum_{i=1}^{n} (y_i - (ax_i + b))^2$$
$$a^*, b^* = \underset{a, b}{\operatorname{argmin}}\ E(a, b)$$
$$\Rightarrow \nabla E(a, b) = 0 \Rightarrow a^*, b^*$$
$$\nabla E(a, b) = \begin{bmatrix} \frac{dE}{da} \\ \frac{dE}{db} \end{bmatrix} = \begin{bmatrix} \sum 2(y_i - ax_i - b)(x_i) \\ \sum 2(y_i - ax_i - b)(-1) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$
$$\underbrace{\begin{bmatrix} \sum x_i^2 & \sum x_i \\ \sum x_i & \sum 1 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} a \\ b \end{bmatrix}}_{x} = \underbrace{\begin{bmatrix} \sum y_i x_i \\ \sum y_i \end{bmatrix}}_{b}$$
$$\boxed{x = A^{-1} b}$$
$$A = \begin{bmatrix} \sum x_i^2 & \sum x_i \\ \sum x_i & m \end{bmatrix}$$
$$x = \begin{bmatrix} a \\ b \end{bmatrix} \quad b = \begin{bmatrix} \sum x_i y_i \\ \sum x_i \end{bmatrix}$$

Poor model — lust optimization won't work.

minimizing geometric distance
$l^T P = 0$ — point P is on the line if $(a, b, c)^T$, $(x, y, 1)$
line in homog. / geometric coordinates

$ax + by + c = 0$ — normal coord., inver. dis from origin.

Geometric Line fitting
$$E(l) = \sum_i (l^T P_i)^2 \;★$$
$P_i = (x_i, y_i, 1)$
$$E(l) = l^T \underbrace{\sum P_i P_i^T}_{S} l$$
$$\boxed{E(l) = l^T S l}$$
$$\nabla E(l) = 0 \Rightarrow \boxed{Sl = 0}$$
l is eigen vector of S belonging to 0 eigen values

Ellipse fitting
explicit eqn
$$\left(\frac{x - x_0}{a}\right)^2 + \left(\frac{y - y_0}{b}\right)^2 = 1$$
★ Implicit eqn
$$ax^2 + bxy + cy^2 + dx + ey + f = 0$$
$$b^2 - 4ac < 0$$
or $l^T P = 0$
(a, b, c, d, e, f) $(x^2, xy, \ldots, 1)$

Ellipse fitting
find param. l by min. alg. dis
$$E(l) = l^T S l$$
$$S = \sum P_i P_i^T$$
$$l^* = \underset{l}{\operatorname{argmin}}\ E(l)$$
s.t. $b^2 - 4ac < 0$

★ Rewriting $b^2 - 4ac \Leftrightarrow l^T C l = -1$
$$C = \begin{bmatrix} 0 & 0 & -2 & 0 & 0 & - \\ 0 & 1 & - & - & - & - \\ -2 & 0 & - & - & - & - \\ 0 & - & - & - & - & - \\ 0 & - & - & - & - & - \\ 0 & - & - & - & - & - \end{bmatrix}_{6\times6}$$

Modified objective
$$E(l) = l^T S l + \lambda(l^T C l + 1)$$
$$l^* = \operatorname{argmin} E(l)$$
$$\nabla E(l) = 0$$
$$l^* = \lambda S^{-1} C l$$
l^* is the eigenvector of $S^{-1}C$ belong. to $-ve$ eigen values.

Algebraic distance
x_i on ellipse $l^T x_i = 0$
x_i off ellipse $q_i = l^T x_i$
$$q_i \approx \frac{d_i}{d_i + r_i}$$
$d_1 \approx d_2$ but $q_1 > q_2$
because $(r_2 > r_1)$
so pts close to short axis will affect more to fitting.

Geometric distance
↳ Distance of a pt to arbitary implicit curve $f(x) = 0$
$d(P, f) = |P - x^*|$
x^* is a closest pt.
★ soln $d(P, f) = |P - x^*| = \frac{|f(P)|}{|\nabla f(x^*)|}$ → algeb. dis.

Review ② $x\cos\theta + y\sin\theta - d = 0$
$y = -\frac{\cos\theta}{\sin\theta} x + \frac{d}{\sin\theta}$
to avoid 0 in denominator.

① $l_1 = x - y - 10\sqrt2 = 0$
$l_2 = x - y + 10\sqrt2 = 0$

③ with $y = ax + b$ we cannot model vertical or near vertical lines
$a = \frac{dy}{dx}$
$b = y - ax$
$a \rightarrow$ slope

④ Given gr. vector $(1, 2)$ & $(3, 4)$
$$\text{Corr. matrix} = \begin{bmatrix} \sum x_i^2 & \sum x_i y_i \\ \sum x_i y_i & \sum y_i^2 \end{bmatrix} = \begin{bmatrix} 10 & 14 \\ 14 & 20 \end{bmatrix}$$

⑤ Given pts $(10, 10)$ & $(20, 20)$ write implicit line eqn. (normal non)
$$A = \begin{bmatrix} \sum x^2 & \sum x_i \\ \sum x_i & n \end{bmatrix} = \begin{bmatrix} 500 & 30 \\ 30 & 2 \end{bmatrix} \quad n = 2$$
$x = A^{-1} b$
$$\operatorname{Adj}(A) = \begin{bmatrix} 2 & -30 \\ -30 & 500 \end{bmatrix}$$
$\det A = 1000 - 900 = 100$
$A^{-1} = \frac{\operatorname{Adj} A}{\operatorname{Det} A}$
$$A^{-1} = \begin{bmatrix} 1/50 & -3/10 \\ -3/10 & 5 \end{bmatrix}$$
$x = [\ ][\ ]$
$$b = \begin{bmatrix} \sum x_i y_i \\ \sum y_i \end{bmatrix} = \begin{bmatrix} 500 \\ 30 \end{bmatrix} \quad x = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$
hence $a = 1$, $b = 0$
$\therefore y = ax + b$
$y = x$ or $(x - y = 0)$ $(-x + y = 0)$
Normalized normal vector: $[-1/\sqrt2, 1/\sqrt2]^T$ or $[1/\sqrt2, -1/\sqrt2]^T$

⑥ line coeffic. $(1, 2, 3)$ find y intercept
$$[1\ 2\ 3]\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = 0$$
$x + 2y + 3 = 0 \Leftrightarrow y = -2.5$

⑦ point $(1, 1)$ find vote in space for $\theta = 0$.
$d = (1)\cos 0 + y\sin(0)$
$d = 1$ (vote in PS)

⑧ Points $(1, 2)$ & $(3, 4)$
$$m = \begin{bmatrix} \sum x_i^2 & \sum x_i y_i & \sum x_i \\ & \sum y_i^2 & \\ & & n \end{bmatrix}$$

⑨ implicit curve f & point P
d of P from curve? $f(P) = 1$, $|\nabla f(P)|$
Geometric dis.
$d(P, f) = \frac{|f(P)|}{|\nabla f(P)|} = 1/2$ (if not mentioned take algeb.)
Alg. dis $= |f(P)| = 1$.

⑩ Discrete contour 3 pts:
$P_1 = (1, 2)$, $P_2 = (2, 3)$, $P_3 = (3, 4)$
$E_{conti} = |P_{i+1} - P_i|^2 = |P_2 - P_1|^2 = (\sqrt{1^2 + 1^2})^2 = 2$
$E_{curv} = |P_{i+1} - 2P_i + P_{i-1}|^2$
$E_{curv} = |(P_3 - P_2) - (P_2 - P_1)|^2 = 0.$

⑪ For tight fitting $\beta = 0$.

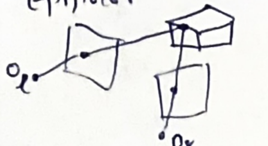
⑫ Given implic. curve f & point P, Refer Eqn that need to solve for fitting ellipse using alg. dis.
$$E(l) = \sum_{i=1}^{n} (l^T P_i)^2 + \lambda(\ )$$
$\nabla E(l) = \lambda S^{-1} C l$.
alg. dis. refer

⑬ Objective function for active contours
$$E(\psi(s)) = \int (\alpha E_{conti} + \beta E_{curv} + \gamma E_{img})\, ds,$$
where; $E_{cont} := \left|\frac{\partial \psi}{\partial s}\right|^2$, $E_{curv} = \left|\frac{\partial^2 \psi}{\partial s^2}\right|^2$, $E_{img} = -\sum |\nabla I(P_i)|^2$

⑭ In high curve, we set $\beta = 0$ to acco. sharp corners if $|P_{i+1} - 2P_i - P_{i-1}| > 2$ → $\beta_i = 0$

Multiple view geometry

epipolar



Use projection of same 3D pt in multiple views to recover structure.

Problems → Correspondence
↳ Reconstruction

EM (IV) $N \leftarrow P_l \rightarrow T$

To specify on epipolar plane we need vector normal to it.

$N = T \times P_l$ | $P_r \cdot N = 0$

★ $\boxed{P_r^T E P_l = 0}$

$\boxed{E = R^T (T)_x}$ → Essential matrix rank-2

Using E we can find l — where on left

P_l | P_r, $l = E P_l$ right | $l = E^T P_r$

Correspon.
Approach: 1) sparse: feature based, handle large disparities
2) dense: correlation, SSD, produce more pts.

Cases when corr. not possible:
↳ obj. pts not visible in both views (occlusion)
↳ ambiguous, ↳ uniform regions

Summary ★
In Camera Coor.
$P_r^T E P_l = 0$ ← essential matrix

In image coordinates:
$P_r^T F P_l = 0$ ← Fundamental matrix

Dense Corresp. W_1 W_2

Correlation $V(w_1, w_2) = \sum_i W_1(x_i, y_i) \cdot W_2(x_2, y_2)$

SSD $= -\sum_i |W_1(x, y) - W_2(x, y)|^2$

High value = good correspondence

with normalization (reduce dependency)

ZNCC $= \sum_i \dfrac{(W_1(x_1, y_1) - \mu_{w_1})(W_2(x_2, y_2) - \mu_{w_2})}{\sigma_{w_1} \sigma_{w_2}}$ zero mean normal.

ZNSSD $= \sum \left(\dfrac{(-\mu)}{\sigma_{w_1}} - \dfrac{(-\mu)}{\sigma_{w_2}} \right)^2$

★ **8 point algo. Summary**

Given: $[P_i]_{i=1}^{m} \Leftrightarrow [P_i']_{i=1}^{m}$

① normalise pts. $q_i = M_p P_i$, $q_i' = M_p' P_i'$

② Use 8 pt algo → F'

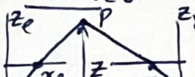
③ convert F' to F

$\boxed{F = M_p'^T F' M_p}$

Axis-aligned stereo

$Z = f \dfrac{T}{d}$

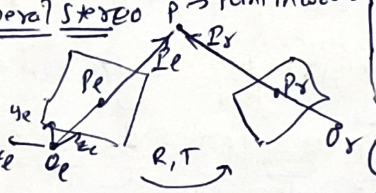
if $d \downarrow$ $Z \uparrow$



$(d = x_r - x_l = \text{disparity})$

General stereo

P → point in world



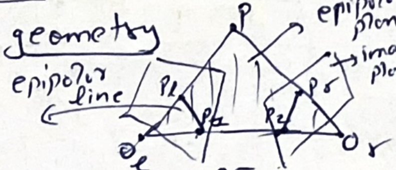
$M_{l \leftarrow r} = M_{l \leftarrow w} * M_{w \leftarrow R}$
$= R_l^T T_l^{-1} (R_r^T T_r^{-1})^{-1}$
$= \boxed{R_l^T T_l^{-1} \cancel{T_r} R_r}$

For R, T w.r.t to world

$\boxed{R = R_l^T R_r}$, $\boxed{T = R_l^T (T_r - T_l)}$

Epipolar geometry

Point P_l, P_r epipoles



epipolar line:
↳ (every e_l must pass through P_1, P_2) ↳ P_1, P_2 may be inside or outside the img ↳ (P_1, P_2 may be ∞ when img planes become parallel)

epipolar constraint
e_l (E) Essential Matrix

$P_r = R^T (P_l - T)$

★ $\boxed{P_l = R P_r + T}$

EM (II)
Moving from camera to img. coor.

$P_l = \dfrac{f_l}{Z_l} P_l$

$P_r = \dfrac{f_r}{} P_r$

Cross product

$A \times B = [A]_x B$

$= \begin{bmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{bmatrix}$ ✓

skew sym.

RANSAC

↳ fundamental problem with outliers is that model fitting can be mismatched by the influence of outliers.

Function used for Robust estimation:

$E(\theta) = \sum_{i=1}^{n} \rho_\sigma\, d(x_i, \theta)$ in re. the function gives low weight for high value outlier, however in least square function give high weight to high value outlier. Here ρ_σ will lower the influence of outlier

$\rho_\sigma(x) = \dfrac{x^2}{x^2 + \sigma^2}$ Geman Mc. fn.

↳ if σ is bigger it will include more pts.
if σ is small it will include few pts.

↳ we can control the loss function

↳ weight of outlier is up to 1 (up. bond)

↳ ★ start with large σ & decrease as converging:

$\sigma^{(n)} = 1.5 \text{ median}[d(x_i, \theta^{n-1})]$

Note:
For $x = 1, \sigma = 1$ $\rho_\sigma = 0.5$

Principal of RANSAC: It is a learning technique to estimate parameters of a model by random sampling of observed data. Given a dataset whose data elements contain both inliers & outliers RANSAC uses the voting scheme to find to optimum fitting result.

In RANSAC, the no. of points drawn at each attempt should be small. because smaller no. of points drawn can avoid including more outliers.

Parameters of RANSAC algo: • n: is the no. of points at each evaluation
• d: is the min. no. of pts needed • t: threshold to identify inliers.
• k: is no. of trials. Formula for no. of trials $\boxed{k} = \log(1-P)/\log(1-w^n)$

Given: desired $P = 0.99$ of at least 1 exp. not having outliers, & a prob. $w = 0.9$ that point is an inlier. No. of expon needed?

$K = \log(0.01)/\log(1 - 0.9^n)$

n	k	n	k
1		6	
2		7	
3		8	
4		9	
5		10	

Epipolar Review Q's

① Note for NCC & SSD
If allowing search space to entire image, will get more errors / solution & epipole line.

② For axis aligned stereo.
$(x_l, y_l) = (100, 200)$
$(x_r, y_r) = (103, 200)$
$\cdot\ Z = f \dfrac{T}{d}$ $f = 10$
base $= T = 100$
$d = x_r - x_l$
$= \dfrac{1000}{3}$ $= 3$

③ **Ambiguity**
↳ Point pair may not match correctly & incorrect matching will lead to incorrect depth.

④ A & B vectors
$(1,2,3)$ & $(2,3,4)$ matrix × B = A × B ?

$\begin{bmatrix} 0 & -3 & 2 \\ 3 & 0 & -1 \\ -2 & 1 & 0 \end{bmatrix}$

⑤ $P_r^T F P_l = [2\ 3\ 1] \overset{F}{\begin{bmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \\ 3 & 4 & 5 \end{bmatrix}} \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} = [11\ 17\ 23] \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix}$

(Rows to be formed) $= 68$.

⑥ Matrix to solve

given left pt $P_l = (x_l = 1, y_l = 2)$ P_r $(x_r = 2, y_r = 3)$

$\boxed{P_r^T F P_l} = [x_r x_l\ \ x_r y_l\ \ x_r\ \ y_r x_l\ \ y_r y_l\ \ y_r\ \ x_l\ \ y_l\ \ 1]$

$= [2\ 4\ 2\ 3\ 6\ 3\ 1\ 2\ 1]$

Segmentation

Semantic segmen. Not classify type whereas Instance segm does.

Object recognition invariant to:
↳ pose, illumination, deformation, occlussion, background, natural variability (eg cars) different kinds of car.

1) extract features (eg sift, Hog) & train a classifier.
2) Bag of words: extract patches & cluster them 'code book'.
3) Eigenfaces: less sensitive to variations, map images to lower dimensional space. 4) CNN: learn feature extraction and classification / class generalization / network learn features / supervised

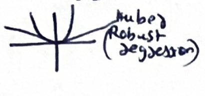
Training & Inference: Train data → $\boxed{\text{extract features}}$ $\xrightarrow{x}$ $\boxed{\text{train algo.}}$ → $\hat{\theta}$ — Large models may overfit & require more data.

Inference: data → extract features $\xrightarrow{x}$ $\boxed{\text{classify}}$ → $\hat{y}$

Activation function: (non-linear)

① Step: $h(a) = \begin{cases} 1 & \text{if } a > 0 \\ 0 & \text{otherwise} \end{cases}$, $\theta^T x + \theta_0$ | ② Sigmoid (binary) $h(a) = \frac{1}{1+\exp(-a)}$ | ③ tanh $h(a) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$ (1, −1)

④ ReLu $= \max(0, a)$; $\max(\alpha a, a)$ – Leaky ReLu | ⑤ ELU $= \begin{cases} a & a \geq 0 \\ \alpha(e^a - 1) & a < 0 \end{cases}$ | ⑥ Softmax (for k class) $= \frac{\exp(s_j^{(i)})}{\sum_{l=1}^{k} \exp s_l^{i}}$



Loss function: $L_1 = \sum_{j=1}^{k} |\hat{y}_j - y_j|$ | $L_2 = \sum (\hat{y}_j - y_j)^2$ ✓
(In regression) (min known & predicted) Huber loss → $\rho_\sigma(d) = \begin{cases} \frac{1}{2}d^2 & \text{if } d \leq \sigma \\ \sigma(d - \frac{1}{2}\sigma) & \text{otherwise} \end{cases}$ (quadratic for small d, linear for large)

Cross entropy loss → convert similarity scores to probabilities:
→ $\hat{y}_j^{(i)} = P(y=j|x^{(i)})$, $j \in [1,k]$
→ likelihood $\boxed{L(\theta) = \prod_{i=1}^{m} \prod_{j=1}^{k} (P(y=j|x^{(i)}))^{y_j^{(i)}}}$ → log likelihood (−ve LL) $= -\log L(\theta) = -\sum\sum y_j \log(\hat{y}_j)$
↳ maximize
∘ prob. of class (joint prob).

Cross entropy loss (possible values) $L_i(\theta) \in [0, \infty]$: 0 when $P \to 1$, ∞ when $P \to 0$

Optimization: to minimize loss solve:

$\theta^* = \arg\min_\theta L(\theta)$, $\nabla L(\theta) = 0 \Rightarrow \theta^*$
→ If $\nabla L(\theta)$ is not linear it is hard to find soln. ⇒ numerical iterative soln (GD)

GD ① start with guess θ_0
② $\theta^{(i+1)} \leftarrow \theta^i - \eta \nabla L(\theta^i)$
③ Stop $|L\theta^{(i+1)} - L\theta^{(i)}| < \tau$ — local minima

SGD: update after every example | (mini batch) ↳ update using a subset of examples | (GD) update based on all examples

Regularization → simple explanation are better

→ simple soln when weight θ are lower (when θ < 0 we remove our coefficient).
→ Smaller coeff: more stable soln that will generalize better.

$L(\theta) = \frac{1}{m} \sum \mathcal{L}(f(\,), y) + \lambda R(\theta)$ ↑ Regularization term

$\boxed{R\theta = \sum\sum \theta_{ij}^2}$ (L2 regularization) → λ weight of regularization (hyperparameter)

$\boxed{R(\theta) = \sum\sum |\theta_{ij}|}$ (L1 regularization)

→ L2 regularization minimizes weight while spreading them. ✓ eg) $0.5^2 + 0.5^2 < 1^2 + 0^2$
→ L1 does not have that property & may concatenate weight (eg. $|0.5| + |0.5| = |1| + |0|$)

Learning rate decay

↳ make learning smaller as iterations progress.
∘ strategies 1) step decay $\eta \to \eta/2$
2) exponential decay $\eta = \eta_0 e^{-k/t}$

Adaptive Optimization Methods

① Newton's method
→ compute learning rate (instead of specifying it / Hessian matrix)
② AdaGrad (replace matrix with approx.)
③ RMS Prop (add decay to AdaGrad)
④ Adam (add gradient momentum to RMSprop)

Feature Map Shape:

$fm = \left[\frac{i - k + 2P}{s}\right] + 1$ (P → padding)

eg) $fm = \left[\frac{125 - 3 + 0}{1}\right] + 1 = 126$

for 16 filters | resultant tensor 126×126×16.

with stride = 2: $125/2 \to 63$

Zero Padding on 4×4

$$\begin{bmatrix} 0&0&0&0&0&0 \\ 0&2&2&2&2&0 \\ 0&2&2&2&2&0 \\ 0&2&2&2&2&0 \\ 0&2&2&2&2&0 \\ 0&0&0&0&0&0 \end{bmatrix} \times \begin{bmatrix} 1&1&1 \\ 1&1&1 \\ 1&1&1 \end{bmatrix} = \begin{bmatrix} 8&12&12&8 \\ 12&18&18&12 \\ 1&1&1&1 \end{bmatrix}$$

If RGB channel then add final output.

Dilated kernel:

CNN zero pad → to prevent shrink | Stride ↑ size ↓ or downsamp

1×1 kernel is used for Dimension Reduction (depth ↓) & in making computation less expensive. 28×28×64 → 32 conv. filters of size (1×1×64) → 28×28×32

Dilated convolution (Atrous convolution).
→ increase receptive field (part of orig image that filter sees) without ~~~~ it.

Pooling → downsample spatial dimensions (depth unchanged).
→ no learning
→ supports multiplescale analy

$\begin{bmatrix} 1&4&2&5 \\ 0&6&1&1 \\ 1&8&1&1 \\ 1&1&7&1 \end{bmatrix}$ $\xrightarrow{\text{stride}=2}$ $\begin{bmatrix} 6&5 \\ 8&7 \end{bmatrix}$ reduced by 75%.

Padding = 'valid' (no padding)
= same (yes padding)

(Steps per epoch = $\frac{\#\text{ examples}}{\text{batch-size}} = \frac{2000}{20} = 100$)

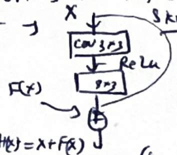
Overfitting: when model does not perform well on validation se
↳ transfer learning / dropout / data augmentation (Rotat/Trans/Shear/Flip)

Transfer learning: – Add "conv-base" (the loaded model)
– freeze weights of pretrained network – train end to end.

VGG-16 → use for feature extraction / Classification (not state of art)

Inception → multiple filters (1×1, 3×3, 5×5)

ResNet →



→ helpful in vanishing gradient.

MobileNet
→ use depthwise separable conv
↳ width multi
↳ resolution m

Detection evaluation (for Localisation)
① IoU (Intersection of Union) | Jaccard index (IoU should be high)

$IoU = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}$ | $JI = 1 - IoU$

$IoU \in [0,1]$, same, mutually exclusive

② mAP (Mean Avg. Precision) (for object detection)
↳ classification accuracy.

$\text{Recall} = \frac{TP}{TP+FN}$ ⇄ tradeoff: higher confidence < high precision, low recall

$\text{Precison} = \frac{TP}{TP+FP}$

AUC (record precision & recall at diff. levels)
$AUC \in [0,1]$ high is better.

CC. (Projection matrix)

3×4

$p_i = M P_i$

image pts 2DH; world pts 3DH

$p_i' = M p_i$ (3×3), 2DH → 2DH

$M = K^*[R^* | T^*]$ — (1)

intrinsic, extrinsic

$K^* = \begin{bmatrix} \alpha_u & s & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}$ $\alpha_u = f k_u$, $\alpha_v = f k_v$, $s = \alpha_u \tan\theta$

Non-Planar: Find → K^*, R^*, T^* → $R, T, f, k_u, k_v, u_0, v_0, \tan\theta$

Approaches → direct / indirect (calling M)

2DH, 3×4, 3DH

$p_i' = M P_i$

$\begin{bmatrix} x_i' \\ y_i' \\ w_i' \end{bmatrix} = \begin{bmatrix} -m_1^T- \\ -m_2^T- \\ -m_3^T- \end{bmatrix} P_i$

$x_i' = m_1^T P_i$, $y_i' = m_2^T P_i$, $w_i' = m_3^T P_i$

$x_i = \frac{x_i'}{w_i'}$, $y_i = \frac{y_i'}{w_i'}$

$x_i = \frac{m_1^T P_i}{m_3^T P_i}$, $y_i = \frac{m_2^T P_i}{m_3^T P_i}$

For m pts

$m_1^T P_i - x_i m_3^T P_i = 0$
$m_2^T P_i - y_i m_3^T P_i = 0$

for each point 2 eqn with 12 unknowns ↳ need at least 6 point pairs

For a single pt.

$\begin{bmatrix} P_i^T & 0 & -x_i P_i^T \\ 0 & P_i^T & -y_i P_i^T \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \\ m_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$

2×12, 12×1, 2×1

For m pts ($m \geq 6$ for better accuracy)

$2m\times12$, 12×1, $2m\times1$

Solve using SVD $Ax = 0$, $A = UDV^T$

Sol^n col^n of V belonging to zero singular value

$\hat{x} = \begin{bmatrix} \hat{m}_1 \\ \hat{m}_2 \\ \hat{m}_3 \end{bmatrix} \Rightarrow \hat{M} = \begin{bmatrix} -\hat{m}_1^T- \\ -\hat{m}_2^T- \\ -\hat{m}_3^T- \end{bmatrix}$ — (2)

sol^n not unique. if $A\hat{x} = 0$ ∴ $A(\rho\hat{x}) = 0$

from (1) & (2)

$K^*[R^* | T^*] = \rho \hat{M}$

$\begin{bmatrix} \alpha_u r_1^T + s r_2^T + u_0 r_3^T & \\ \alpha_v r_2^T + v_0 r_3^T & K^* T^* \\ r_3^T & \end{bmatrix} = \rho \begin{bmatrix} -a_1^T- & \\ -a_2^T- & b \\ -a_3^T- & \end{bmatrix}$

unkn. ; UK ; knowns

knowns a_1, a_2, a_3, b

$r_1 \cdot r_2 = 0$, $r_2 \cdot r_3 = 0$, $r_3 \cdot r_1 = 0$ | $r_1 \times r_2 = r_3$, $r_2 \times r_3 = r_1$, $r_3 \times r_1 = r_2$

Solve: $|r_3^T| = |\rho| |a_3^T|$, 1; $\rho = \frac{1}{|a_3^T|}$

Summary

$|\rho| = 1/|a_3|$

$u_0 = |\rho|^2 a_1 \cdot a_3$

$v_0 = |\rho|^2 a_2 \cdot a_3$

$\alpha_v = \sqrt{|\rho|^2 a_2 \cdot a_2 - v_0^2}$

$s = \frac{|\rho|^4}{\alpha_v} ((a_1 \times a_3) \cdot (a_2 \times a_3))$

$\alpha_u = \sqrt{|\rho|^2 a_1 \cdot a_1 - s^2 - u_0^2}$

$\boxed{K^* = \begin{bmatrix} \alpha_u & s & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}}$

$\epsilon = \text{sgn}(b_3)$

Finding params from M.

$T^* = \epsilon |\rho| (K^*)^{-1} b$

$r_3 = \epsilon |\rho| a_3$

$r_1 = \frac{|\rho|^2}{\alpha_v} a_2 \times a_3$

$r_2 = r_3 \times r_1$

$R^* = [r_1^T \; r_2^T \; r_3^T]^T$

Recovering R & T from R^* & T^*

$\boxed{R = (R^*)^T}$ (or $R^* = R^T$)

$T^* = -R^T T \Rightarrow T = -R T^*$

$\boxed{T = -(R^*)^T T^*}$

Quality $E = \frac{1}{m} \sum_i \left(\left| x_i - \frac{m_1^T P_i}{m_3^T P_i} \right|^2 + \left| y_i - \frac{m_2^T P_i}{m_3^T P_i} \right|^2 \right)$

Degenerate config.

↳ Failed if points are on plane (same plane) / (many possible solutions)
+ Quadratic surface.

eg) Given $R^*_{\text{w.r.t camera}}$, $T^*_{\text{w.r.t camera}}$; find R & T w.r.t world.

$R^* = \begin{bmatrix} 6 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

$\cdot R = (R^*)^T = \begin{bmatrix} 6 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

$\cdot T = -R T^*$
$= -(R^*)^T T^*$
$= -\begin{bmatrix} 6 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}^T_{3\times1}$
$= -[6 \; 2 \; 3]^T$

eg) Given $M = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 5 \\ 3 & 4 & 5 & 6 \end{bmatrix}$ (Calculation)

find u_0, v_0, E?

$a_1^T = [1\,2\,3]$
$a_2^T = [2\,3\,4]$
$a_3^T = [3\,4\,5]$

$u_0 = \frac{1}{|a_3|^2} a_1 \cdot a_3$

$|a_3|^2 = (\sqrt{9+16+25})^2 = 50$

$a_1 \cdot a_3 = 3 + 8 + 15 = 26$

$u_0 = 26/50$. Given: image pts

For E we have $(x_i, y_i) = (1, 2)$ & $P_1 = \begin{bmatrix} 3 \\ 4 \\ 5 \\ 1 \end{bmatrix}$ (world pts) + Matrix M

For error we need m_1^T, m_2^T, m_3^T

$m_1^T = [1\,2\,3\,4]$ Similarly m_2^T & m_3^T

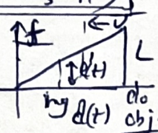
$m_1^T P = [1\,2\,3\,4] \begin{bmatrix} 3 \\ 4 \\ 5 \\ 1 \end{bmatrix} = 30$

$E = \ldots \sum (\ldots)^2 + |2 - (\ldots)|^2$

Video Sequence

$t_k = t_0 + k\Delta t$

Correspondence is easier but reconst. is not (from stereo) big error



Single: still camera / moving object; object / camera

$\boxed{z = \frac{l(t)}{l'(t)}}$ time to impact → no need to know object size, f, V.

$\boxed{z = \frac{d(t)}{V}}$

$\boxed{d(t) = d_0 - Vt}$

$l'(t) = \frac{fLV}{d(t)^2}$

$l(t) = \frac{fL}{d(t)}$

$\frac{l'(t)}{l(t)} = \frac{d(t)}{V} = z$

Dilated kernel (with rate 2)

$k = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 \end{bmatrix}_{5\times5}$

Planar Calibration $H = [h_1\, h_2\, h_3] = \begin{bmatrix} h_{11} & h_{21} & h_{31} \\ h_{12} & h_{22} & h_{32} \\ h_{13} & h_{23} & h_{33} \end{bmatrix}$

$v_{ij} = [h_{i1}h_{j1},\; h_{i1}h_{j2} + h_{i2}h_{j1},\; h_{i2}h_{j2},\; h_{i3}h_{j1} + h_{i1}h_{j3},\; h_{i3}h_{j2} + h_{i2}h_{j3},\; h_{i3}h_{j3}]^T$

$\begin{bmatrix} v_{12}^T \\ v_{11}^T - v_{22}^T \\ \vdots \end{bmatrix} \begin{bmatrix} s_{11} \\ s_{12} \\ s_{22} \\ s_{13} \\ s_{23} \\ s_{33} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \end{bmatrix}$

$c_1 = (s_{12}s_{13} - s_{11}s_{23})$
$c_2 = (s_{11}s_{22} - s_{12}^2)$
$v_0 = c_1/c_2$
$\lambda = s_{33} - (s_{13}^2 + v_0 c_1)/s_{11}$
$\alpha_u = \sqrt{\lambda/s_{11}}$, $\alpha_v = \sqrt{\lambda s_{11}/c_2}$
$s = -s_{12}\alpha_u^2 \alpha_v / \lambda$
$u_0 = s v_0/\alpha_u - s_{13}\alpha_u^2/\lambda$

$K^* = \begin{bmatrix} \alpha_u & s & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}$

$|\alpha| = 1 / |(K^*)^{-1} h_1|$

$\text{sign}(\alpha) = \text{sign}(\quad)$

$\alpha = |\alpha| \,\text{sign}(\alpha)$

$r_1 = \alpha (K^*)^{-1} h_1$

$r_2 = \alpha (K^*)^{-1} h_2$

$r_3 = r_1 \times r_2$

$T^* = \alpha (K^*)^{-1} h_3$

$p_i = M P_i$ (2DH, 3×3, 2DH) Assume $[P_i]_z = 0$

$P_i = [x_i, y_i, 0]$

$p_i = K^*[R^* | T^*] P_i$

$= K^* \begin{bmatrix} r_1 & r_2 & r_3 & T^* \end{bmatrix}_{3\times4} \begin{bmatrix} x_i \\ y_i \\ 0 \\ 1 \end{bmatrix}$

$= K^*(x_i r_1 + y_i r_2 + T^*)$

$= K^*(r_1 \; r_2 \; T^*) \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix}$ 2DH

2D projective map 3×3 M (Homography) H

Similarly,

$\begin{bmatrix} u_i \\ v_i \\ w_i \end{bmatrix} = H P_i^* = \begin{bmatrix} -H_1^T- \\ -H_2^T- \\ -H_3^T- \end{bmatrix} P_i^*$

$\begin{bmatrix} P_i^T & 0 & -x_i P^{*T} \\ 0 & P^{*T} & -y_i P^{*T} \\ \vdots \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \end{bmatrix}$

$2m\times9$, 9×1, $2m\times1$

$K^*[r_1 \; r_2 \; T^*] = \alpha[\hat{h}_1 \; \hat{h}_2 \; \hat{h}_3]$

$[r_1 \; r_2 \; T^*] = \alpha K^{*-1}[\quad]$

$r_1 = \alpha K^{*-1} \hat{h}_1$
$r_2 = \alpha K^{*-1} \hat{h}_2$
$r_3 = \alpha K^{*-1} \hat{h}_3$

$r_1 \cdot r_2 = 0 = r_2 \cdot r_1 = 1$

$\boxed{S = K^{*-T} K^{*-1}}$

$\hat{h}_1^T S \hat{h}_2 = 0$, $\hat{h}_1^T S \hat{h}_1 = \hat{h}_2^T S \hat{h}_2$

knowns $\hat{h}_1, \hat{h}_2$; 5 unknowns.

unknowns $S \to K^* \to \alpha_u, \alpha_v, u_0, v_0, s$

need at least 3 planar views to estimate K^*

Direct Method

$E(K^* R^* T^*) = \sum \| (\quad) - p_i \, p_{iy} \|^2$

(Tsai method) iterative solution / radial lens.

CC: Projection matrix

$p_i' = M\, p_i$ (3×4)

image pts 2DH, world pts 3DH

$p_i' = M p_i$ (3×3) — 2DH, 2DH

$M = K^*[R^* | T^*]$ — ①

intrinsic, extrinsic

$$K^* = \begin{bmatrix} \alpha_u & s & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

$\alpha_u = f k_u$, $\alpha_v = f k_v$, $s = \alpha_u \tan\theta$

Find → K^*, R^*, T^* → $R, T, f, k_u, k_v, u_0, v_0, t_\theta$

Non-Planar Summary

$|\rho| = 1/|a_3|$

$u_0 = |\rho|^2 a_1 \cdot a_3$

$v_0 = |\rho|^2 a_2 \cdot a_3$

$\alpha_v = \sqrt{|\rho|^2 a_2 \cdot a_2 - v_0^2}$

$s = \frac{|\rho|^4}{\alpha_v} (a_1 \times a_3) \cdot (a_2 \times a_3)$

$\alpha_u = \sqrt{|\rho|^2 a_1 \cdot a_1 - s^2 - u_0^2}$

$$\boxed{K^* = \begin{bmatrix} \alpha_u & s & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}}$$

$\epsilon = \mathrm{sgn}(b_3)$

Finding params from M:

$T^* = \epsilon |\rho| (K^*)^{-1} b$

$r_3 = \epsilon |\rho| a_3$

$r_1 = \frac{|\rho|^2}{\alpha_v} a_2 \times a_3$

$r_2 = r_3 \times r_1$

$R^* = [r_1^T\ r_2^T\ r_3^T]^T$

Recovering R & T from R^* & T^*

$\boxed{R = (R^*)^T}$ (as $R^* = R^T$)

$T^* = -R^T T \Rightarrow T = -R T^*$

$\boxed{T = -(R^*)^T T^*}$

Approaches → direct / indirect (calibrating M)

$p_i' = M P_i$ (2DH, 3×4, 3DH)

$$\begin{bmatrix} x_i' \\ y_i' \\ w_i' \end{bmatrix} = \begin{bmatrix} -m_1^T- \\ -m_2^T- \\ -m_3^T- \end{bmatrix} P_i$$

$x_i' = m_1^T P_i$, $y_i' = m_2^T P_i$, $w_i' = m_3^T P_i$

$\boxed{x_i = \frac{x_i'}{w_i'},\ y_i = \frac{y_i'}{w_i'}}$

$x_i = \frac{m_1^T P_i}{m_3^T P_i}$, $y_i = \frac{m_2^T P_i}{m_3^T P_i}$

For m pts

$m_1^T P_i - x_i m_3^T P_i = 0$

$m_2^T P_i - y_i m_3^T P_i = 0$

for each point 2 eqn with 12 unknowns

↳ need at least 6 point pairs

For a single pt:

$$\underbrace{\begin{bmatrix} P_i^T & 0 & -x_i P_i^T \\ 0 & P_i^T & -y_i P_i^T \end{bmatrix}}_{2\times12} \underbrace{\begin{bmatrix} m_1 \\ m_2 \\ m_3 \end{bmatrix}}_{12\times1} = \underbrace{\begin{bmatrix} 0 \\ 0 \end{bmatrix}}_{2\times1}$$

For m pts ($m \ge 6$ for better accuracy)

2m×12, 12×1, 2m×1

Solve using SVD $Ax = 0$, $A = UDV^T$

Sol^n col^n of V belonging to zero singular value

$\hat{x} = \begin{bmatrix} \hat{m}_1 \\ \hat{m}_2 \\ \hat{m}_3 \end{bmatrix} \Rightarrow \hat{M} = \begin{bmatrix} \hat{m}_1^T \\ \hat{m}_2^T \\ \hat{m}_3^T \end{bmatrix}$ — ②

Sol^n not unique. If $A\hat{x} = 0 \therefore A(\rho\hat{x}) = 0$

from ① & ②

$K^*[R^*|T^*] = \rho \hat{M}$

$$\underbrace{\begin{bmatrix} \alpha_u r_1^T + s r_2^T + u_0 r_3^T & \\ \alpha_v r_2^T + v_0 r_3^T & K^* T^* \\ r_3^T & \end{bmatrix}}_{\text{unkn.}} = \rho \underbrace{\begin{bmatrix} -a_1^T- & \\ -a_2^T- & b \\ -a_3^T- & \end{bmatrix}}_{\text{knowns}}$$

knowns a_1, a_2, a_3, b

$r_1 \cdot r_2 = 0$, $r_2 \cdot r_3 = 0$, $r_3 \cdot r_1 = 0$

$r_1 \times r_2 = r_3$, $r_2 \times r_3 = r_1$, $r_3 \times r_1 = r_2$

Solve: $|r_3^T| = |\rho|\,|a_3^T|$, 1, $\rho = \frac{1}{|a_3^T|}$

Quality $E = \frac{1}{m} \sum_i \left(\left| x_i - \frac{m_1^T P_i}{m_3^T P_i} \right|^2 + \left| y_i - \frac{m_2^T P_i}{m_3^T P_i} \right|^2 \right)$

Planar Calibration

$$H = [h_1\ h_2\ h_3] = \begin{bmatrix} h_{11} & h_{21} & h_{31} \\ h_{12} & h_{22} & h_{32} \\ h_{13} & h_{23} & h_{33} \end{bmatrix}$$

$v_{ij} = [h_{i1}h_{j1},\ h_{i1}h_{j2} + h_{i2}h_{j1},\ h_{i2}h_{j2},\ h_{i3}h_{j1} + h_{i1}h_{j3},\ h_{i3}h_{j2} + h_{i2}h_{j3},\ h_{i3}h_{j3}]^T$

$$\begin{bmatrix} v_{12}^T \\ v_{11}^T - v_{22}^T \\ \vdots \end{bmatrix} \begin{bmatrix} s_{11} \\ s_{12} \\ s_{22} \\ s_{13} \\ s_{23} \\ s_{33} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \end{bmatrix}$$

$c_1 = (s_{12}s_{13} - s_{11}s_{23})$

$c_2 = (s_{11}s_{22} - s_{12}^2)$

$v_0 = c_1/c_2$

$\lambda = s_{33} - (s_{13}^2 + v_0 c_1)/s_{11}$

$\alpha_u = \sqrt{\lambda/s_{11}}$, $\alpha_v = \sqrt{\lambda s_{11}/c_2}$

$s = -s_{12}\alpha_u^2 \alpha_v / \lambda$

$u_0 = s v_0/\alpha_u - s_{13}\alpha_u^2/\lambda$

$$K^* = \begin{bmatrix} \alpha_u & s & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

$|\alpha| = 1/|(K^*)^{-1} h_1|$

$\mathrm{sign}(\alpha) = \mathrm{sign}(\quad)$

$\alpha = |\alpha|\, \mathrm{sign}(\alpha)$

$r_1 = \alpha (K^*)^{-1} h_1$

$r_2 = \alpha (K^*)^{-1} h_2$

$r_3 = r_1 \times r_2$

$T^* = \alpha (K^*)^{-1} h_3$

$p_i = M P_i$ (2DH, 3×3, 2DH). Assume $[P_i]_z = 0$

$P_i = [x_i, y_i, 0]$

$p_i' = K^*[R^*|T^*] P_i$

$= K^* \begin{bmatrix} r_1 & r_2 & r_3 & T^* \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ 0 \\ 1 \end{bmatrix}$

$= K^*(x_i r_1 + y_i r_2 + T^*)$

$= K^* (r_1\ r_2\ T^*) \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix}$ 2DH

2D projective map 3×3 M (Homography) H

Similarly,

$$\begin{bmatrix} u_i \\ v_i \\ w_i \end{bmatrix} = H P_i^* = \begin{bmatrix} -H_1^T- \\ -H_2^T- \\ -H_3^T- \end{bmatrix} P_i^*$$

$$\underbrace{\begin{bmatrix} P_i^T & 0 & -x_i P_i^T \\ 0 & P_i^T & -y_i P_i^T \end{bmatrix}}_{2m\times9} \underbrace{\begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix}}_{9\times1} = \underbrace{\begin{bmatrix} 0 \\ 0 \\ \vdots \end{bmatrix}}_{2m\times1}$$

$K^*[r_1\ r_2\ T^*] = \alpha[\hat{h}_1\ \hat{h}_2\ \hat{h}_3]$

$[r_1\ r_2\ T^*] = \alpha K^{*-1}[\quad]$

$r_1 = \alpha K^{*-1} \hat{h}_1$

$r_2 = \alpha K^{*-1} \hat{h}_2$

$r_3 = \alpha K^{*-1} \hat{h}_3$

$r_1 \cdot r_2 = 0 = r_2 \cdot r_1 = 1$

$\boxed{S = K^{*-1^T} K^{*-1}}$

$\hat{h}_1^T S \hat{h}_2^T = 0$, $\hat{h}_1 S \hat{h}_1 = \hat{h}_2 S \hat{h}_2$

Knowns $\hat{h}_1, \hat{h}_2$ — 5 unknowns.

unknowns $S \to K^* \to \alpha_u, \alpha_v, u_0, v_0, s$

* Need at least 3 planar views to estimate K^*

Direct Method

$E(K^* R^* T^*) = \sum \| (\quad) - p_{ix}\ p_{iy} \|^2$

(Tsai method) iterative solution / radial lens.

Degenerate config.

↳ Failed if points are on plane (same plane) (many possible solutions)

* Quadratic surface.

eg) Given $M = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 5 \\ 3 & 4 & 5 & 6 \end{bmatrix}$ (a_1^T, a_2^T, a_3^T rows)

Calculation

find u_0, v_0, E?

$a_1^T = [1\ 2\ 3]$

$a_2^T = [2\ 3\ 4]$

$a_3^T = [3\ 4\ 5]$

$u_0 = \frac{1}{|a_3|^2} a_1 \cdot a_3$

$|a_3|^2 = (\sqrt{9+16+25})^2 = 50$

$a_1 \cdot a_3 = 3 + 8 + 15 = 26$

$u_0 = 26/50$. Given: image pts

For E we have $(x_i, y_i) = (1, 2)$

& $P_i = \begin{bmatrix} 3 \\ 4 \\ 5 \\ 1 \end{bmatrix}$ world pts + matrix M

For error we need m_1^T, m_2^T, m_3^T

$m_1^T = [1\ 2\ 3\ 4]$ similarly m_2^T & m_3^T

$m_1^T P = [1\ 2\ 3\ 4] \begin{bmatrix} 3 \\ 4 \\ 5 \\ 1 \end{bmatrix} = 30$

$E = 1/\ldots \sum (|1 - (\ )|^2 + |2 - (\ )|^2)$

eg) Given R^* w.r.t camera, T^* w.r.t camera

Find R & T w.r.t world.

$$R^* = \begin{bmatrix} 6 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

• $R = (R^*)^T = \begin{bmatrix} 6 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

• $T = -RT^*$

$= -(R^*)^T T^*$

$= -\begin{bmatrix} 6 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$ (3×3, 3×1)

$= -[6\ 2\ 3]^T$

Dilated Kernel (with size 2)

$$k = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 \end{bmatrix}_{5\times5}$$

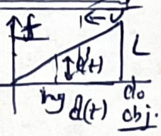
Video Sequence

$t_k = t_0 + k\Delta t$

Correspondence is easier but reconstr is not (from stereo)

→ big error

Single: still camera / moving object; object & now camera



$\boxed{Z = \frac{l(t)}{l'(t)}}$ → no need to know object size, f, V.

(Z = time to impact)

$\boxed{Z = \frac{d(t)}{V}}$

$\boxed{d(t) = d_0 - Vt}$

$l'(t) = \frac{fLV}{d(t)^2}$

$l(t) = \frac{fL}{d(t)}$

$\frac{l(t)}{l'(t)} = \frac{d(t)}{V} = Z$

Graphs

Adjacency Matrix

1. for directed graph is not symmetric

$\boxed{A \neq A^T}$

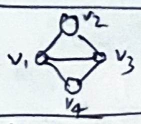
2. for undirected graph is symmetric.

$\boxed{A = A^T}$

Global Clustering Coeff & Triples (measures transitivity in undirected graphs)

$$C = \frac{(\text{No. of } \Delta\text{'s} \times 3)}{(\text{No. of Connected Triples of nodes})}$$

eg)



$$C = \frac{2 \times 3}{2 \times 3 + \underset{v_2 v_1 v_4,\ v_2 v_3 v_4}{2}} = 0.75$$

Entropy

PT 1. A null graph is an empty graph.

2. $\sum d_i = 2|E|$ (undirected graph)

3. No. of nodes with odd degree is even.

4. In directed $\sum_i d_i^{out} = \sum_j d_j^{in}$

Local Clustering Coefficient:

It measures transitivity at node level.

$$C(v_i) = \frac{\text{No. of pairs of neighbors of } v_i \text{ that are connected}}{\text{No. of pairs of neighbors of } v_i}$$

TF-IDF

$$\boxed{w_{j,i} = tf_{j,i} \times idf_j}$$

where; $tf_{j,i}$ is the frequency of word j in document i)

$$idf_j = \log_2 \frac{|D| \text{ (total no. of docs in corpus)}}{|\{document \in D \mid j \in document\}|}$$

(no. of docs where the term j appears)

tf-idf of term t, document d & document corpus D is calculated as

$$w_{ji} = tf_{j,i} * idf_j$$

Nominal → categorical

eg) dog, cat

Ordinal → Same as nominal (feature value rank/quantity/number) ($31 \le Ages \le 41$ ordinal)

eg) Small, medium, large; Humidity (High/Low)

Interval:

Add/Sub allowed

Mul/Div not allowed

eg) Temperature (F/C), Year, ~~Hum~~

Ratio ~~$31 \le Ages \le 41$~~

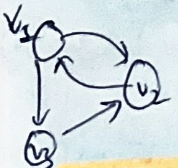
Meaningful zero pt

Div/mul allowed

eg) Height, weight, Humidity (80/9...), Age (27, 28 ...)

Reciprocity

$$R = \frac{1}{m} Tr(A^2)$$



$$A = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

Reciprocal nodes: v_1, v_2

$$R = \frac{1}{m} Tr(A^2) = \frac{1}{4} Tr\left(\left[\ \right]\right)$$

$$= 2/4 = 1/2$$

Deg

TF-IDF

Vertex Similarity $\sigma(v_i, v_j) = |N(v_i) \cap N(v_j)|$

Jaccard Simil. $= \frac{|N(v_i) \cap N(v_j)|}{|N(v_i) \cup N(v_j)|}$

Cosine Similarity $= \frac{N(v_i) \cap N(v_j)}{\sqrt{|N(v_i)|\,|N(v_j)|}}$

no. of edges   no. of edges

Katz Centrality

↳ A major problem with eigenvector centrality is when it deals with directed graphs

Degree Centrality

$C_d(v_i) = d_i = n - 1$

→ Normalize by max possible degree $= [d_i / n-1]$

→ max degree $= \frac{d_j}{\max_j d_j}$

→ by degree sum $= \frac{d_i}{2m}$

$$\boxed{C_{Katz}(v_i) = \alpha \underbrace{\sum_{j=1}^{n} A_{j,i} C_{Katz}(v_j)}_{\text{Eigenvector Centrality}} + \underbrace{\beta}_{\text{Bias}}}$$

→ when $\alpha = 0$ eigenvector centrality $= 0$ & all nodes gets same centrality β

controlling term

$$C_{Katz} = \alpha A^T C_{Katz} + \beta$$

vector of all 1's

$\det(A^T - \alpha^{-1} I) = 0$

Note: For the matrix $(I - \alpha A^T)$ to be invertible;

$\det(I - \alpha A^T) \neq 0$

~~$\det(A^T - \alpha^{-1} I)$~~ $\neq 0$

$$C_{Katz} = \beta (I - \alpha A^T)^{-1} \cdot 1$$