SAIMANOHAR VEMPI
A20514848

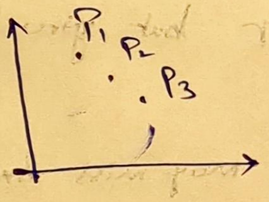
CS 512 COMPUTER VISION
ASSIGNMENT - 3

1) Line Detection

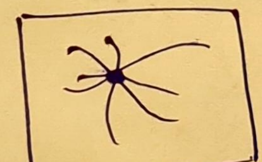
a) * The Range of m & b lies between ~~(-∞ to ∞)~~ $-\infty$ and ∞. So there is no definite range for m & b in the line equation.

* m and b in the line equation are not unique i.e different lines can have the same slope and intercept which leads to ambiguity.

b) In polar representation each point in the parameter plane looks like



Points which form the line



~~Sinuso~~

Sinusoids intersecting at one point

$$(n_x, n_y) \cdot (x, y) = d$$

$$x \cos\theta + y \sin\theta - d = 0$$

c) $(x, y) = (1, 1)$

$\theta = 1$

$d_i = x \cos\theta_i + y \sin\theta_i$

$d_i = \cancel{(1)\cos\theta}\ 1 \cos 0 + 1 \sin 0$

$d_i = 1$

d) We can detect the lines by the ~~following~~ checking the parameter plane using the below equation

$$x\cos\theta + y\sin\theta = d$$

Where,

d - distance from the ~~spa~~ origin to the line.

θ - angle

Hough transform works by iterating each pixel in the image and maps it to a Sinusoidal curve in the Hough space. Then we take the intersection of these Sinusoidal curves to detect the line.

e) * Smaller bin sizes consumes more memory but gives better localization of the lines. It was slow.

* Larger bin size requires less memory but may miss smaller ~~and~~ lines and does not provide better localization. It was efficient and faster to compute.

f) If we know the normal at each voting point in the parameter plane allows for a more robust and accurate line detection algorithm by assigning a weight to each vote that considers the alignment with the line orientation. We can improve the voting by computing the orientation instead of iterating it from 0 to 180.

g) When using the hough transform for circles, the no. of dimensions of the parameter space will be '<u>3</u>' represents (x, y, r)

where,

r - radius.

h) $\theta \in [45°, 135°]$, $x \in [0, n[$

$$y = -\frac{\cos\theta}{\sin\theta} x + \frac{d}{\sin\theta}$$

$$x\cos\theta + y\sin\theta = d$$

$$\frac{x\cos\theta}{\sin\theta} + y = \frac{d}{\sin\theta}$$

$$y = \frac{d}{\sin\theta} - \frac{x\cos\theta}{\sin\theta}$$

$$y = -\frac{x\cos\theta}{\sin\theta} + \frac{d}{\sin\theta}$$

$x = 0$, $\theta = 45°$

$$y = \frac{d}{\sin 45°} \Rightarrow \frac{d}{1/\sqrt{2}} = \sqrt{2}(d) \Rightarrow (0, \sqrt{2}d)$$

$x = n$, $\theta = 135$

$$y = -\frac{\cos 135}{\sin 135}(0) + \frac{d}{\sin 135}$$

$$= \frac{d}{1/\sqrt{2}} \Rightarrow \sqrt{2}d$$

$$= (0, \sqrt{2}d)$$

We scan x and compute y because we need to determine the y coordinate of each pixel on the line and we move horizontally from left to right.

i)

$$x\cos\theta + y\sin\theta = d$$

$$x = -\frac{\sin\theta}{\cos\theta}y + \frac{d}{\cos\theta}$$

$y \in [0,m]$ & $\theta \in [-45, 45]$

if $y=0$ & $\theta = 45$

$$x = -\frac{\sin 45}{\cos 45}(0) + \frac{d}{1/\sqrt{2}} \Rightarrow \sqrt{2}d \Rightarrow (\sqrt{2}d, 0)$$

if $y=0$, $\theta = 45$

$$x = \frac{d}{1/\sqrt{2}} \Rightarrow \sqrt{2}d \Rightarrow (\sqrt{2}d, 0)$$

Scanning y to 0 & Computing x will allow to plot the line on the image.

2. Model fitting :-

a) Disadvantage of using $y = ax+b$ for line fitting.

i) Geometric distance & real point are not minimized which will result in non accurate fitting.

ii) Lines with higher slopes cannot be fitted accurately.

b) Given

~~Hypothesis~~

$\theta = 45°$ $\qquad$ $ax+by+c=0$

$\cos\theta\, x + \sin\theta\, y - d = 0$, $\cos 45 = 1/\sqrt{2}$ $\sin 45 = 1/\sqrt{2}$

$\frac{1}{\sqrt{2}} x + \frac{1}{\sqrt{2}} y - 10 = 0$ $a=1, b=1, c=-10\sqrt{2}$

$x + y - 10\sqrt{2} = 0$ if $x=0$, $y = 10\sqrt{2}$

if $y=0$, $x = 10\sqrt{2}$

Cross check

Bottom triangle

$\cos 45 = \frac{x}{10\sqrt{2}}$

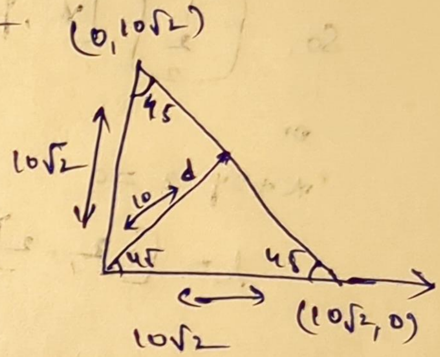
$10\sqrt{2} \times \cos 45 = x$

$\boxed{10 = x}$.

Top triangle

$\cos 45 = \frac{x}{10\sqrt{2}}$

$\boxed{x = 10}$



c) Given the points (10, 10) & (20, 20)

Implicit equation would be :-

$$A = \begin{bmatrix} \sum x^2 & \sum x_i \\ \sum x_i & m \end{bmatrix} \quad b = \begin{bmatrix} \sum x_i y_i \\ \sum y_i \end{bmatrix}$$

$X = A^{-1} b$

$$A = \begin{bmatrix} 10^2 + 20^2 & 10+20 \\ 10+20 & 2 \end{bmatrix} \Rightarrow \begin{bmatrix} 500 & 30 \\ 30 & 20 \end{bmatrix} \Rightarrow \begin{bmatrix} \frac{1}{50} & -\frac{3}{10} \\ -\frac{3}{10} & 5 \end{bmatrix}$$

$$b = \begin{bmatrix} 500 \\ 30 \end{bmatrix}$$

$$\begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} \frac{1}{50} & -\frac{3}{10} \\ -\frac{3}{10} & 5 \end{bmatrix} \begin{bmatrix} 500 \\ 30 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

eq is $y = x + 0$

$\boxed{y = x}$

d) Given

Normal $= (1, 2)$, $d = 2$

$l^T x = 0$ (Note l is a 3x1 vector)

So, $\begin{bmatrix} 1 \\ 2 \\ -2 \end{bmatrix} [x, y, 1]$

or

$x + 2y - 2 = 0$

$l = [1, 2, -2]$

e) Given line coefficient $(1, 2, 3)$ & $x = 2$

$(1, 2, 3)\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = 0$

$x + 2y + 3 = 0$

$2 + 2y + 3 = 0$ $(\because x = 2)$

$2y + 5 = 0$

$2y = 5$

$y = 5/2$

$\boxed{y = 2.5}$

f) fit the line using the implicit line equation.

Let the line equation homogeneous coordinate and point p on the line l

$l^T p = 0$

$E(l) = \Sigma (l^T p)^2$ (minimize objective function)

$E(l) = \Sigma l^T p_i \; p_i^T l \Rightarrow l^T \Sigma p_i p_i^T l \Rightarrow l^T S l$

$\nabla E(l) = 0 \Rightarrow Sl = 0$

where l is eigenvector of S

$S = \sum p_i p_i^T$ Correlation matrix of $P(x_i\ y_i\ 1)$ ~~input~~

$$S = \begin{bmatrix} \sum x_i^2 & \sum x_i y_i & \sum x_i \\ \sum x_i y_i & \sum y_i^2 & \sum y_i \\ \sum x_i & \sum y_i & n \end{bmatrix}$$

g) Given points $\{(0,1), (1,3), (2,6)\}$

Substitute in the above equation S

~~$S \Rightarrow$~~

$$S = \begin{bmatrix} 1+4 & 3+12 & 3 \\ 18 & 1+9+36 & 6+3+1 \\ 3 & 10 & 3 \end{bmatrix} \Rightarrow \begin{bmatrix} 5 & 15 & 3 \\ 15 & 46 & 10 \\ 3 & 10 & 3 \end{bmatrix}$$

h) Difference b/w Algebraic & geometric distance:

Algebraic distance :- Algebraic distance is based on the linear algebra and it is a distance b/w a point and a line in the vertical distance along the y axis. We can find the best fit line by minimizing the sum of squared algebraic distances.

geometric distance :- Here we compute the distance b/w a point and line in both x and y directions.

i) The geometric distance of point P from an implicit curve $f(P)=0$

$f(x)=0$

P = point

f = curve

$d(P,f) = \|P - x^*\|$ x^* being the closest point

$$d(P,f) \cong \frac{|f(P)|}{|\nabla f(x^*)|} \quad \text{(algebraic distance)}$$

The reason we ~~reduce the~~ approximate is to reduce the algebraic for points with larger gradients

$|\nabla f(P)|$

(ii) algebraic distance

$f(P)=1$

$\nabla f(P)=2$

$|f(P)|$ = algebraic distance of point

$\therefore$ algebraic distance $= 1$

(iii)

$f(P)=1$

$\nabla f(P)=2$

$$d(P,f) \approx \frac{|f(P)|}{|\nabla f(P)|} = \frac{1}{2} = 0.5$$

And this is approximate geometric distance.

(Ans)

l)

Error function:-

$$E[\phi(S)] = \int_{\phi S} \left(\alpha(S) E_{Continuity} + B(S)_{Curvature} + \gamma(S) E_{image}\right) ds$$

Where $\alpha(S)$, $B(S)$, $\gamma(S)$ are coefficients of different energy terms.

$\alpha E_{Continuity} + B_{Curvature} \rightarrow$ internal parameter

$\gamma E_{image} \rightarrow$ External parameter

Using the parameter Curve $\phi(S)$ to represent Curvature and gradually deform initial Curvature to fit the object

$$\phi(S) = \begin{bmatrix} x(S) \\ y(S) \end{bmatrix}$$

$$\phi(S) = \begin{bmatrix} \cos(S) \\ \sin(S) \end{bmatrix}$$

m) Given point, $P_1 = (1,2)$, $P_2 = (2,3)$ & $P_3 = (3,4)$

$$E_{Continuity} = (P_{i+1} - P_i)^2$$

$$= (P_2 - P_1)^2$$

$$= \sqrt{(3-2)^2 + (4-3)^2}$$

$$= 2$$

$E_{\text{Curvature}} = |P_{i+1} - 2P_i + P_{i-1}|^2$

$= |P_3 - 2P_2 + P_1|^2$

$= |(3,4) - 2(2,3) + (1,2)|^2$

$= |(4,6) - (4,6)|^2$

$= 0$

h) We have to take the value $\beta = 0$

when $|P_{i+1} - 2P_i + P_{i-1}| > 2$

to allow tight fitting or to allow precise Corners.

3

a) Outliers are extreme values that deviate from other observations on data, they may indicate a variability in a measurement, expirmental errors or a novelty.

Problems with outliers is that it influences the model fitting values estimates, the outliers donot fit value estimates; the outliers do not fit right model, we need to detect the outliers to make the model better.

Probability that all k expreiments failed :-

b) robust estimate objective function.

$E(\theta) = \sum \delta_\sigma (d(x_i, \theta))$

δ_σ :- MSE is a special are where $\delta_\sigma = x^2$

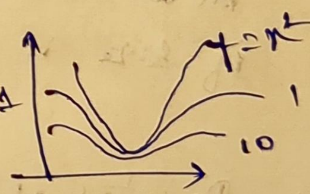
In standard least squares objective function outliers will have higher values & influence the model more ~~strongly~~ while in robust estimation σ_a will ~~rather~~ lower the influence of outliers on the model.

c) German - Mcclure estimation

$$\delta_\sigma(x) = \frac{x^2}{x^2 + \sigma^2}$$

It will lower the influence of outliers on model.

* If σ is large it will include more points else will include fewer points.



Advantages :-

* Reduces effect of outliers
* Capture the error

* Initially we will start with large σ value and decrease over the iterations.

d) $x = 1$ $\sigma = 1$

$$\delta_\sigma = \frac{x^2}{x^2 + \sigma^2} = \frac{1}{1+1} = \frac{1}{2} = 0.5$$

$$E(\theta) = \sum 0.5 d(x_i, \theta)$$

e) Random Sample Consensus (RANSAC)

* Perform multiple requirements
* Choose best results
* Use small set in hope that atleast one set will not have outliers

Repeat K-times :-

* Draw n points uniformly at random (with replacement)
* fit a model to points
* find inliers
* Recompute the model if atleast 'd' inliers

<u>Choose Best Solution:-</u>

The no.of points drawn at each attempt should be small As we need to minimize points to fit the model.
If size is big chances of having an outliers in it increases

f) Parameters of RANSAC

n = no.of points drawn at each evaluation

d = Minimum no.of points needed to estimate model.

K = no.of trials

t = distance threshold to identify inliers

$w = \frac{\# \text{ inliers}}{\# \text{ Points}}$ probability that a point is a inlier.

$w = 0.5$ initially

P = probability P that atleast 1 expirement does not have outliers

Probability that all k expriments failed :-

$$(1-P) = (1-W^n)^k$$

$$\log(1-P) = k \log(1-W^n)$$

$$k = \frac{\log(1-P)}{\log(1-W^n)}$$

q) Given,

$P = 0.99$

$W = 0.9$

assume $n = 1$

$$\therefore k = \frac{\log(1-0.99)}{\log(1-(0.9)^2)} = \frac{\log(0.01)}{\log(0.1)}$$

$$k = \frac{-2}{-1} = 2$$

$$\boxed{k = 2}$$

$\therefore$ no.of expirments needed to be performed in RANSAC algorithm is $\boxed{2}$