CS 512 HW3.

Li Xu
A20300818

1. corner detection

(a) ① find correlation matrix in local neighborhood.
② Find the eigenvalues of the matrix.
③ check if the $\lambda_1 \lambda_2 > \tau$, if yes, it's a corner.

(b) $$E(v) = \sum_{i=1}^{n} (p_i v)^2 = \sum_{i=1}^{n} (p_i v)(p_i v)$$
$$= \sum_{i=1}^{n} v^T p_i p_i^T v$$
$$= v^T \left(\sum_{i=1}^{n} p_i p_i^T \right) v.$$

To find $v^* = \underset{v}{\operatorname{argmin}}\, E(v)$, need to find derivative of $E(v)$ w.r.t v.

$\begin{bmatrix} \frac{\partial E}{\partial v_x} \\ \frac{\partial E}{\partial v_y} \end{bmatrix}$ ← gradient. $\Rightarrow$ Solve $\nabla E(v) = 0$ to find v^*.

(c) $$A = \sum_{i=1}^{n} p_i p_i^T$$
$$= \begin{bmatrix} x_i \\ y_i \end{bmatrix} \begin{bmatrix} x_i & y_i \end{bmatrix} = \begin{bmatrix} \Sigma x_i^2 & \Sigma x_i y_i \\ \Sigma x_i y_i & \Sigma y_i^2 \end{bmatrix}$$
$$= \begin{bmatrix} 4 & 6 \\ 6 & 44 \end{bmatrix}$$

(d) $\lambda_1 \lambda > \tau$, it's a corner.

(e) We can use non-maximal suppression to elimate points whose cornerness is not larger ~~enough~~ than those cornerness values of all points within a certain distance.

(f)

$c(G) = \det(G) - k\,\mathrm{trace}^2(G)$

And determiant and trace of correlation matrix can be compted by deriatives.

(g).

$E(p) = \sum_i (p_i - p)^T \nabla I(p_i) \nabla(p_i)^T (p_i - p)$

$\nabla E(p) = 2\sum_i \nabla I(p_i) \nabla I(p_i)^T (p_i - p) = 0$

$\Rightarrow \underbrace{\sum_i \nabla I(p_i) \nabla I(p_i)^T p_i}_{V} = \underbrace{\sum_i \nabla I(p_i) \nabla I(p_i)^T}_{C} p$

$\Rightarrow p = C^{-1} V$

if p is the corner, then C can be inversed.

(h) HOG is the histogram of oriented gradients, which counts how many ~~a~~ gradient in particular direction. In this way, each point of interest can ~~be~~ be characterlized from each other.

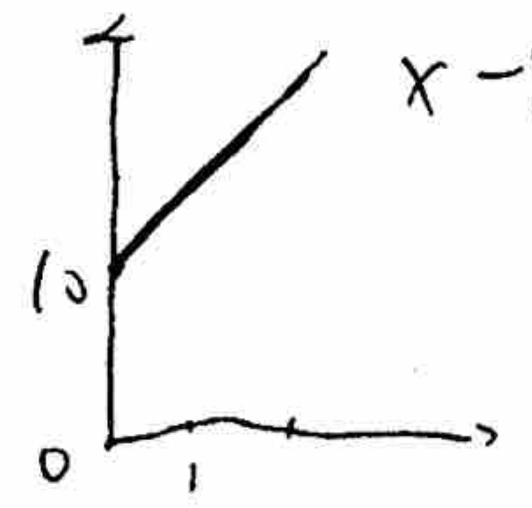
Requirement: translation invariant
rotational invariant.
scale invariant
illumination invariant.

(i) SIFT is Scale Invariant Feature Transform: First to compute difference of Gaussian of the image with different σ. Then search the local extrema to find key points, then use Taylor series expansion to get more accurate ~~of~~ location of extrema. Take the neighborhood of the extrema to compute gradiant and direction to form an orientation histogram. Then use this histogram to form a descriptor to do feature matching.

2. Line detection

(a) Using the slope and y-intercept can not represent vertical lines ; And ~~parameter~~ can take on infinite values

(b)



$x - y + 10 = 0$.

when $x = 0 \Rightarrow y = 10$

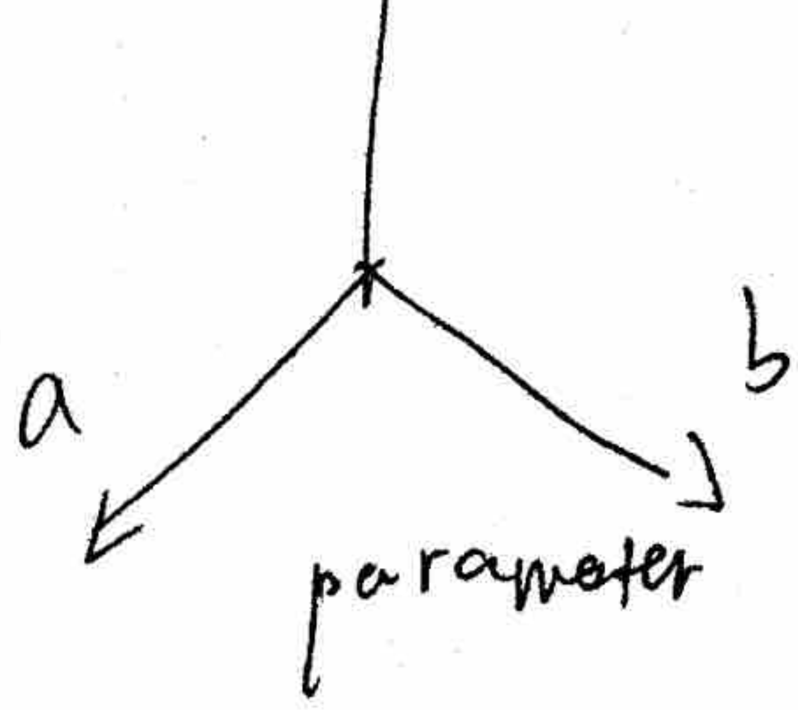
(c) In polar representation, a point in image plane looks like a curve ($x\cos\theta + y\sin\theta = \rho$)

(d) To check to votes in the parameter plane, ~~when votes~~ by incrementing the accumulator's bin to find local maxima in parameter space.

(e) We need to choose bin size carefully, e.g. if too small bin, we cannot find the peak.

(f) if the each vote point is known, we can use $\theta \in [\theta_{min}, \theta_{max}]$ to minimize the range to improve.

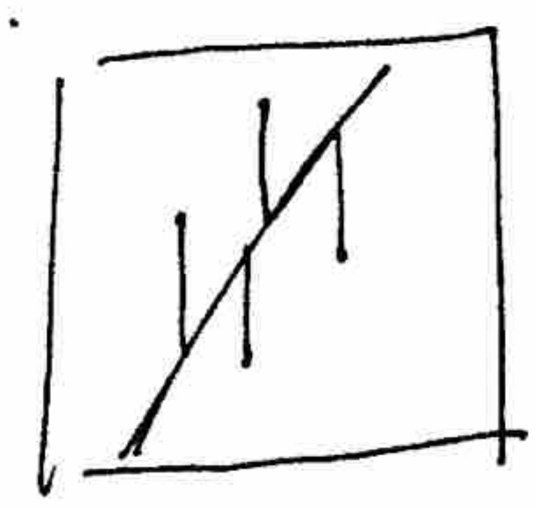
(g) $\begin{cases} a = x - r\cos\theta \\ b = y - r\cos\theta \end{cases}$ 3 dimensions



parameter space.

3. Model Fitting

(a) The line with a very large slope (almost vetical line) is very in-accurate.



(b) $(a, b) = (1, 2)$

$c = 2$

$l = [1\ 2\ 2] \Rightarrow l^T x = \begin{bmatrix} 1 \\ 2 \\ 2 \end{bmatrix} [x\ y\ 1] = 0$

$\Rightarrow x + 2y + 2 = 0$

(c) $l^T x = 0 \Rightarrow ab + by + c = 0$

$E(l) = \sum_{i=1}^{n} (l^T x_i^2)$

$= l^T \underbrace{(\sum_{i=1}^{n} x_i x_i^T)}_{C} l = l^T C l$

$\nabla E(l) = 0 \Rightarrow$ solve to find the line

(d) $C = \begin{bmatrix} 0+1+4 & 0+3+12 & 0+1+2 \\ 0+3+12 & 1+9+36 & 1+3+6 \\ 0+1+2 & 1+3+6 & 3 \end{bmatrix}$

$= \begin{bmatrix} 5 & 15 & 3 \\ 15 & 46 & 10 \\ 3 & 10 & 3 \end{bmatrix}$

(e) $ax^2 + bxy + cy^2 + dx + ey + f = 0$

$b^2 - 4ac < 0 \Leftrightarrow$ guarantee on ellipse.

(f) $E(l) = \underbrace{\sum_{i=1}^{n} (l^T p_i)^2}_{S} + \lambda(l^T C l + 1)$

$\nabla E(l) = 2Sl + \lambda 2Cl = 0$

$\Rightarrow$ solve for $Sl = -\lambda Cl$

The point close to short axis affects more in fitting.

(g) $E(l) = \sum_i \frac{|f(P_i; l)|}{|\nabla f(P_i; l)|}$

It's complicated because it's non-linear.

(h) $E[\phi s] = \int \underbrace{\alpha(s) E_{continuity} + \beta(s) E_{curvature}}_{internal} + \underbrace{\gamma(s) E_{img}}_{external}$

$E_{continuity} = \left|\frac{\partial \phi}{\partial s}\right|^2$, $E_{curvature} = \left|\frac{\partial^2 \phi}{\partial s^2}\right|^2$

$E_{img} = -|\nabla I|^2$

(i) $E_{continuity} = \left|\frac{\partial \phi}{\partial s}\right|^2$ can be estimated by $\sum_{i=1}^{n} |P_i - P_{i-1}|^2$

$E_{curvature} = \left|\frac{\partial^2 \phi}{\partial s^2}\right|^2$: $\sum_{i=1}^{n} |(P_{i+1} - P_i) - (P_i - P_{i-1})|^2$

(j) $|P_i - P_{i-1}|$ gets minimum 0 when point collapse to center.

So we need to the distance between points to be d

to prevent shrinking. $|P_i - P_{i-1}| - d$