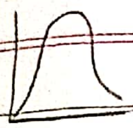
COMPUTER VISION



Gaussian Noise $\rightarrow n(x,y) = e^{-\frac{n^2}{2\sigma^2}}$

Derivative masks

One-D

Backward diff $[-1 \quad 1]$

Forward diff $[1 \quad -1]$

Central diff $[-1 \quad 0 \quad 1]$

two-D

$$f_x = \frac{1}{3}\begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$$

$$f_y = \frac{1}{3}\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$$

Correlation

$$f \otimes h = \sum_k \sum_l f(k,l)\, h(i+k, j+l)$$

Convolution

$$f * h = \sum_k \sum_l f(k,l)\, h(i-k, j-l)$$

(X-flip, Y-flip) OR 180° rotation

For Gaussian, Convolution = Correlation (∵ its symmetry)

Gaussian filter

$$g(x,y) = e^{-\frac{(x^2+y^2)}{2\sigma^2}}$$

$\sigma = 1$ $\quad g(x) = [.011 \quad .13 \quad .6 \quad 1 \quad .6 \quad .13 \quad .011]$

more σ, graph will flatten up

* Smooth
* Fourier transform of gaussian is gaussian
* Convolution of gaussian with itself is gaussian, with bigger σ value.

MATLAB funct

conv conv2 filter2 gradient mean
special eig
Gaussian smoothing

Edge detectors

Gradient operators

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

Laplacian of Gaussian (Mirr Hildreth)
Gradient of Gaussian (Canny)

Mirr Hildreth Edge detector

Gaussian smoothing

$$S = g * I \qquad g = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Laplacian

$$\Delta^2 S = \frac{\partial^2}{\partial x^2} S + \frac{\partial^2}{\partial y^2} S$$

$$\Delta^2 S = \Delta^2 (g * I) = (\Delta^2 g) * I$$

$$\Delta^2 G_\sigma = -\frac{1}{\sqrt{2\pi}\,\sigma^3}\left(2 - \frac{x^2+y^2}{\sigma^2}\right)e^{-\frac{x^2+y^2}{2\sigma^2}}$$

$$\Delta^2 S = \Delta^2 (g * I) = (\Delta^2 g) * I$$

n^2 $h(x,y) = I(x,y) * g(x,y)$

$2n$ $h(x,y) = (I(x,y) * g_1(x)) * g_2(y)$

$$g(x) = e^{-\frac{x^2}{2\sigma^2}}$$

n^2 $\Delta^2 S = \Delta^2 (g * I) = (\Delta^2 g) * I = I * (\Delta^2 g)$

$4n$ $\Delta^2 S = (I * g_{xx}(x)) * g(y) + (I * g_{yy}(y)) * g(x)$

4 convolutions require 4n multiplications

Algorithm: Compute LoG
- Use 2D filter $\Delta^2 g(x,y)$
- Use 4 1D filters $g(x), g_{xx}(x), g(y), g_{yy}(y)$

Find zero crossings from each row
Find slope of zero crossings
Apply threshold to slope and mark edges

Canny Edge Detector

1) Smooth image with gaussian filter
2) Compute derivative of filtered image
3) Find magnitude and orientation of gradient
4) Apply "Non-maximum suppression"
5) Apply "Hysteresis threshold"

Smoothing

$$S = I * g(x,y) = g(x,y) * I$$

$$g(x,y) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Derivative

$$\nabla S = \nabla(g * I) = (\nabla g) * I$$

$$\nabla S = \begin{bmatrix} g_x \\ g_y \end{bmatrix} * I = \begin{bmatrix} g_x * I \\ g_y * I \end{bmatrix}$$

$$\nabla g = \begin{bmatrix} \frac{\partial g}{\partial x} \\ \frac{\partial g}{\partial y} \end{bmatrix} = \begin{bmatrix} g_x \\ g_y \end{bmatrix}$$

Gradient magnitude and direction

(S_x, S_y) Gradient Vector

magnitude $= \sqrt{S_x^2 + S_y^2}$

direction $= \theta = \tan^{-1}(S_y / S_x)$

Scanned by CamScanner

Hysteresis.

connected

Gradient magnitude

Suppress the pixels in $|\nabla S|$ which are not local maximum

$$M(x,y) = \begin{cases} |\nabla S|(x,y) & \text{if } |\nabla S|(x,y) > |\Delta S|(x',y') \\ & \& \ |\Delta S|(x,y) > |\Delta S|(x'',y'') \\ 0 & \text{otherwise} \end{cases}$$

x' and x'' are the neighbours of x in the normal direction ~~of gradient~~ to an edge.

$f \otimes h$: Cross correlation.

$f \otimes f$ Auto-~~cross~~ correlation

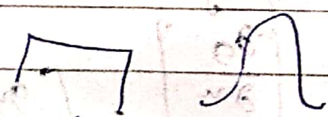
$$SSD = \sum_k \sum_l \left(f(k,l) - h(i+k, j+l)\right)^2$$
minimise

$$\text{Correlation} = \sum_k \sum_l \left(h(i+k, j+l) f(k,l)\right)$$
maximize

Harris detector

$$E(u,v) = \sum_{x,y} \underbrace{w(x,y)}_{\text{window function}} \Big[\underbrace{I(x+u, y+v)}_{\text{shifted intensity}} - \underbrace{I(x,y)}_{\text{intensity}}\Big]^2$$

$w(x,y) \Rightarrow$



uniform gaussian

Taylor series

$$f(a) + \frac{f'(a)}{1!}(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \frac{f'''(a)}{3!}(x-a)^3 + \cdots$$

$f(x)$ can be represented at point a

Approximation using Taylor series

$$E(u,v) = \sum_{x,y} w(x,y)\left[I(x,y) + uI_x + vI_y - I(x,y)\right]^2$$

$$E(u,v) = \sum_{x,y} w(x,y)\left[uI_x + vI_y\right]^2$$

$$E(u,v) = \sum_{xy} w(x,y)\left[\begin{pmatrix} u & v \end{pmatrix}\begin{pmatrix} I_x \\ I_y \end{pmatrix}\right]^2 \qquad \because A^2 = AA^T,\ (AB)^T = B^T A^T$$

$$E(u,v) = \sum_{xy} w(x,y)\begin{pmatrix} u & v \end{pmatrix}\begin{pmatrix} I_x \\ I_y \end{pmatrix}\begin{pmatrix} I_x & I_y \end{pmatrix}\begin{pmatrix} u \\ v \end{pmatrix}$$

$$E(u,v) = \begin{pmatrix} u & v \end{pmatrix}\left[\sum_{xy} w(x,y)\begin{pmatrix} I_x \\ I_y \end{pmatrix}\begin{pmatrix} I_x & I_y \end{pmatrix}\right]\begin{pmatrix} u \\ v \end{pmatrix}$$

$$E(u,v) = \begin{pmatrix} u & v \end{pmatrix} M \begin{pmatrix} u \\ v \end{pmatrix}$$

$$M = \sum_{xy} w(x,y)\begin{bmatrix} I_xI_x & I_xI_y \\ I_xI_y & I_yI_y \end{bmatrix}$$

$E(u,v)$ is an equation of an ellipse, where M is the covariance

Let λ_1 and λ_2 be the eigen values of M

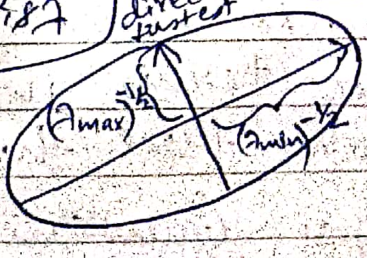
$Ax = \lambda x$ property

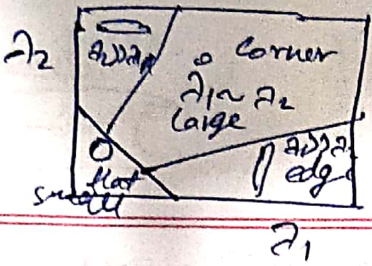
Calculation

$\det(A - \lambda I) = 0$ Step 1 find roots

$(A - \lambda I)x = 0$ Step 2 solve linear system for each root to find corresponding eigen vector

eigen vector is x
eigen value is λ

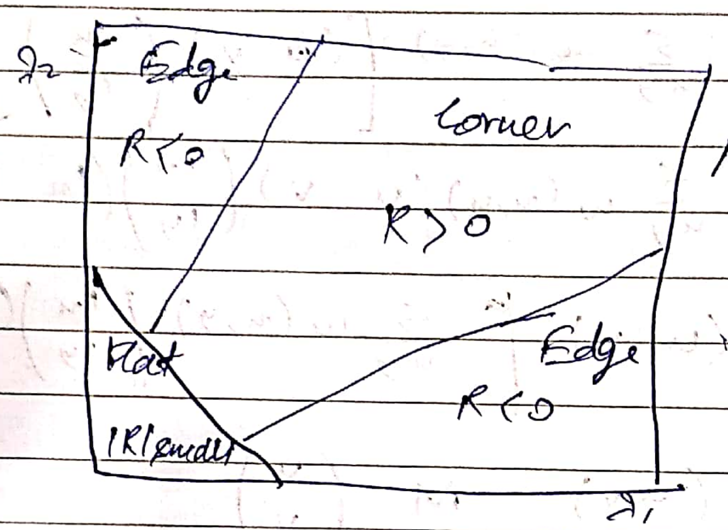




$$M = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}$$

$$R = \det M - k(\text{trace } M)^2$$

$$R = \lambda_1\lambda_2 - k(\lambda_1 + \lambda_2)^2$$

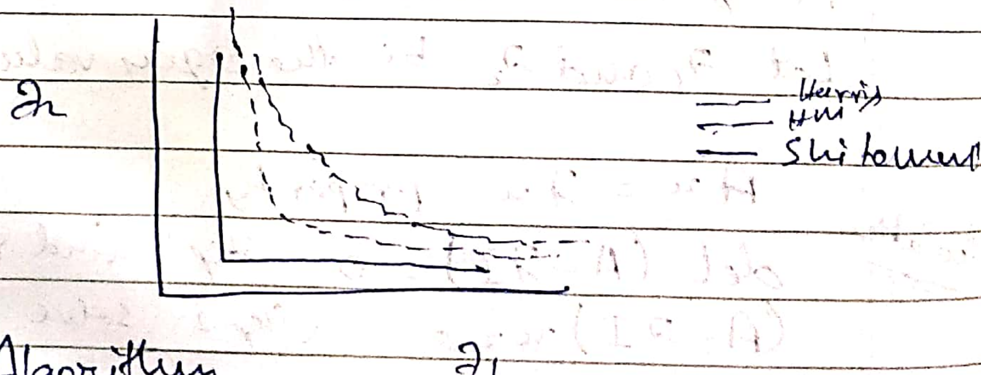


R depends only on eigen values of M

$R = \lambda_1 - \alpha\lambda_2$ Triggs

$R = \dfrac{\det(M)}{\text{trace}(M)} = \dfrac{\lambda_1\lambda_2}{\lambda_1 + \lambda_2}$ Szeliski Harmonic mean

$R = \lambda_1$ Shi Tomasi



Algorithm

1. Compute I_x, I_y
2. Compute three images corresponding to three terms in matrix M
3. Convolve these three images with a larger Gaussian (window)

x_4 Compute scalar cornerness value using one of the R measures.

x_5 Find local maxima above some threshold as detected interest points.

SIFT

1) Apply whole spectrum of scales
2) Plot zerocrossings vs scales in a scale-space
3) Interpret scale space contours

Contours are arches, open at the bottom, closed at the top

Interval tree

Each interval corresponds to a node in a tree, whose parental node represents larger interval, from which interval emerged, and whose offsprings represent smaller intervals.

Stability of a node is a scale range over which the interval exists.

4) Top level description

Iteratively removes nodes from the tree, splicing out nodes that are less stable than any of their parents & offsprings.

LoG

Interest points: Local maxima in scale space of LoG

Scanned by CamScanner

Approximation of LoG

$$\frac{\partial G}{\partial \sigma} = \sigma \Delta^2 G \quad \text{Heat equation}$$

$$\sigma \Delta^2 G = \frac{\partial G}{\partial \sigma} = \frac{G(x,y,k\sigma) - G(x,y,\sigma)}{k\sigma - \sigma}$$

$$G(x,y,k\sigma) - G(x,y,\sigma) \approx (k-1)\sigma^2 \Delta^2 G$$

Typical values: $\sigma = 1.6$ $k = \sqrt{2}$

$$G(x,y,k\sigma) = \frac{1}{2\pi(k\sigma)^2} e^{-\frac{(x^2+y^2)}{2k^2\sigma^2}}$$

Resolution is higher for lower level
Impact of filtering will be twice

classmate

Date ______

Page ______

Optical Flow

Brightness constancy assumption

$f(x, y, t) = f(x+dx, y+dy, t+dt)$

$\downarrow$ Taylor series

$f(x,y,t) = f(x,y,t) + \frac{\partial f}{\partial x} dx + \frac{\partial f}{\partial y} dy + \frac{\partial f}{\partial t} dt$

$f_x dx + f_y dy + f_t dt = 0$

$f_x u + f_y v + f_t = 0$