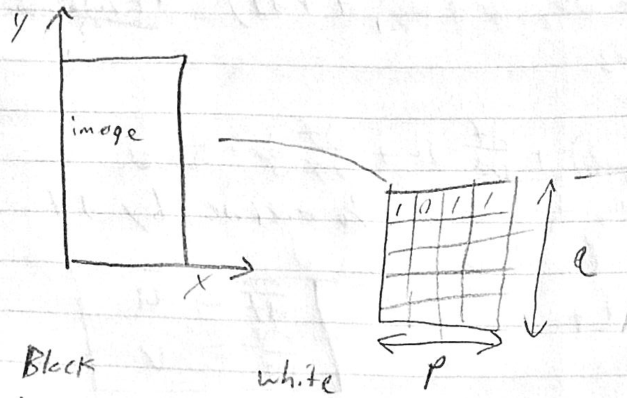
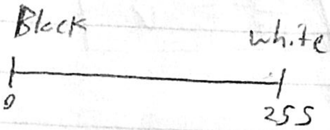
Image filtering





- color image 3 matricies
  RBG

- <u>Image Noise</u>
  - Light Variations
  - Camera Electronics
  - Surface reflections
  - Lens

– Noise is random, occurs with some probability

$$\hat{I}(x,y) = I(x,y) + n(x,y)$$ additive noise

<u>Image ~~der~~ derivatives</u>

$$f'(x) = \frac{f(x) - f(x-h)}{h}$$

$f(x) = 10 \quad 15 \quad 10 \quad 10 \quad 25 \quad 20 \quad 20 \quad 20$
$f'(x) = 0 \quad 5 \quad -5 \quad 0 \quad 15 \quad -5 \quad 0 \quad 0$

Gradient vector

$$\nabla f(x,y) = \begin{bmatrix} \frac{\partial f(x,y)}{\partial x} \\ \frac{\partial f(x,y)}{\partial y} \end{bmatrix} = \begin{bmatrix} f_x \\ f_y \end{bmatrix}$$

Gradient magnitude

$$|\nabla f(x,y)| = \sqrt{f_x^2 + f_y^2}$$

2 1 0 1 2

Derivative masks

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

$$I = \begin{bmatrix} 10 & 10 & 20 & 20 & 20 \\ 10 & 10 & 20 & 20 & 20 \\ 10 & 10 & 20 & 20 & 20 \\ 10 & 10 & 20 & 20 & 20 \\ 10 & 10 & 20 & 20 & 20 \end{bmatrix}$$

$$I_x = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 10 & 10 & 0 & 0 \\ 0 & 10 & 10 & 0 & 0 \\ 0 & 10 & 10 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

<u>correlation</u>

$$f * h = \sum_K \sum_I f(k,l)\, h(k,l)$$

f = Images
h = kernel (mask)

☆ try guassion smoothing filter

- [FX, FY] = gradient(A)

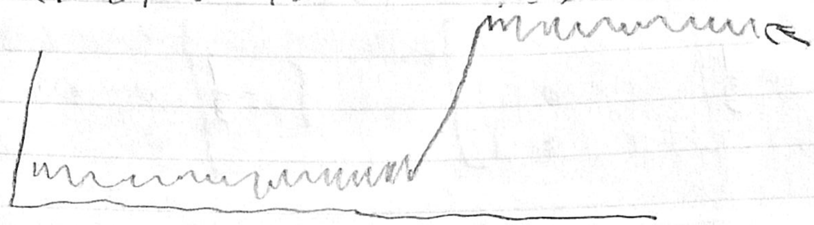
<u>Guassion filter</u> $g(x) = e^{\frac{-x^2}{2\sigma^2}}$ $\qquad g(x,y) = e^{-\frac{(x^2+y^2)}{2\sigma^2}}$
(smooths image)

Guassion mask $g(x) = [.011 \;\; .13 \;\; .6 \;\; 1 \;\; .6 \;\; .13 \;\; .011]$

$$\begin{bmatrix} 1 & 3 & 0 \\ 2 & 10 & 2 \\ 4 & 1 & 1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 & -1 \\ 1 & .1 & -1 \\ 1 & 0 & -1 \end{bmatrix} = \begin{bmatrix} & & \\ & 5 & \\ & & \end{bmatrix}$$

Edge detection
- use derivative to find edge

$f(x)$

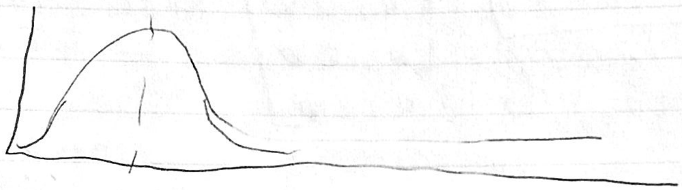


noise

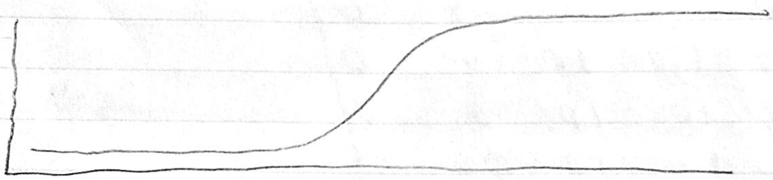
0 2000 pixle

- remove noise with gause filter

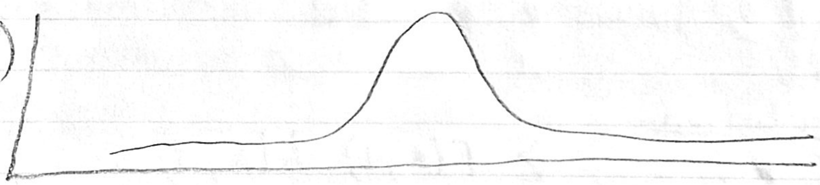
$\frac{d}{dx} g(x)$



$f * g$



$\frac{d}{dx}(f * g)$



- associative

$$\frac{d}{dx}(f * g) = f * \frac{d}{dx} g$$

☆ practice generating 2D gaussian

Intrest point detection

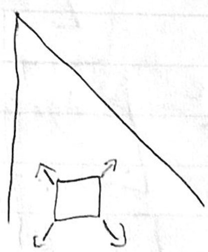
Trade offs

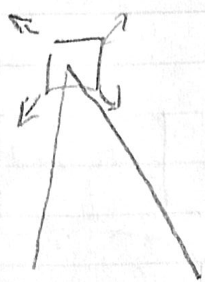
Detection of intrest points

$\longleftrightarrow$

More Repeatable
- Robust detection
- Precise localization

More Points
- Robust to occlusion
- Works with less texture





Flat region
- no obvious change in any direction

Corner
- Unique change in all directions

Correlation

$f \otimes h = \sum_K \sum_l f(K,l)\,h(K,l)$ % cross correlation

f = Image

h = Kernel

$$\overset{f}{\begin{bmatrix} f_1 & f_2 & f_3 \\ f_4 & f_5 & f_6 \\ f_7 & f_8 & f_9 \end{bmatrix}} \otimes \overset{h}{\begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix}}$$

$f * h = f_1 h_1 + f_2 h_2 + f_3 h_3 \ldots$

$f \otimes f = \sum_K \sum_l f(K,l)\,f(K,l)$

Sum of Square difference

$$SSD = \sum_{k}\sum_{l} (f(k,l) - h(k,l))^2$$

*minimize

$$SSD = \sum_{k}\sum_{l} (\cancel{f(k,l)^2} - 2h(k,l)f(k,l) + \cancel{h(k,l)^2})$$

$^1/_0$ terms^2 don't play role in correlation because independent

$$SSD = \sum_{k}\sum_{l} (-2h(k,l)f(k,l))$$

minimize

$$SSD = \sum_{k}\sum_{l} (2h(k,l)f(k,l))$$ ✓

maximize

<u>Harris detector</u>

- look for change of intensity for the shift (u,v)

$$E(u,v) = \sum_{xy} [\underbrace{I(x+u,y+v)}_{\text{Shifted Intensity}} - \underbrace{I(x,y)}_{\text{intensity}}]^2$$

<u>Taylor Series</u>

$$f(x) = f(a) + (x-a)f_x + \frac{(x-a)^2}{2!}f_{xx} + \frac{(x-a)^3}{3!}f_{xxx} + \dots$$

$$I(x+u, y+v) = I(x,y) + I_x u + I_y v$$ 2.ed series

$$E(u,v) = \sum_{xy} [\cancel{I(x,y)} + I_x u + I_y v - \cancel{I(x,y)}]^2$$

$$E(u,v = \sum_{xy} [I_x u - I_y v]^2$$

$$E(u,v) = \sum_{xy} \left[(u,v)\begin{pmatrix} I_x \\ I_y \end{pmatrix}\right]^2$$

$$M = \sum_{x,y} \begin{bmatrix} I_x I_x & I_x I_y \\ I_x I_y & I_y I_y \end{bmatrix}$$

Eigen Vectors Eigen Values

- The eigen vector x of matrix A is a special vector, with the following property

  $Ax = \lambda x$ % λ is eigen value

- To find eigen values of matrix A first find the roots of

  $\det(A - \lambda I) = 0$

- Then solve the following linear system for each eigen value to find corresponding eigen vector

  $(A - \lambda I)x = 0$

MATLAB function

[vector C, value C] = eig(C)

ex

Show that $\vec{x} = \begin{bmatrix} 2 \\ 1 \end{bmatrix}$ is an eigenvector of $A = \begin{pmatrix} 3 & 2 \\ 3 & -2 \end{pmatrix}$ corresponding to $\lambda = 4$

$$A\vec{x} = \lambda \vec{x}$$

$$\begin{pmatrix} 3 & 2 \\ 3 & -2 \end{pmatrix} \begin{pmatrix} 2 \\ 1 \end{pmatrix} = 4 \begin{pmatrix} 2 \\ 1 \end{pmatrix}$$

$$\begin{pmatrix} 6+2 \\ 6-2 \end{pmatrix} = \begin{pmatrix} 8 \\ 4 \end{pmatrix}$$

$$\begin{pmatrix} 8 \\ 4 \end{pmatrix} \checkmark = \begin{pmatrix} 8 \\ 4 \end{pmatrix}$$

$$A(\alpha \vec{x}) = \lambda(\alpha \vec{x})$$ α is non zero number

Harris detector