Week 9 15 oct

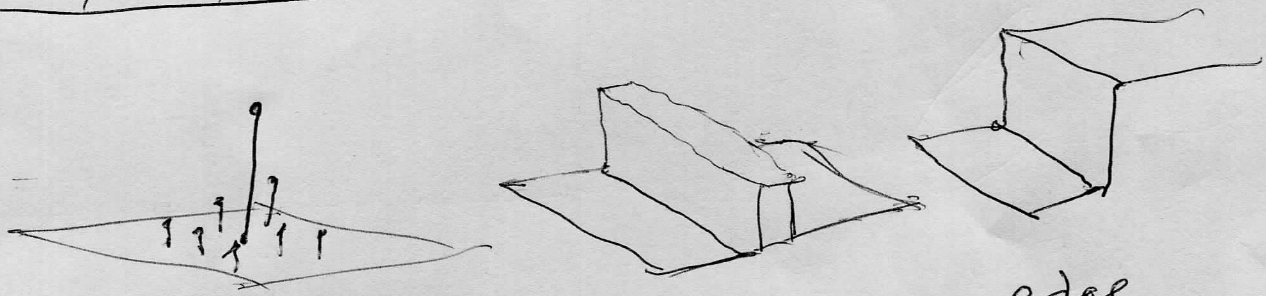
segmentation: find consistent semantic regions

maybe connected or not

major approaches:

* discontinuity : edge, split
* similarity : region, merge

point, line, edge detection



use differential operators

point : Laplacian

line : Laplacian responds according to width of line

see CS6640_week9

edge : + magnitude of gradient

+ canny

9/2

Marr-Hildreth

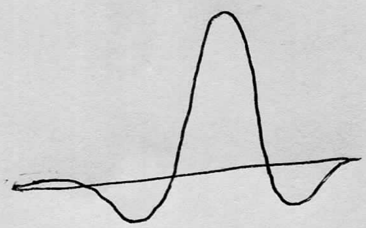
* get 1^{st} + 2^{nd} deriv
* tuned to scale

$\Rightarrow \nabla^2 G$ Laplacian of Gaussian

$$\nabla^2 f(x,y) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

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

$$\Rightarrow \nabla^2 G(x,y) = \left(\frac{x^2+y^2-2\sigma^2}{\sigma^4}\right) e^{-\left(\frac{x^2+y^2}{2\sigma^2}\right)}$$



$\equiv -\nabla^2 G$

(1) $g(x,y) = \left[\nabla^2 G(x,y)\right] \star f(x,y)$

(2) find zero-crossings

can be approximated by difference of Gaussians

$$D_G(x,y) = \frac{1}{2\pi\sigma_1^2} e^{-\left(\frac{x^2+y^2}{2\sigma_1^2}\right)} - \frac{1}{2\pi\sigma_2^2} e^{-\left(\frac{x^2+y^2}{2\sigma_2^2}\right)}$$

$\sigma_1 > \sigma_2$

same zero crossings if $\sigma^2 = \frac{\sigma_1^2 \sigma_2^2}{\sigma_1^2 - \sigma_2^2} \ln\left[\frac{\sigma_1^2}{\sigma_2^2}\right]$

Canny (see text)

* Low error rate
* localized edge points
* single edge point response

Linking edge points

(1) Join neighboring pixels if

$$|M(\bar{p}) - M(nei(\bar{p}))| \leq E$$

threshold

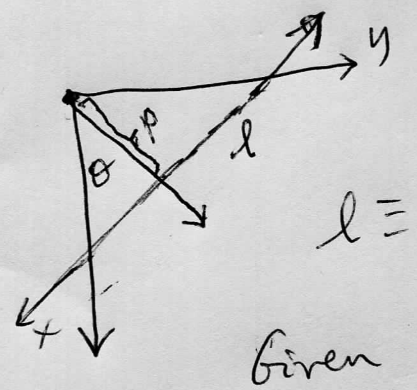
gradient mag

and $|\alpha(\bar{p}) - \alpha(nei(\bar{p}))| \leq A$

threshold

gradient orientation

Hough Transform: find lines

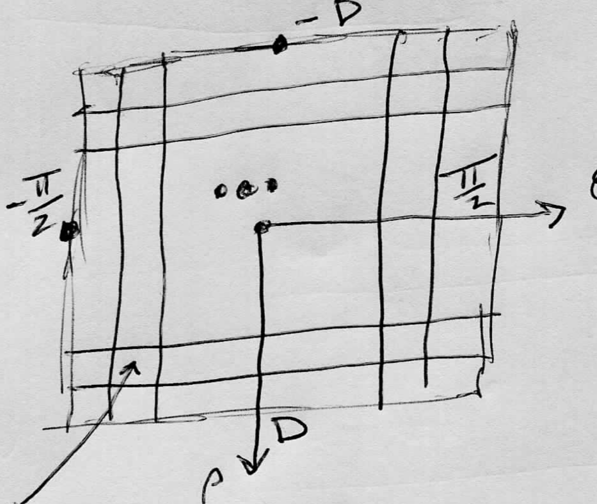


$$l \equiv x\cos\theta + y\sin\theta = \rho$$

Given an edge point (x,y), all lines through (x,y) are given by varying θ and computing ρ

(9/4)

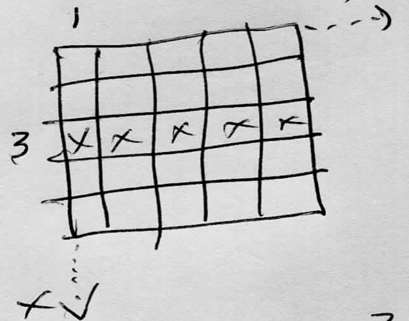
Consider: ρ-θ space
create discrete cells



every cell is a (θ, ρ) pair

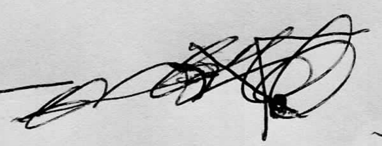
Given an (x,y) edge, every cell (θ, ρ) is incremented for which $\rho = x\cos\theta + y\sin\theta$

Consider:

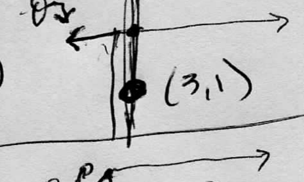


$(x,y) = (3,1) \Rightarrow \rho = 3\cos\theta + \sin\theta$

θ	ρ
$-\pi/2$	-1
$-\pi/4$	$\sqrt{2}$
0	3
$\pi/4$	$2\sqrt{2}$
$\pi/2$	1

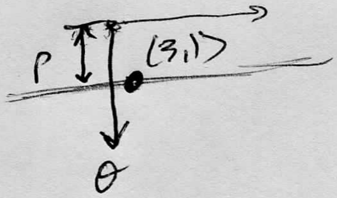


$(-\frac{\pi}{2}, -1)$



$(-\frac{\pi}{4}, \sqrt{2})$

$(0,3)$



Thresholding

Bimodal if mostly 2 classes
find value between them

Otsu's method : maximize between class variance

t = graythresh(im);
imt = imbinarize(im, t);

can use edge or zero-crossing locations to sample about equally from objects and background

Variable thresholding

Region Growing: + Merging + Splitting

need strong similarity measure

clustering : kmeans
we've seen this in A2 + A3

Simple Linear Iterative Clustering (SLIC)

uses pixel location as well

[L, NL] = superpixel(d, 40);
BW = boundarymask(L);
imshow(imoverlay(d, BW, 'red'));

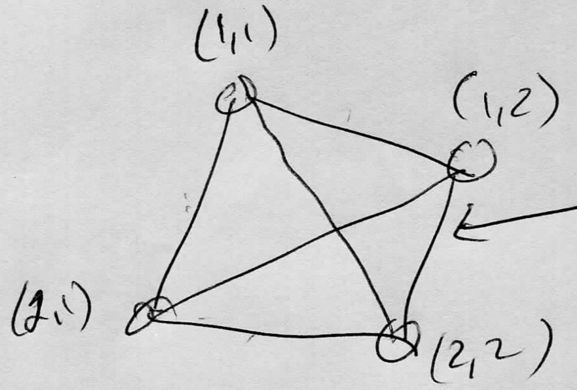
Graphs + Graph Cut

Consider pixels as nodes in a graph

edges connect all pixels

called a K-graph (complete graph)

2x2 image ⇒ 4 nodes +



put weights on edges

weights indicate how similar pixels are (e.g., gray levels or measures based on a window centered at pixel)

standard: sim(pixel1, pixel2) = $|g_1 - g_2|$

where g_1, g_2 are their gray levels

also: sim(pixel1, pixel2) = $\exp^{-|g_1 - g_2|}$

$W \equiv$ similarity matrix

$w(p_1, p_2) = \exp(-|g_1 - g_2|)$

Find eigenvalues + eigenvectors of W
Find 2 classes (use Kmeans)
Our object is in smaller of two classes

Watershed Variation

for every pixel, determine which neighbor has
smallest value
end
if a pixel has the smallest value among its
8-neighbors, then it's a basin origin;
label them
end
while some pixel has no label
check every pixel not labeled and if ~~it~~
~~has~~ its lowest value neighbor has a
value, change its value to that
end
find basin with largest spread,
+ return that.