Lecture 2.

Note Title 7/3/2011

Basic Properties of Images:

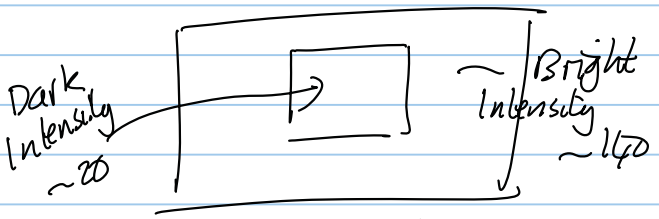
An image is an array of intensity values.
$\{I_{ij} : i = 1 \text{ to } n, j = 1 \text{ to } m\}$

What is this image? →

140	131	132	140
137	20	15	141
143	14	17	144
151	145	132	143

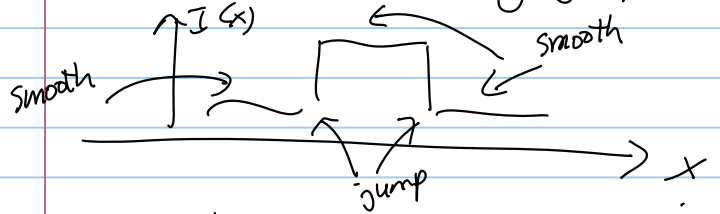
Probably you say:
"It is an image of a dark box surrounded by a bright background."

Why?



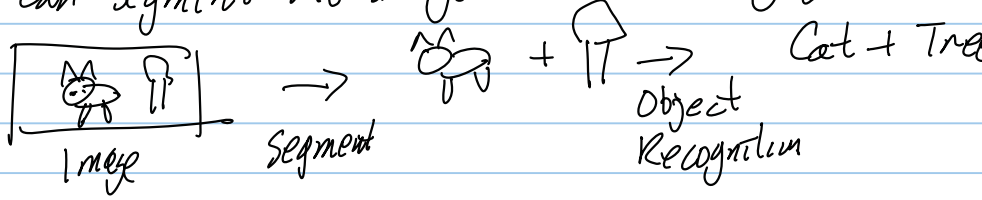
A simple model of images (~1980's) says that an image is piecewise smooth, or weakly smooth.

Neighboring pixels have similar intensity values (e.g. 20 to 15). but sometimes there is a big jump (e.g. 137 to 20)



A one-dimensional image.

This simple model says that images consist of regions. The intensity is roughly constant within each region. The intensity jumps between regions.
This can be used to segment images into regions.
This makes tasks like object detection easier.
– i.e. we can segment the image and then recognize objects.



Note: This model is too simple. Images are more complicated – e.g. they include texture regions.
In particular, we usually cannot segment objects without knowing what they are.

Are Images piecewise smooth?

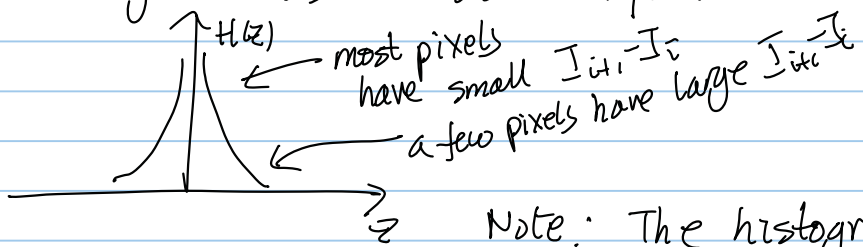
In one-dimension, calculate $\frac{dI}{dx}$ x-derivative. (or $I_{i+1} - I_i$ on lattice)

Calculate the histogram $\leftarrow$ identity

$$H(z) = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}(I_{i+1} - I_i = z)$$

$\mathbb{I}(a=b) = 1$, if $a=b$
$= 0$, otherwise

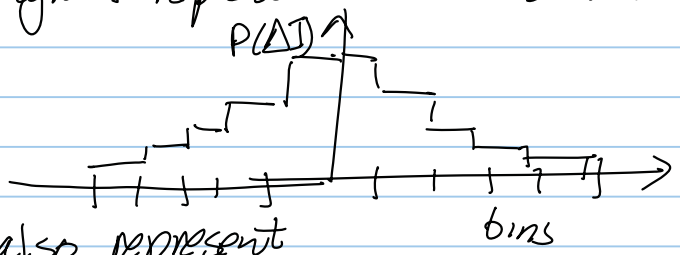
The histogram has the standard form.



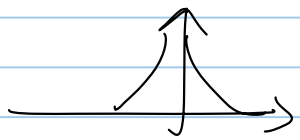
Note: The histogram is an image statistic – ie. a function on the image

It is the observed marginal distribution of the image derivative $p(\Delta I)$ $\Delta I \sim I_{i+1} - I_i$

The histogram represents this distribution by bins



Note: we could also represent $p(\Delta I)$ by a parameterized probability distribution – but we need to know the form of the distribution (it is not a Gaussian).

The statistics of $\frac{dI}{dx}$  suggest that images are piecewise smooth – at least as a very simple approximation.

We obtain similar statistics for two-dimension images – for $\frac{\partial I}{\partial x}$ and $\frac{\partial I}{\partial y}$. (Also for depth)

In fact, we get similar histograms for other derivatives. For $\frac{d^2I}{dx^2}$, $\frac{d^3I}{dx^3}$, .. See Mumford & Lee, Green.

(Note: higher order derivatives are non-local.)

Linear Filter Theory

$\delta_{ij} = 1$, if $i = j$
$= 0$, otherwise

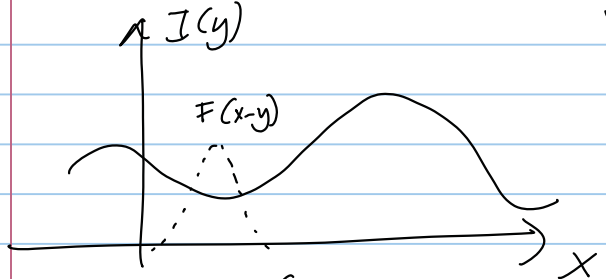
$F * I_i = \sum_j F_{i-j} I_j$ — Eg. $F_{i-j} = \delta_{i+1,j} - \delta_{i,j}$ gives the difference

(image → Filter)

$F * I_i = I_{i+1} - I_i$

A derivative-filter obeys $\sum_j F_{i-j} = 1$, for any i.

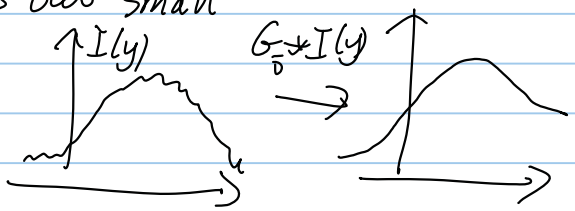
$F_{i-j} = \delta_{i+1,j} - 2\delta_{i,j} + \delta_{i+1,j}$ gives $I_{i+1} - 2I_i + I_{i-1}$ 2nd order derivative.



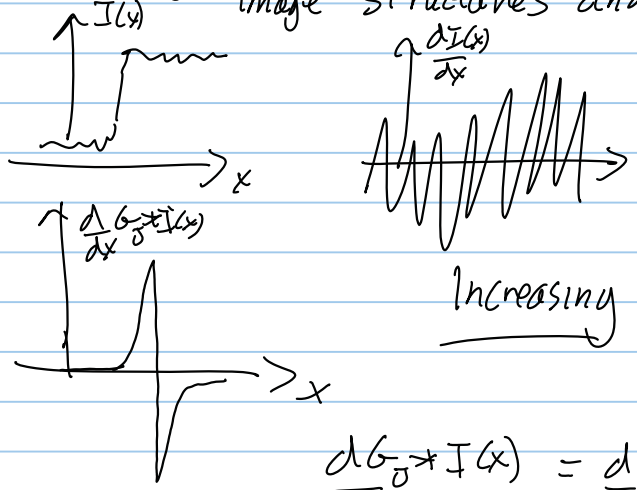
$F * I(x) = \int F(x-y) I(y)\, dy$

Example: Gaussian smoothing: $G_\sigma(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-x^2/2\sigma^2}$

This blurs the image and smooths out small structures in the image:



Often combine differentiation and smoothing. → e.g. smooth the image to eliminate small image structures and then differentiate.



Increasing σ in G_σ smooths the image more and removes more image structure.

$$\frac{d G_\sigma * I(x)}{dx} = \frac{d}{dx} \int G_\sigma(x-y) I(y)\, dy$$

Continuous Filters → Discrete Filters

e.g. $\frac{d}{dx} I(x)$ → $I_{i+1} - I_i$

Note: many ways to discretize derivatives. Some give better approximations.

FilterBanks

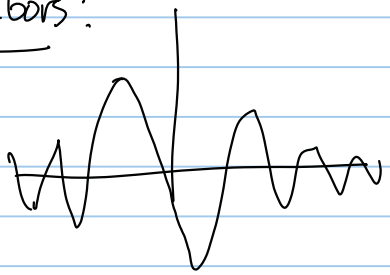
Sets of Filters:

- Derivatives, Smoothing, Derivatives and Smoothing.
- Gabor filters.

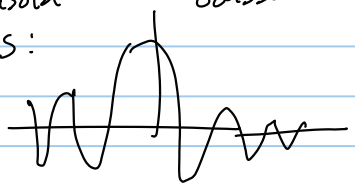
$$G_{ab}(\underline{x}) = e^{i\underline{\omega}\cdot\underline{x}} e^{-\frac{1}{2}\underline{x}^T \underline{\underline{\Sigma}}^{-1} \underline{x}}$$

sinusoid · Gaussian.

Sine Gabors:



Cosine Gabors:



Gabor's detect local frequency structure in images.

Color Images: $\{R_{ij}, B_{ij}, G_{ij}\}$ ij ← pixel location

R_{ij} ↑ red, B_{ij} ↑ blue, G_{ij} ↑ green

Normalized Color:

$$\frac{R_{ij}}{R_{ij}+B_{ij}+G_{ij}}$$

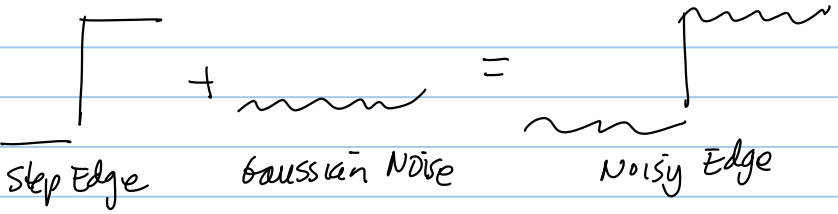
Lecture 3 Day 1.

Note Title 7/3/2011

Statistical Edge Detection.

Edge Detection - Canny Model.

Step Edge Model:

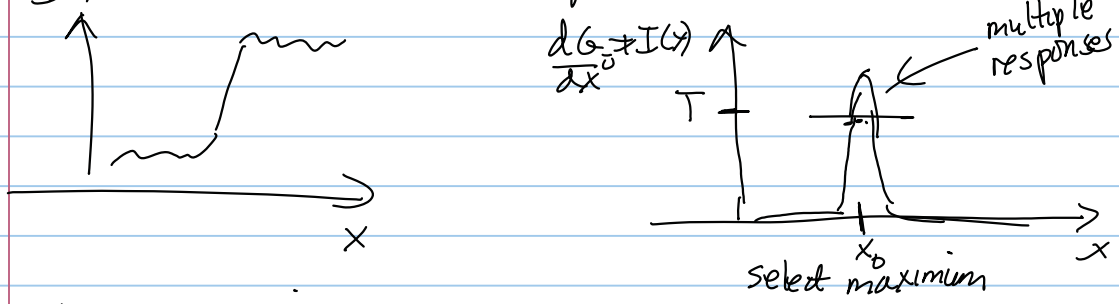


Solution:
- Apply a smoothed derivative filter.
- Threshold response
- Non-Maximal Suppression

E.G. One-Dimension $\left| \frac{d}{dx} G_\sigma * I(x) \right| > T$, Threshold T.

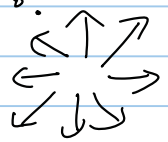
G_σ: Gaussian

If multiple responses above threshold select the maximum response and remove the rest.



In Two-Dimensions

Take Derivatives in many directions — eg.



Select maximum response.

Note: this method assumes a model of an edge. It does not try to learn an edge model from data.

Statistical Edge Detection:

Get dataset of images $\{ I^\mu(x) : \mu \in \Lambda \}$,

Pixels are labelled $\omega^\mu(x) = 1$, pixel x in μ^{th} image is an edge

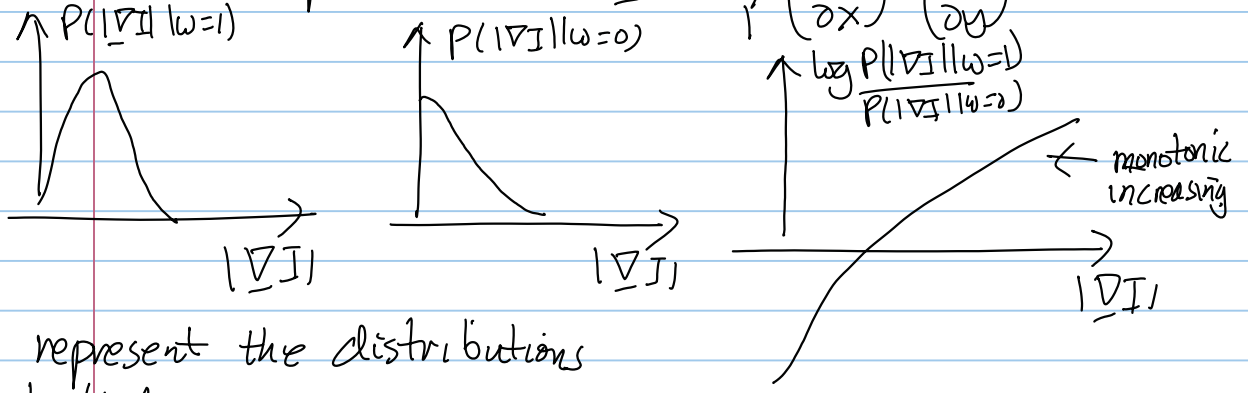
" " " " " is not an edge.

Learn conditional distributions:

$$P(\underline{F} * I(x) \mid \omega(x) = 1)$$
$$P(\underline{F} * I(x) \mid \omega(x) = 0)$$

$\underline{F} = (F_1, \ldots, F_m)$ derivative filters

Example: Filter $|\nabla I| = \sqrt{\left(\frac{\partial I}{\partial x}\right)^2 + \left(\frac{\partial I}{\partial y}\right)^2}$



represent the distributions by histograms.

To detect edges: Bayes decision

$$P(\omega | |\nabla I|) = \frac{P(|\nabla I| | \omega) P(\omega)}{P(|\nabla I|)}$$

Label a pixel as edge $\omega=1$, if $P(\omega=1 | |\nabla I|) > P(\omega=0 | |\nabla I|)$

$\omega=0$, otherwise.

This requires knowing $P(\omega)$.

Estimate from data $\quad P(\omega=1) \approx 0.08$

$P(\omega=0) \approx 0.92$

Because $\log \frac{P(|\nabla I| | \omega=1)}{P(|\nabla I| | \omega=0)}$ is monotonic in $|\nabla I|$

the Bayes decision is equivalent to thresholding $|\nabla I|$

which is similar to Canny.

But, suppose we apply several different filters.

$$P(|\nabla I|, |\nabla G_{\sigma_1} * I|, |\nabla G_{\sigma_2} * I| | \omega).$$

Learn these joint distributions from training examples.

Then apply Bayes.

This does much better than Canny when evaluated on difficult datasets.

Statistical edge detection can combine edge cues from multiple filters in an optimal manner.

Page 3: Statistical edge detection can combine information from multiple scales, or from multiple orientations.

Problem → if we represent $P(\vec{F}|\omega)$ by histograms, then we can need $O(k^m)$ data where k is no. of histogram bins and m is the number of filters. Requires too much data, and becomes impractical for large m. $(m \gg 10)$.

Note: less data is needed if we know a parameterized form for the distributions (typically polynomial no. of parameters). But we do not know the parameterized form of the distributions.

Learning: Learn on Training Dataset
evaluate on Test Dataset
Cross-Validation.

Bayes is a special case of Bayes Decision Theory. This includes a loss function, so that we can penalize the expected loss.

Note: that the Statistical approach described above is a generative model on filter response, but is not a generative model on images.

Alternative related approach (Martin, Fowlkes, Malik)
Extract features $\phi_1(I(x)), \phi_2(I(x)), \ldots, \phi_m(I(x))$
learn distribution $P(\omega | \phi_1 \ldots \phi_m) = \dfrac{e^{\omega \sum_{i=1}^{m} \lambda_i \phi_i(I(x))}}{e^{\sum_{i=1}^{m} \lambda_i \phi_i(I(x))} + e^{-\sum_{i=1}^{m} \lambda_i \phi_i(I(x))}}$
with $\omega \in \{+1, -1\}$,
Regression model - learn parameters $\{\lambda_i\}$.