COMP 435 Spring 2010

Review I

Outline for Review - 1

- ***Overview of Biometric Systems***
  - Biometric features
  - Performance of Biometric system
- ***Image Processing***
  - Pixel based: color conversion, intensity mapping ($y=x^r$), histogram equalization
  - Block based: smoothing, edge finding
  - Transforms: PCA, DCT, LDA, CCA
- ***Pattern Recognition***
  - kNN classifier
  - Bayesian classifier
  - Support Vector Machine & Kernel Method

Outline for Review – 2 (next lecture)

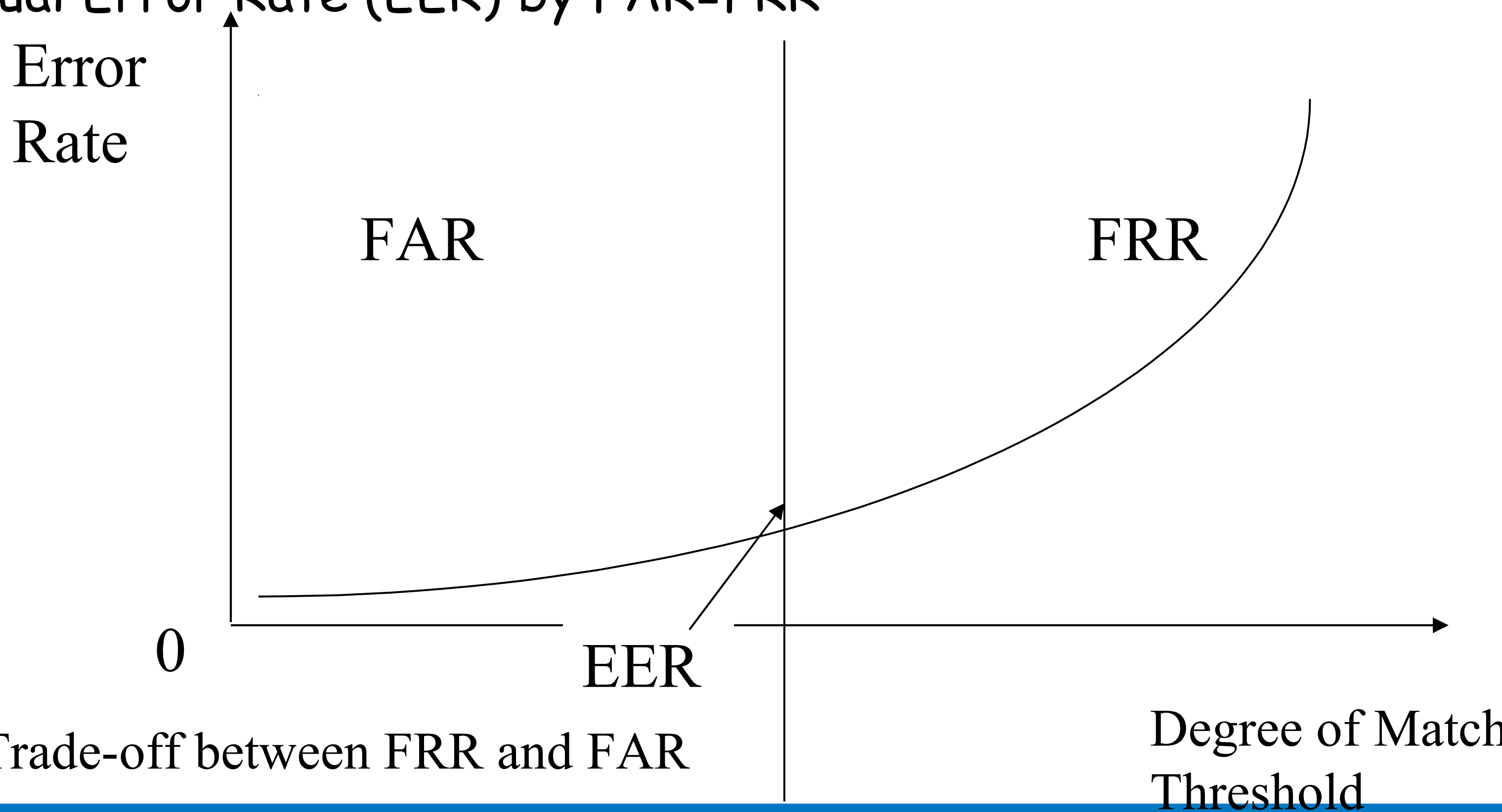
- **Biometric Techniques & Systems**
  - Face
    - » Detection - Boosting, Bootstrapping
    - » Recognition – Eigenface, Fisherface, Randomface
  - Fingerprint
    - » Singular point, Minutia
  - Palm
    - » PalmCode to encode directions at each pixel and matching by hamming distance, texture based
  - Iris & Retina
    - » IrisCode – texture feature based
  - Behavior & Multi Biometrics

Overview of Biometric Systems

- **4 features (UUPC) to be a viable biometric:**
  - Uniqueness – unique to the person for identification
  - Universality – everyone has this feature
  - Permanence – not changing over time
  - Collectability – cost/complexity in capturing biometric sensor data
- **Tasks:**
  - Identification: 1:N match, N can be huge (>300k in FBI suspect db), more difficult
  - Verification: 1:1 match, easier
- **Performance metrics:**
  - Accuracy,
  - Speed/Response Time
  - Storage efficiency

Performance Accuracy Metrics

- False Rejection Rate (FRR) : Type I Error Rate
  - FRR=m/n, n=total attempts, m = total number of authentic id rejected
- False Acceptance Rate (FAR) : Type II Error Rate
  - FAR = m/n: n=total attempts, m= total number of false id accpeted.
- Equal Error Rate (EER) by FAR=FRR



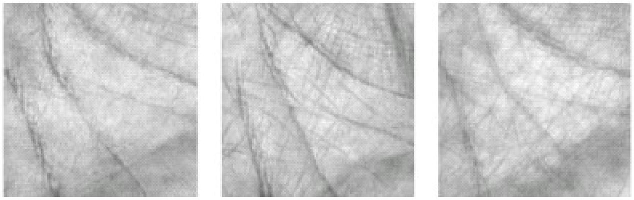
Trade-off between FRR and FAR

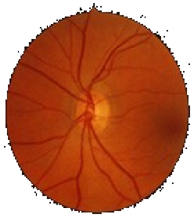
Image Analysis Outline

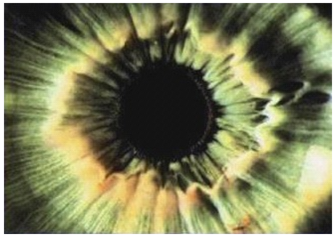
- **Images from biometric sensors**
- **Pixel based image processing**
  - Intensity mapping
  - Histogram Equalization
- **Filter based image processing**
  - Smoothing:
    - Gaussian, Median, Average
  - Edge Detection:
    - DoG, LoG, and why are we doing Gaussian ?
- **Transforms:**
  - PCA: un-supervised (no class labels), data dependent
  - DCT: un-supervised (no class labels), data independent
  - LDA: supervised (need class labels), data dependent
  - CCA: supervised (need class labels), data dependent

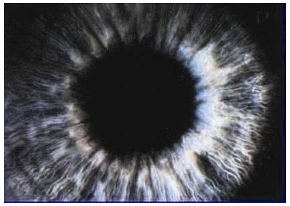
Images from Biometric Sensors

- Faces:
  - Holistic features (appearance)
- Fingerprint
  - Minutia, line features
- Palmprint
  - Texture analysis
- Iris
  - Texture analysis
- Retina
  - Line features









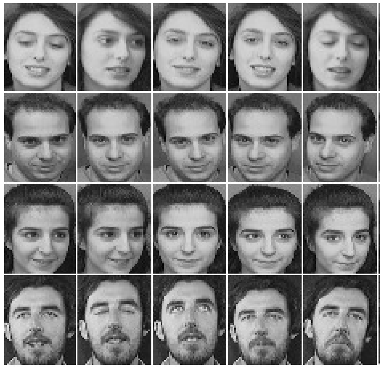


Image Features for Biometric Recognition

- **Face:**
  - Fisherface/Eigenface: appearance transform based approach
- **Fingerprint:**
  - Minutia: texture analysis, i.e, line/edge feature based
    - » Location (x, y), type (bifurcation, termination, …, etc), orientation (angle)
- **Palm:**
  - PalmCode: texture analysis, find dominant directions at each pixel
  - Matching by hamming code distance of coded angles.
- **Iris:**
  - IrisCode: localize iris, spread out as rectangles, texture analysis
  - Matching by hamming code distance

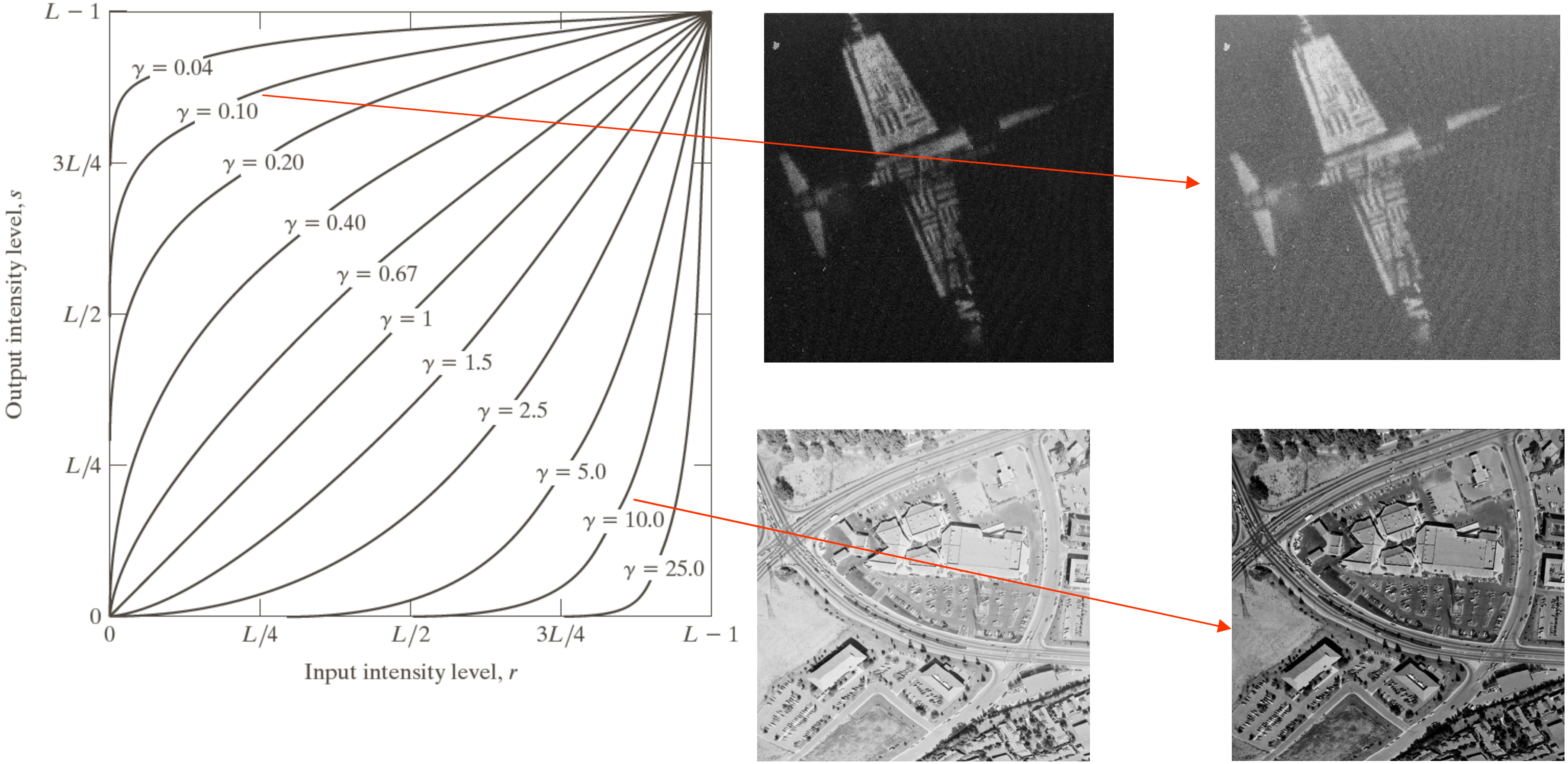
Pixel Based Image Processing

Pixel-Based Operations: Color Conversion

- Digital Color Image
  - Each pixel has 3 channels of senor data: Red, Green and Blue
  - Different mix of R, G and B levels give us different colors
  - For a given color pixel with value R,G and B, the gray level is given as:

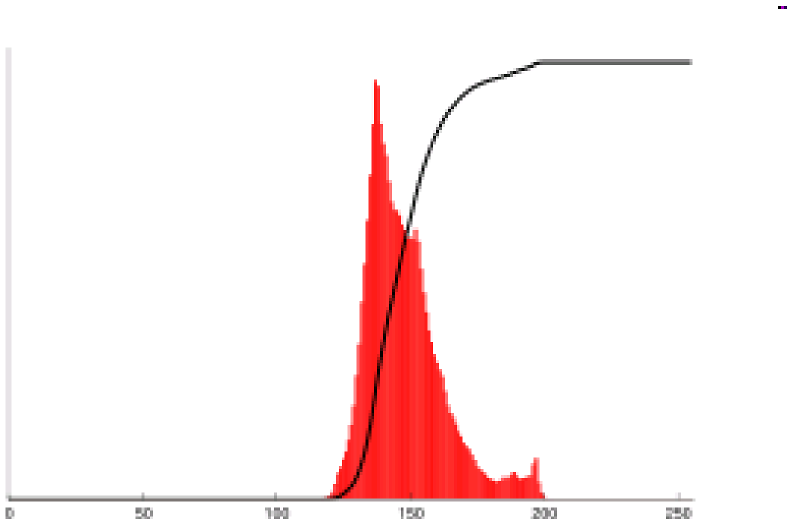
Gray = 0.3R+0.6G+0.1B

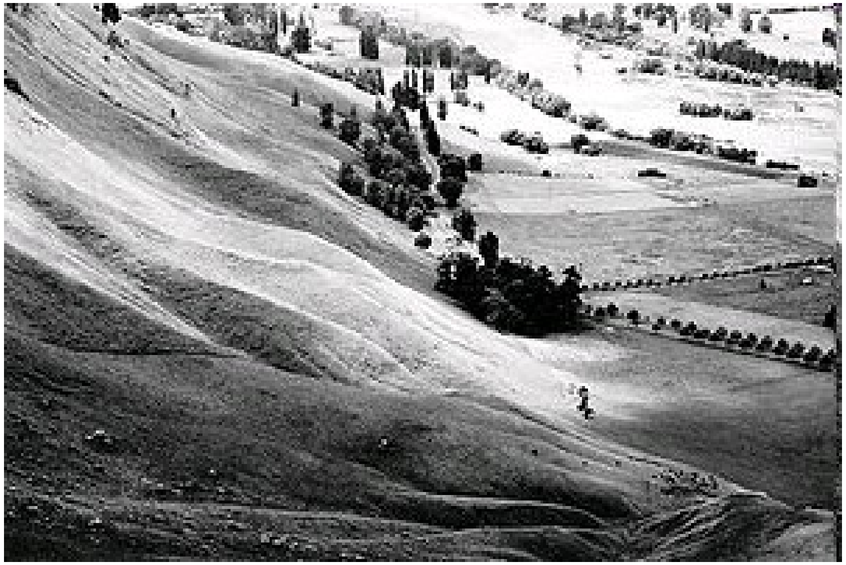
Pixel-Based Operation: Intensity Mapping

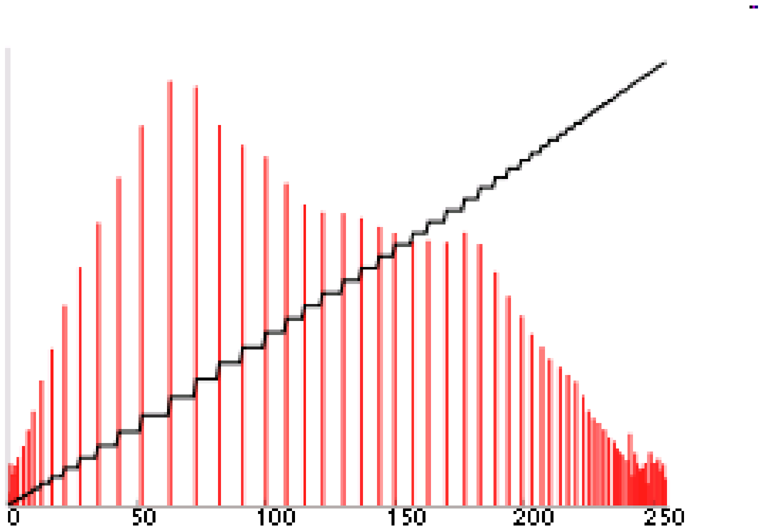


Power<1: enhancing under-exposed images
Power>1: enhancing over-exposed images

Histogram Equalization – to have constant slop CDF



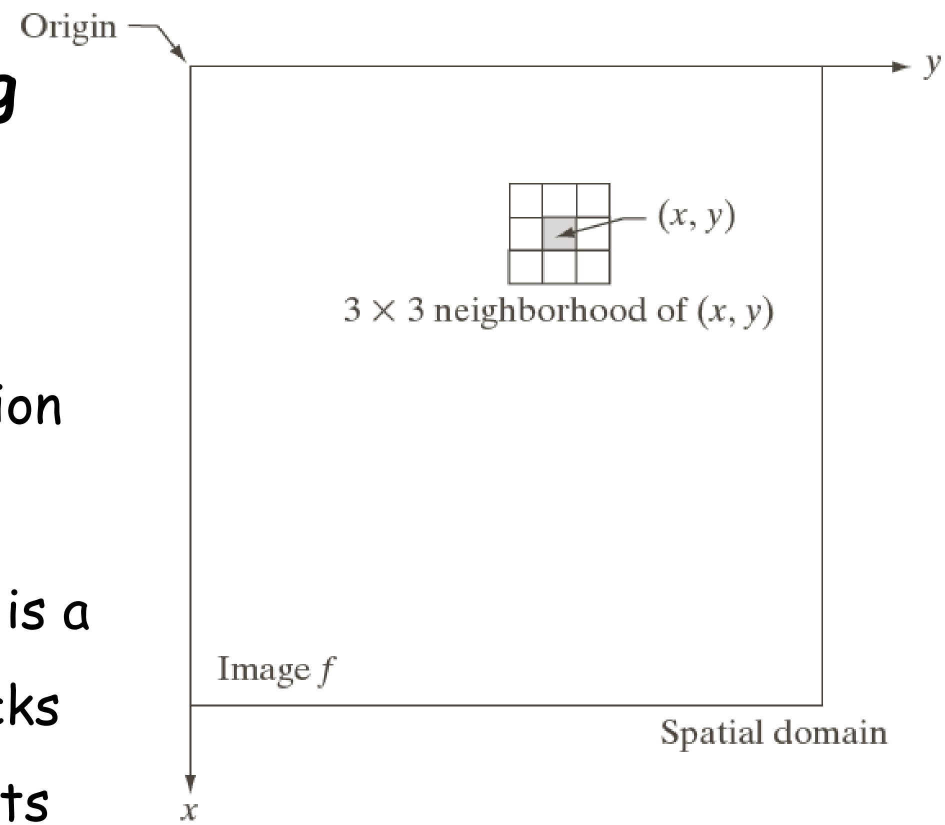




Filtering Based Image Processing

Re-Cap: Block based processing

- **Filter based image processing**
  - A filter mask of size m x m is identified
  - Center at input image pixel location (x,y)
  - Output image pixel value at (x,y) is a weighted sum of input image blocks centered around (x,y) with weights given by the mask
  - Matlab implementations:
    - » im2= Imfilter(im, mask)

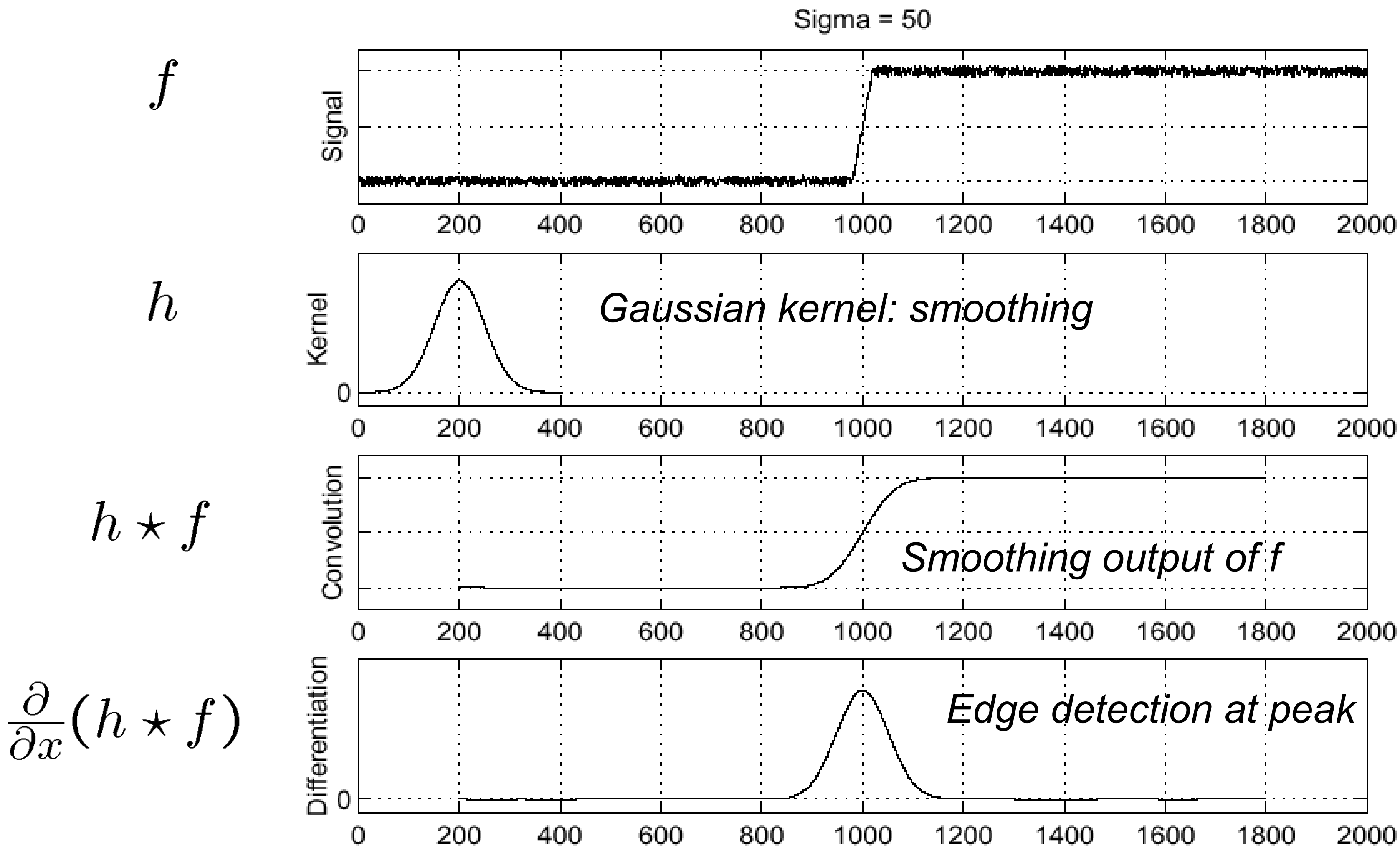


New image pixel value at (x,y): Weighted sum of 3x3 original image pixels

Re-Cap: Filter based Image processing

- **Smoothing/De-Noising**
  - Average, (filter kernel is flat)
  - Gaussian, (filter kernel is like a hat)
  - Median Filter (non-linear)
- **Edge detection**
  - Sobel/Prewitt filters: predefined.
  - 1st order approach: Derivative of Gaussian (DoG) filter:
    - » smoothing + differentiation, looking for peaks
  - 2nd order approach: Laplacian of Gaussian (LoG) filter:
    - » Smoothing + laplacian: look for zero-crossings

DoG Illustration



Where is the edge? At the peaks

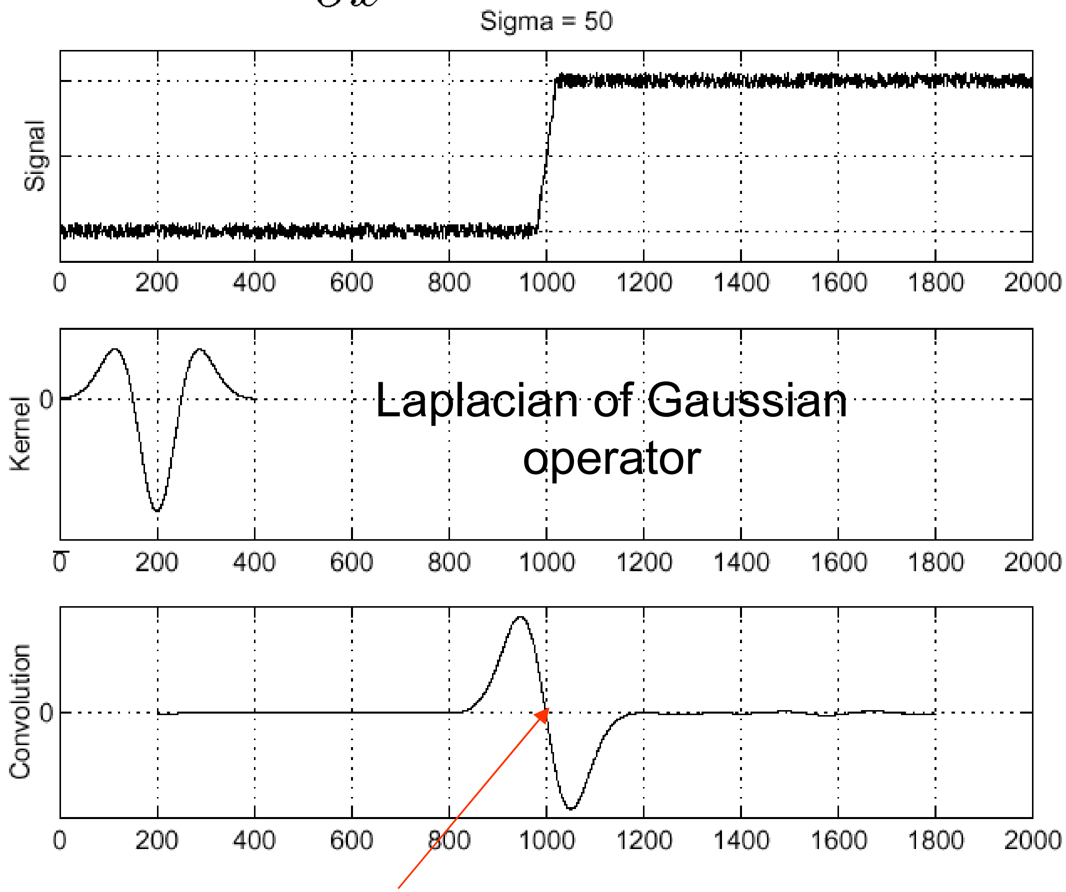
Laplacian of Gaussian

- **Consider** $$\frac{\partial^2}{\partial x^2}(h \star f)$$

f

$$\frac{\partial^2}{\partial x^2}h$$

$$(\frac{\partial^2}{\partial x^2}h) \star f$$



- Where is the edge? Look for zero crossing

LoG, Sobel and Prewitt

- Other edge detection results:

LoG

Sobel

Prewitt

LoG

Sobel

Prewitt

Sobel kernel:

1	2	1
0	0	0
-1	-2	-1

Prewitt kernel:

1	1	1
0	0	0
-1	-1	-1

Transform Based Image Processing

Preliminaries

- **Statistics: [review your homework 2 and quiz]**
  - What is mean ?
  - What is variance ?
  - What is co-variance ?
  - What is scatter ?
- **Linear Algebra:**
  - Transforms
  - Eigen vector and eigen values
    - » What does eigen vector and value mean ?

given data set $\{x_k\}$, $k=1..n$:

mean is: $m=$

$$\frac{1}{n}\sum_{k=1}^{n} x_k$$

variance for scalar array $\{x_k\}$ is:

$$\frac{1}{n-1}\sum_{k=1}^{n}(x_k - m)(x_k - m)$$

, why (n-1) ?

the best estimate if x is Gaussian.

Covariance between 2 scalar array $\{x_k\}$ and $\{y_k\}$ of the same length n:

Cov(X, Y) =

$$\frac{1}{n-1}\sum_{k=1}^{n}(x_k - m_x)(x_k - m_y)$$

Vector data scatter: $X=\{x_k\}$ in R^d. is given by a dx d matrix:

$S(X) = XX^T=$

$$\sum_{k=1}^{n}(x_k - m)(x_k - m)^T$$

Transforms

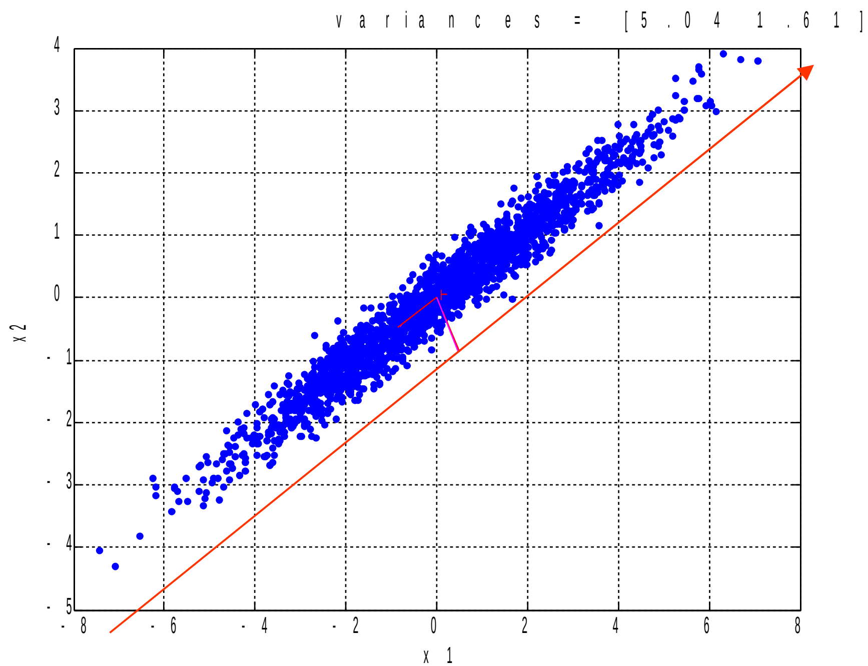
- **PCA – data dependent, un-supervised**
  - Given by eigen vectors of the covariance of the data set
- **DCT – data independent, un-supervised, can be viewed as an approximation of PCA**
  - Given as fixed bases of sinusoid.
- **LDA – data dependent, supervised**
  - Given as the bases that maximizes inter-class scatter over the intra-class scatter
- **CCA – data dependent, supervised,**

PCA

- Finds the directions where data has the max scatter/variance
- Unsupervised, data dependent
- Objective function:

$$S = S(X) = XX^T$$

$$\lambda W = SW$$

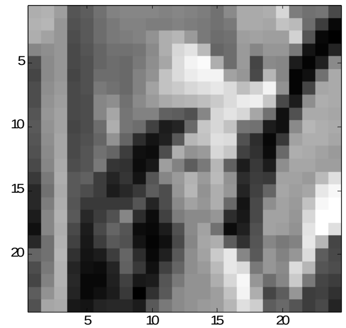


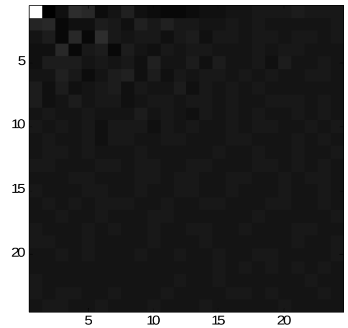
DCT

- An approximation of PCA
- Unsupervised, data dependent
- Basis of the transform is given (not data dependent) as:

$$X_k = \sum_{n=0}^{N-1} x_n \cos\left[\frac{\pi}{N}\left(n+\frac{1}{2}\right)k\right] \qquad k = 0, \dots, N-1.$$

- Application: Image Reconstruction with sparse coefficients
  - Matlab: im2=dct2(im1)





LDA

- Objective:
  - Find directions where the data projection has the largest inter-class scatter over the intra-class scatter
- Math Definitions & Formulation:

LDA

Let us have n samples, X={x_1, x_2, …, x_n} in R^d, belonging to m classes, $\{\varpi_1, \varpi_2, \cdots, \varpi_m\}$, each class j has n_j samples.

The class means are computed as,

$$m_j = \frac{1}{n_j}\sum_{x_k \in \varpi_j} x_k$$

The between class scatter is computed as:

$$S_B = \sum_{j=1}^{m} n_j (m_j - m)(m_j - m)^T$$

Then the within class scatter for class j is:

$$S_j = \sum_{x_k \in \varpi_j}^{n} (x_k - m_j)(x_k - m_j)^T$$, where

m_j is class mean.

The with-class scatter for all data points is,

$$S_W = \sum_{j=1}^{m} S_j = \sum_{j=1}^{m}\sum_{x_k \in \varpi_j} (x_k - m_j)(x_k - m_j)^T$$

LDA is to find transforms W, such that:

$$J(W) = \frac{W^T S_B W}{W^T S_W W}$$

is maximized.

S_W is d x d need at least d data points to be non-singular

- Solve by Eigs(S_B, S_W) in matlab

- **Canonical Correlation Analysis**
  - Find two transforms A, and B, such that the correlation between two feature set projections, AX, and BY, are maximized.
  - Input: two feature set $\{x_k\}$, $\{y_k\}$ for k=1..n,
  - Output: two transforms, A and B
  - Objective: maximize correlations between A and B: <AX, BY>
  - CCA is an supervised learning method
  - CCA is also data dependent.

Q&A regarding the reviews

- **Questions ?**