EE7403: A Postgraduate Course

Image Analysis and Pattern Recognition

Lecturer: **Jiang Xudong**

✉ : exdjiang@ntu.edu.sg

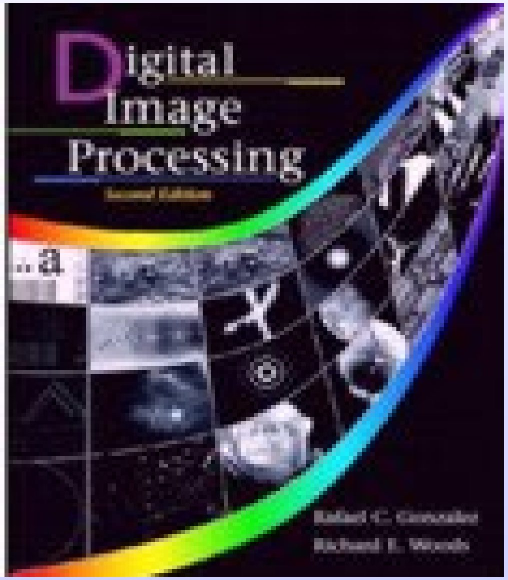
☎ : 67905018

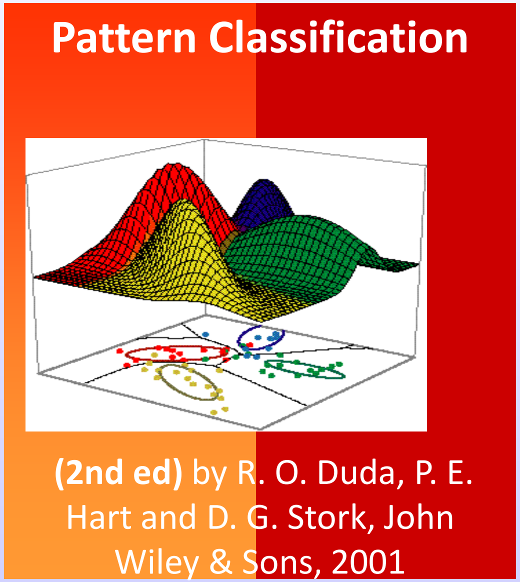
☺ : S1-B1c-105

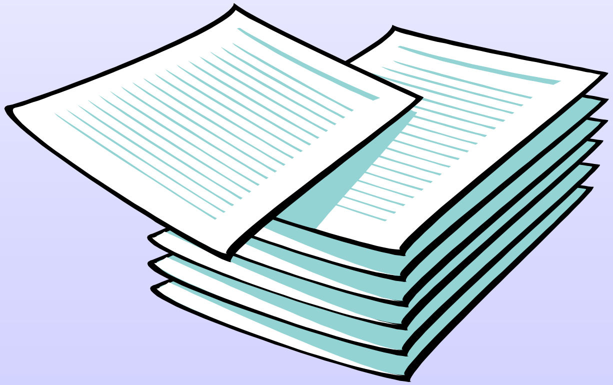
http://www.ntu.edu.sg/home/exdjiang

Textbooks, References & Assessments

- Gonzalez, R. C., Woods, R. E., *Digital Image Processing*, Prentice Hall
- R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, Wile Inter-science
- A number of published reputable *Research Papers*







- **Assessment scheme**
    - Continuous Assessment (CA) 20% by assignment
    - Final Examination 80%

Course Content

Topics:

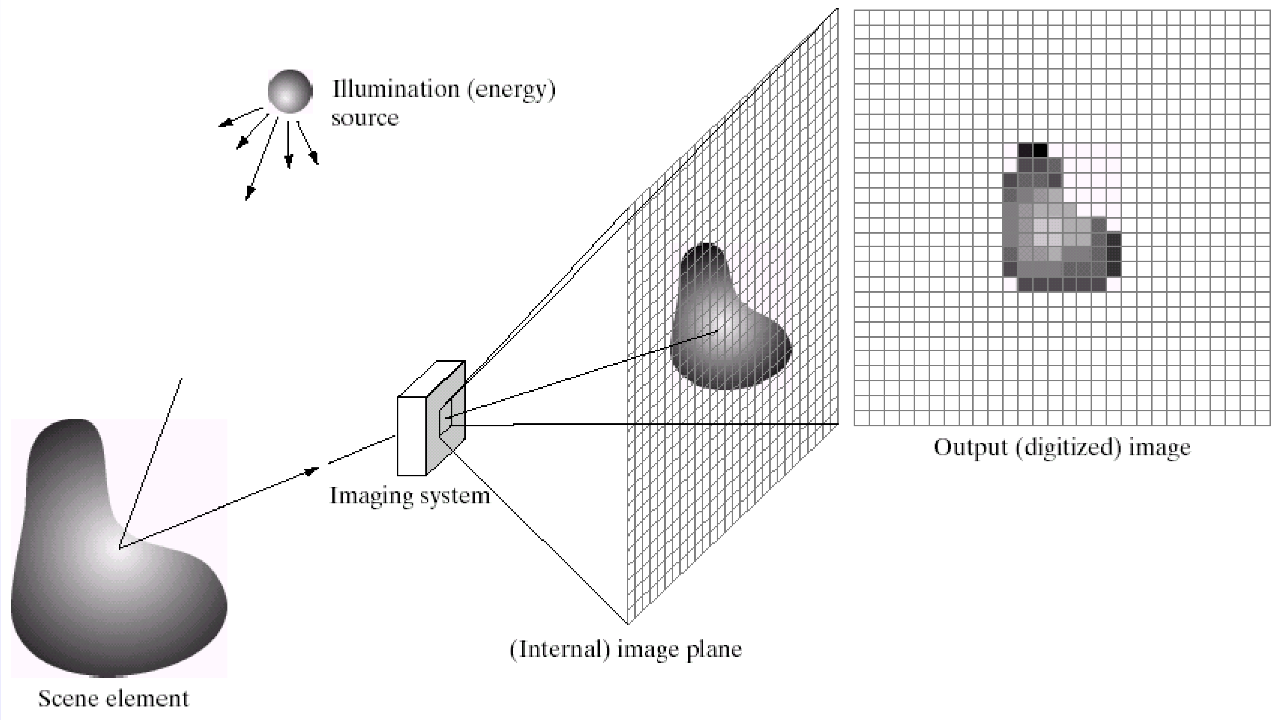
1. Image Fundamentals and Human Perception.
2. LSI Systems and Transformations
3. Image Enhancement
4. Image Restoration
5. Morphological Image Processing
6. Image Analysis I: Segmentation and Edge Detection
7. Image Analysis II: Hough Transform & Orientation Analysis
8. Understand Pattern Recognition and Decision Theory
9. Statistical Estimation and Machine Learning
10. Discriminant Functions and Classifiers.
11. Unsupervised Learning and Clustering
12. Data-driven Feature Extraction and Dimension Reduction
13. Connectionist Approaches, Neural Networks & Deep Learning

1. Image Fundamentals—Outline

- Image Formation and Representation
- Digital Image Representation
- Human Perception of Image and Color image
- Image Histogram
- Image Processing and Its Application

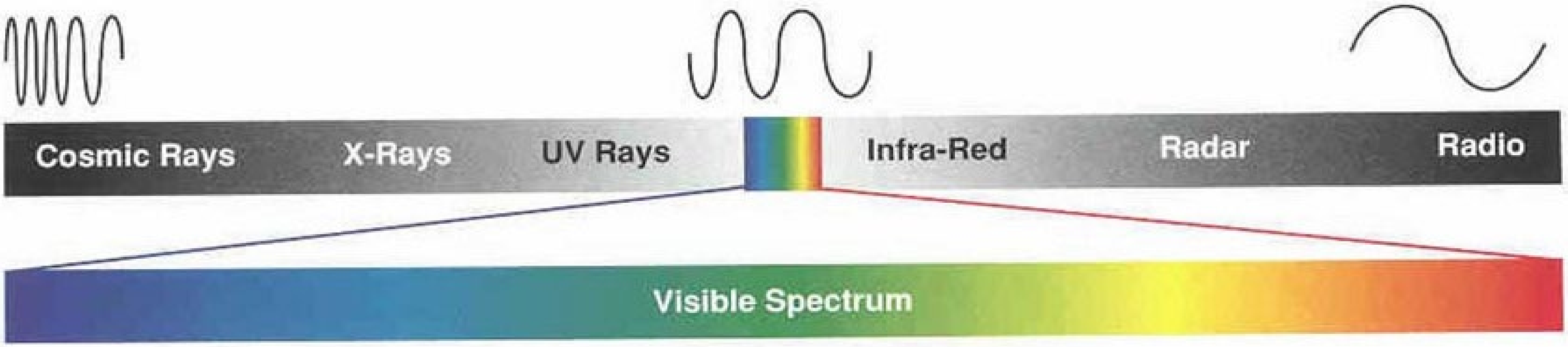
1. Image Fundamentals—image formation

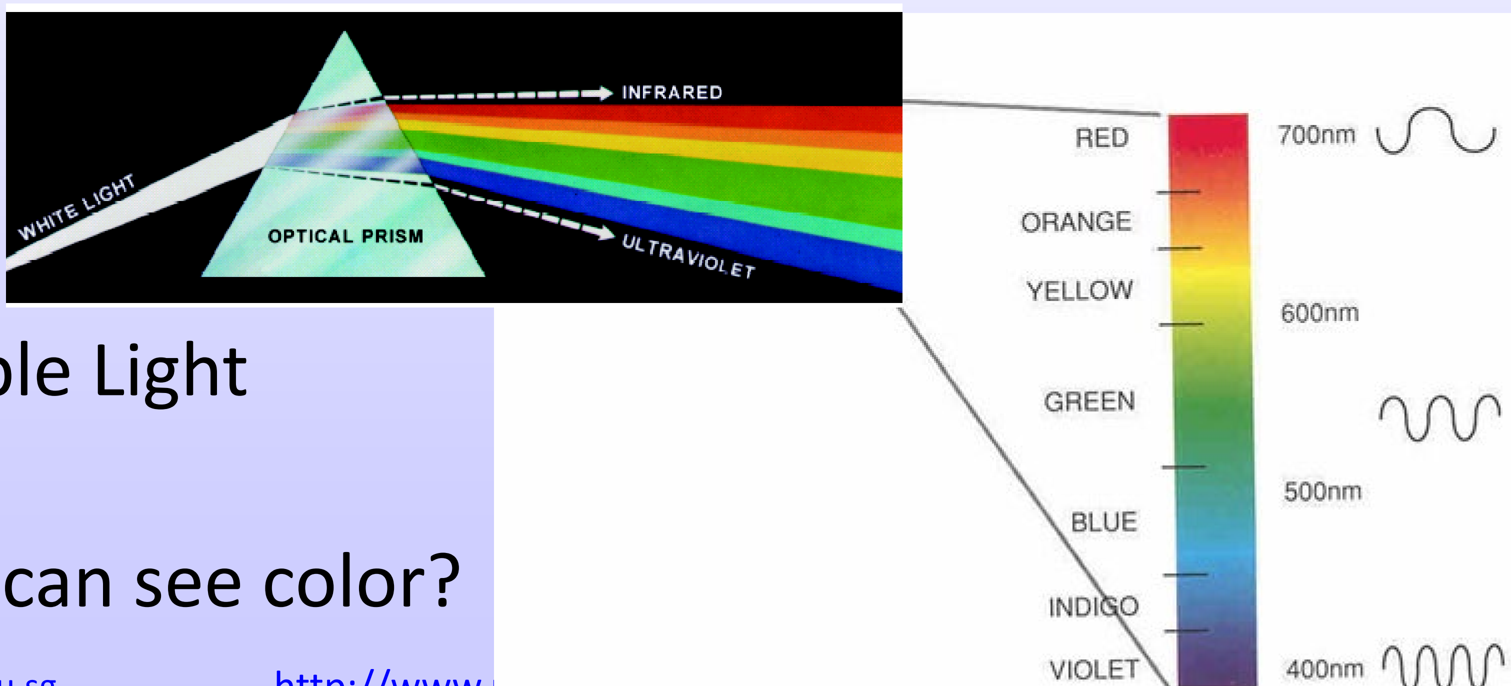
What is an image?



- Physically, an image is a two dimensional (2-D) projection of a three dimensional (3-D) scene, a visual representation, a vivid or graphic description of an object or scene.

1. Image Fundamentals—color image





The Visible Light

Why we can see color?

1. Image Fundamentals—Luminance and brightness

Light received from an object is

$$I(\lambda) = \rho(\lambda)L(\lambda)$$

where $\rho(\lambda)$ is the reflectivity or transmissivity of object, $L(\lambda)$ is the spectral energy distribution of the light source, λ is the wavelength in the visible spectrum, 350nm to 780 nm.

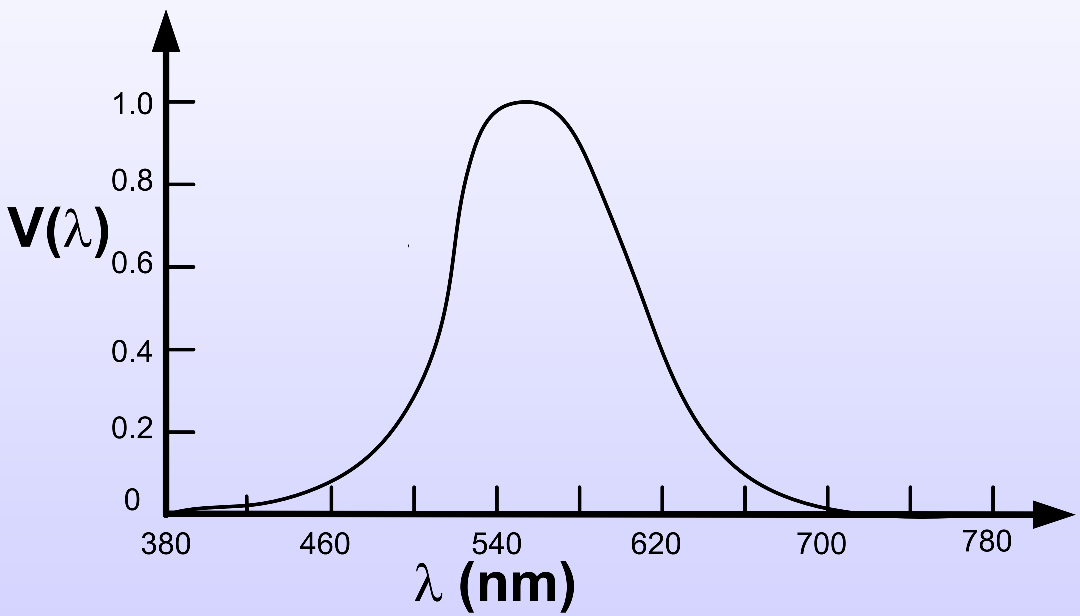
Luminance or intensity of a spatially distributed object with light distribution $I(x,y,\lambda)$ is defined as

$$f(x,y) = \int_0^\infty I(x,y,\lambda)V(\lambda)d\lambda$$

where $V(\lambda)$ is the relative spectral sensitivity function of the visual system

1. Image Fundamentals—Luminance and brightness

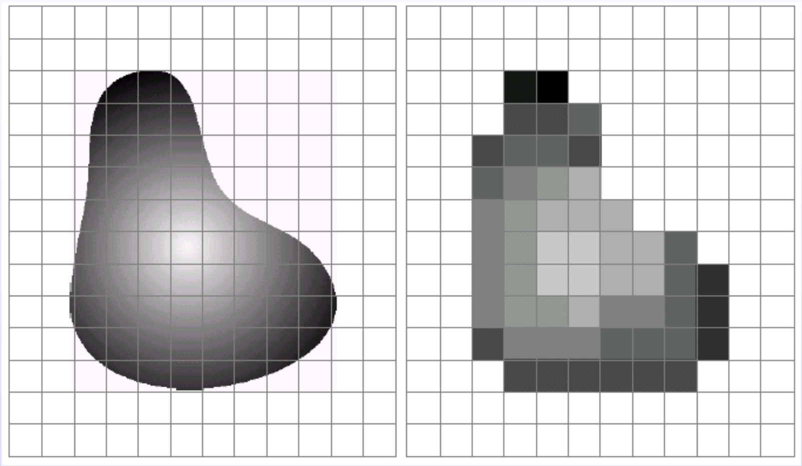
$V(\lambda)$ is a bell shaped curve



- The luminance of an object is independent of the luminance of its surrounding.

1. Image Fundamentals—representation of image

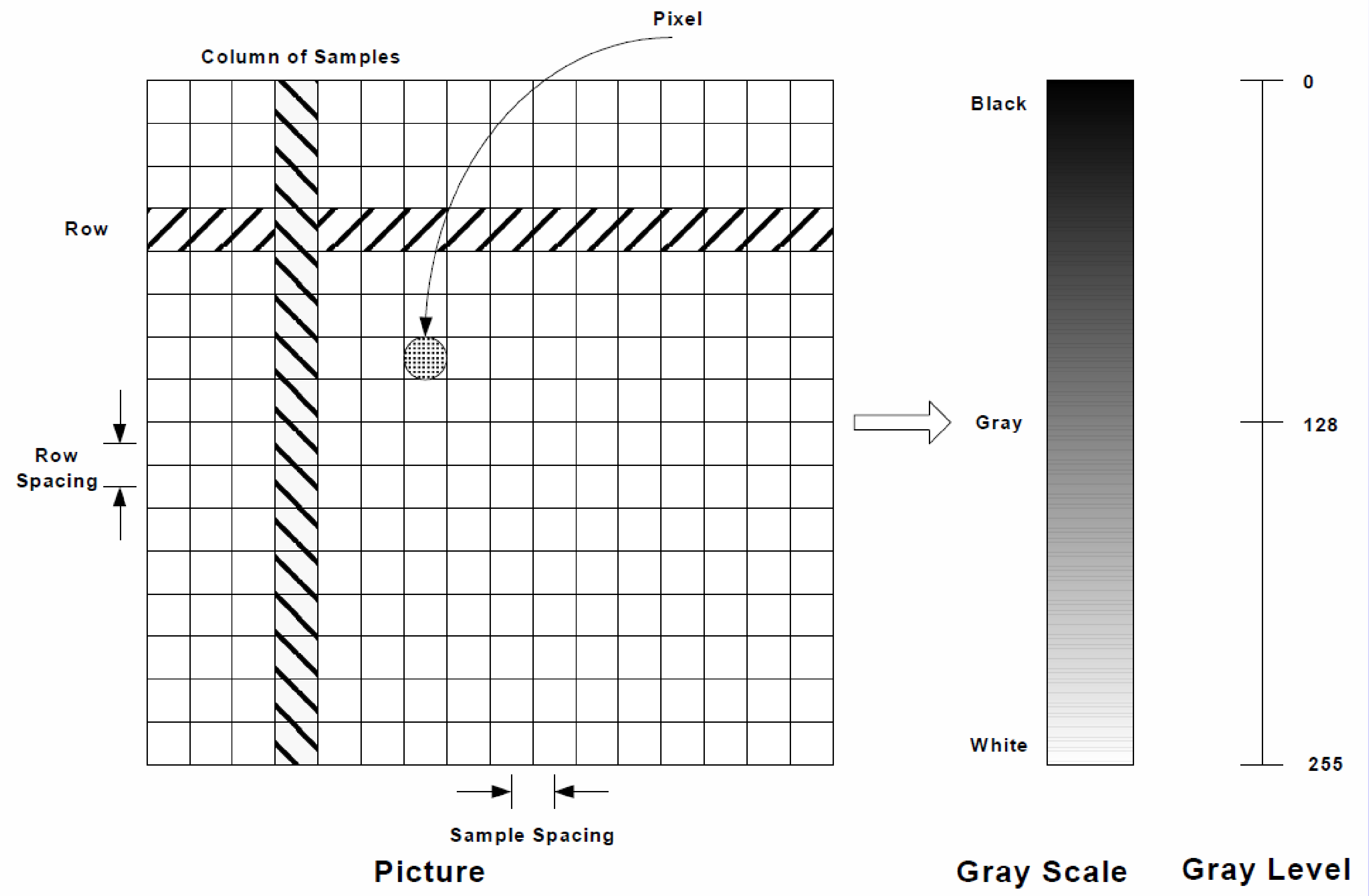
- Mathematically, an Image is a two-dimensional (2-D) function $f(x,y)$, a function of the two spatial coordinates.



- A digital image is a sampled, quantized version of a 2D light-intensity function generated by optical means.
- The function is usually sampled in an equally spaced rectangular grid pattern, with its amplitude quantized in equal intervals.
- A digital image is $f(x,y)$ where x, y and f are all finite and discrete quantities.

1. Image Fundamentals—image digitization

- Digitization of an image: Spatial sampling or discretization and Intensity or gray-level quantization



1. Image Fundamentals—representation of image

- Denote an image, a 2-D light-intensity function, as $f(x,y)$.
- The value or amplitude of f at spatial coordinates (x, y) indicates the intensity (brightness) of the image at that point.
- $f(x,y)$ must be digitized both spatially and in amplitude for computer processing.
- Digitization of spatial coordinates $f(x,y)$ is referred to as spatial sampling or discretization.
- Digitization of intensity amplitude $f(x,y)$ is referred to as intensity or gray-level quantization.

1. Image Fundamentals—representation of image

- The (spatial) resolution of a digital image refers to the size of the mxn array of which the image is sampled.

$$f(x,y) = e.g.: \sin[2\pi(u\sin(\phi)x + v\sin(\phi)y]$$

$$\begin{bmatrix} f(1,1) & f(1,2) & \cdots & f(1,n) \\ f(2,1) & f(2,2) & \cdots & f(2,n) \\ \vdots & \vdots & \ddots & \vdots \\ f(m,1) & f(m,2) & \cdots & f(m,n) \end{bmatrix} \text{or } \mathbf{F} = \begin{bmatrix} f_{11} & f_{12} & \cdots & f_{1n} \\ f_{21} & f_{22} & \cdots & f_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ f_{m1} & f_{m2} & \cdots & f_{mn} \end{bmatrix}$$

- The gray-level resolution of a digital image refers to the number of gray levels (intensities) $g=2^b$, where b is the number of bits per sample.

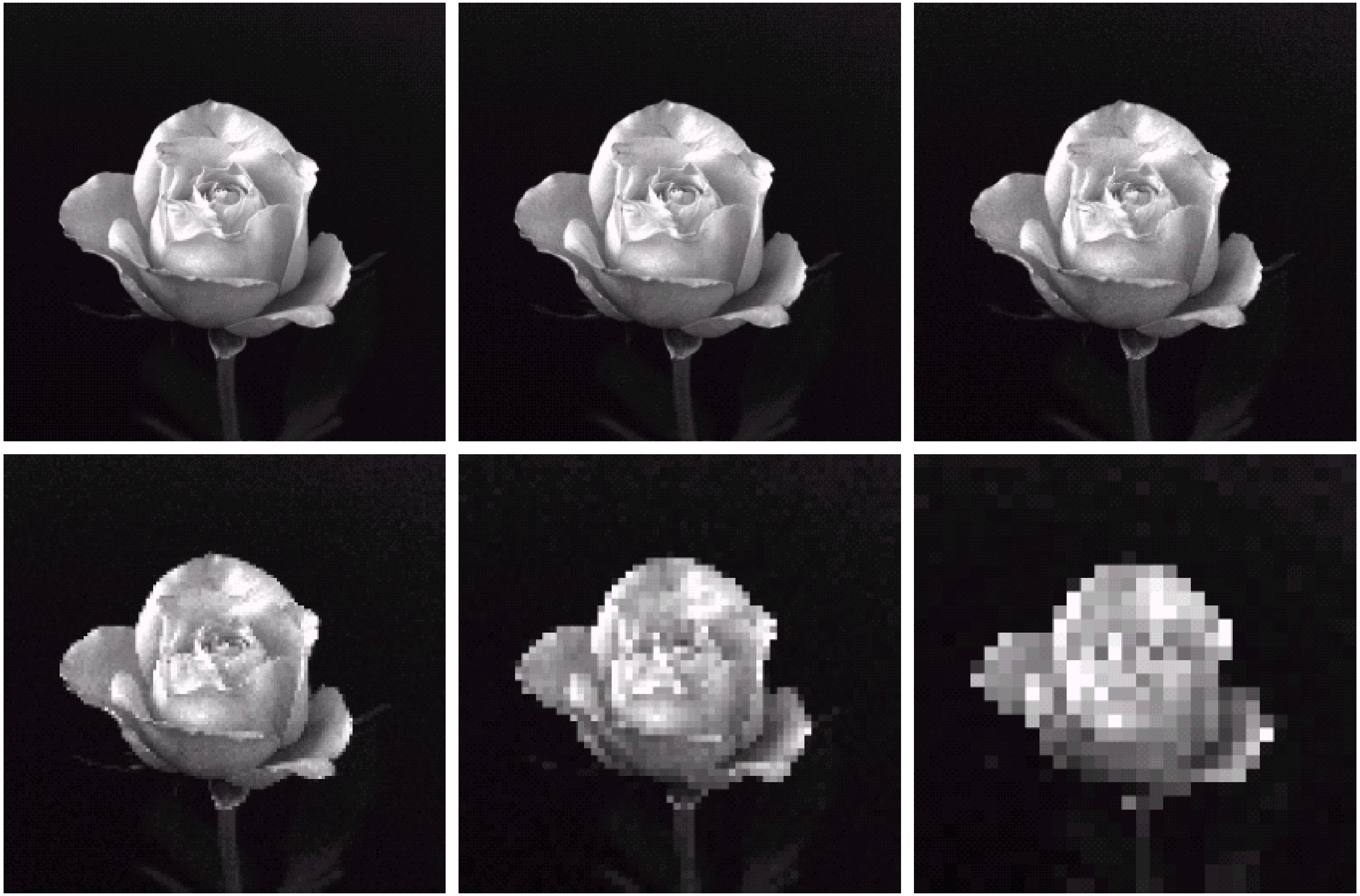
1. Image Fundamentals—spatial resolution

600 x 408 pixels

300 x 204 pixels

150 x 102 pixels

1. Image Fundamentals—spatial resolution



a	b	c
d	e	f

FIGURE 2.20 (a) 1024 × 1024, 8-bit image. (b) 512 × 512 image resampled into 1024 × 1024 pixels by row and column duplication. (c) through (f) 256 × 256, 128 × 128, 64 × 64, and 32 × 32 images resampled into 1024 × 1024 pixels.

exc

1. Image Fundamentals—gray-level resolution

256 levels

128 levels

64 levels

32 levels

16 levels

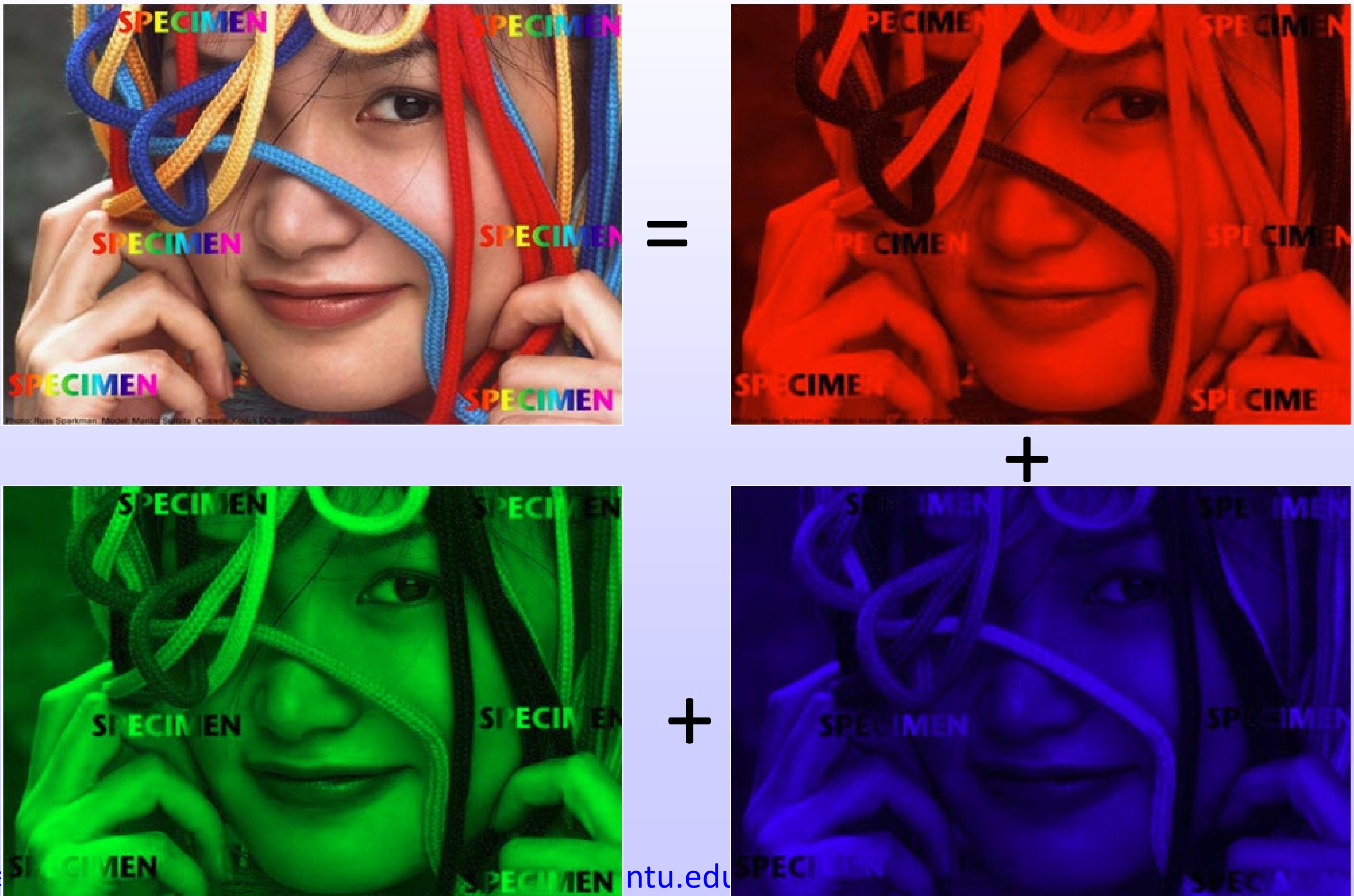
8 levels

4 levels

2 levels

1. Image Fundamentals—color image

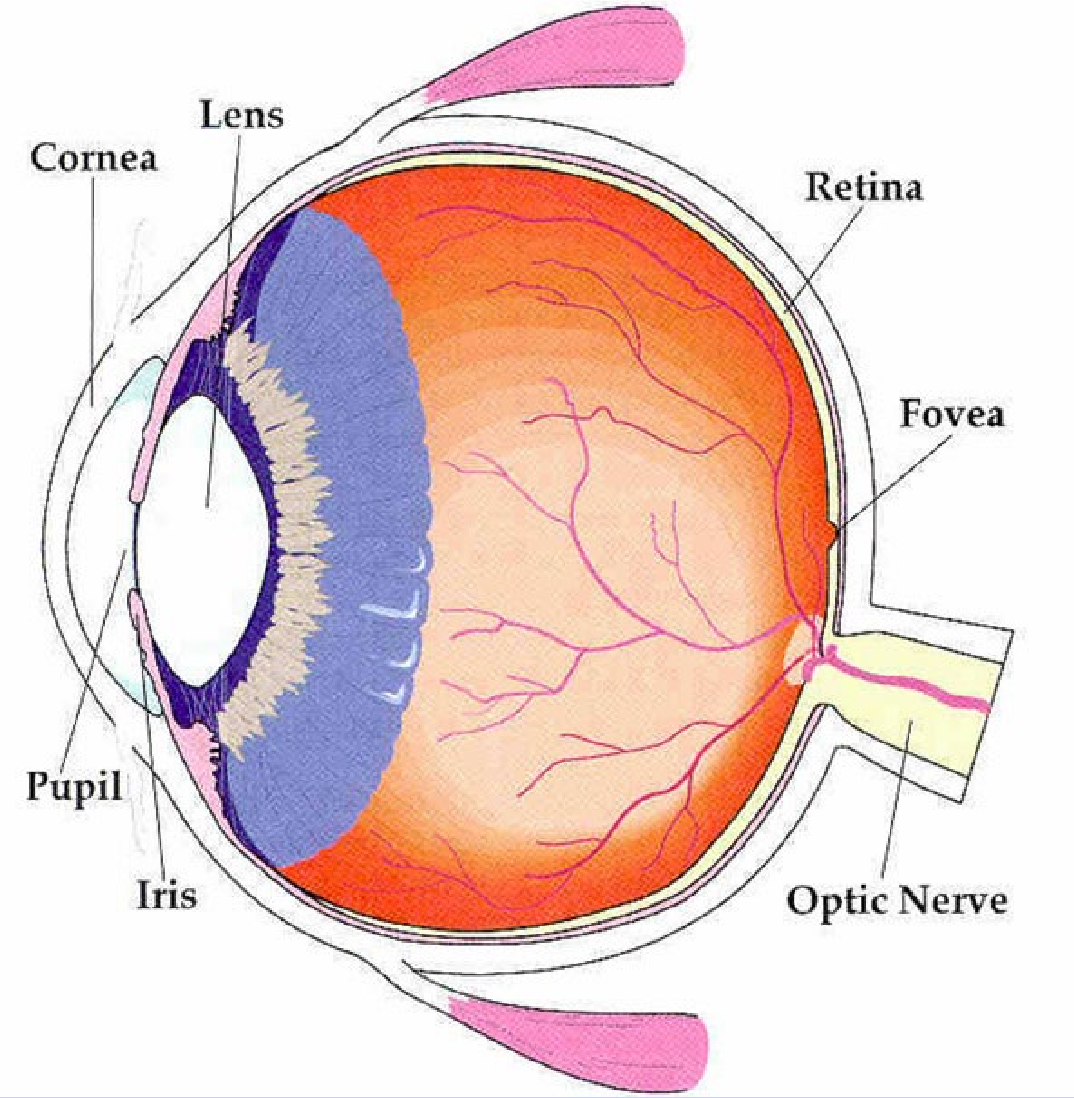
Three values per sample (pixel) are usually required for a color image.



1. Image Fundamentals—Human Perception

The Human eye

- Human eye has many cells shaped like cones to perceive light.
- There are three different type of cones help perceive colour.



1. Image Fundamentals—Human Perception

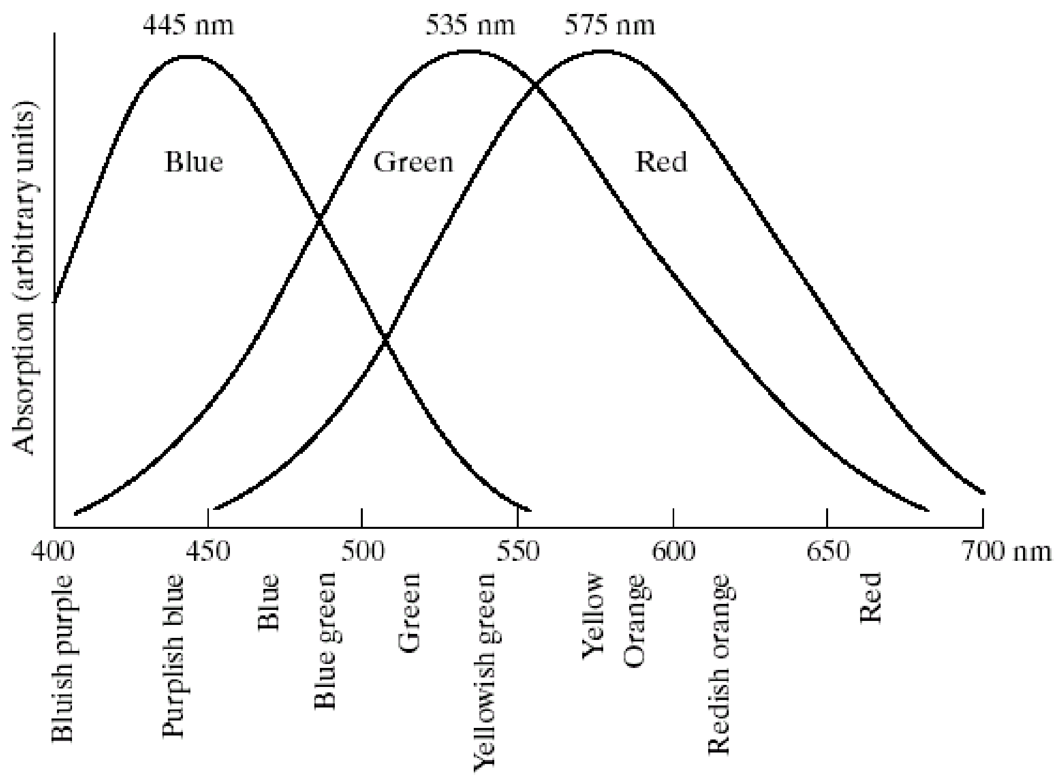
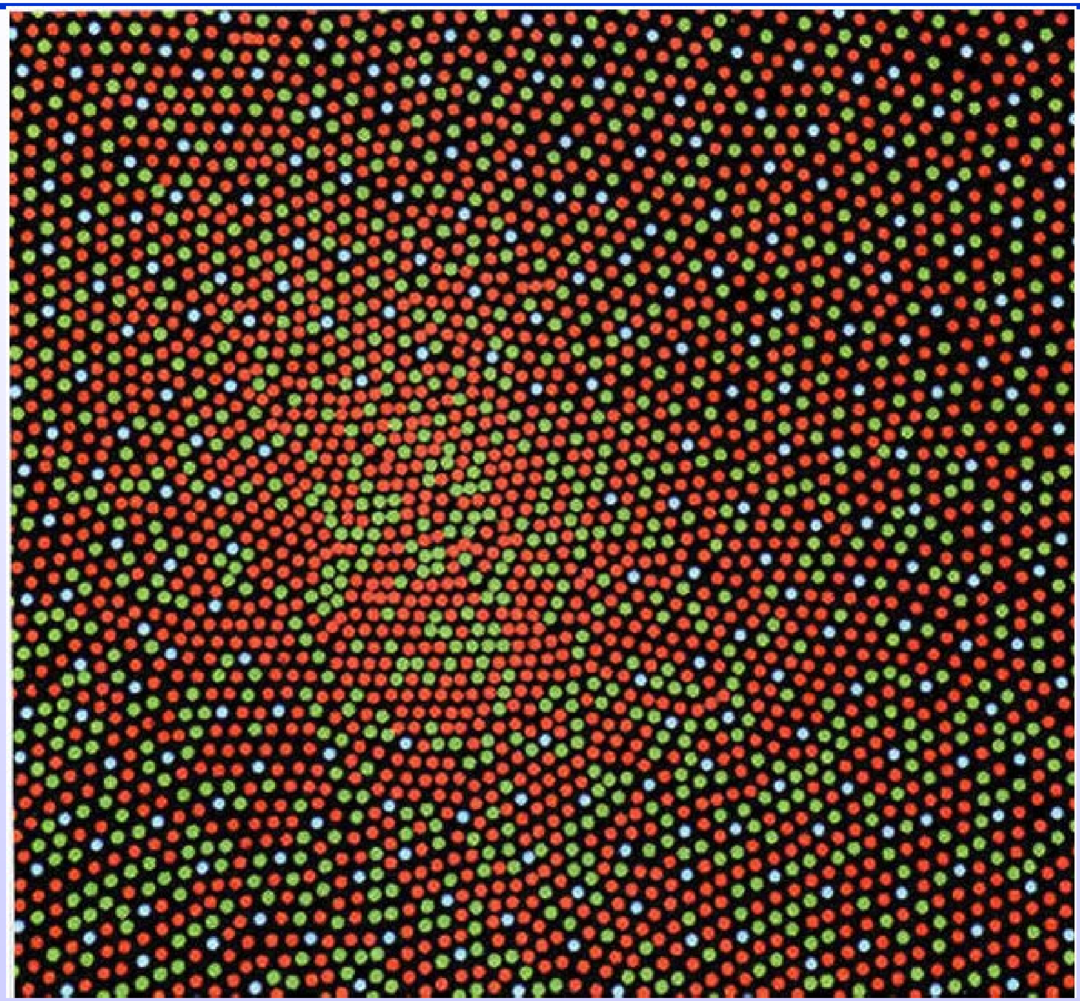


FIGURE 6.3 Absorption of light by the red, green, and blue cones in the human eye as a function of wavelength.

- Color is a function of wavelength (frequency)
- Color Primaries: Red(R), Green(G), Blue(B)

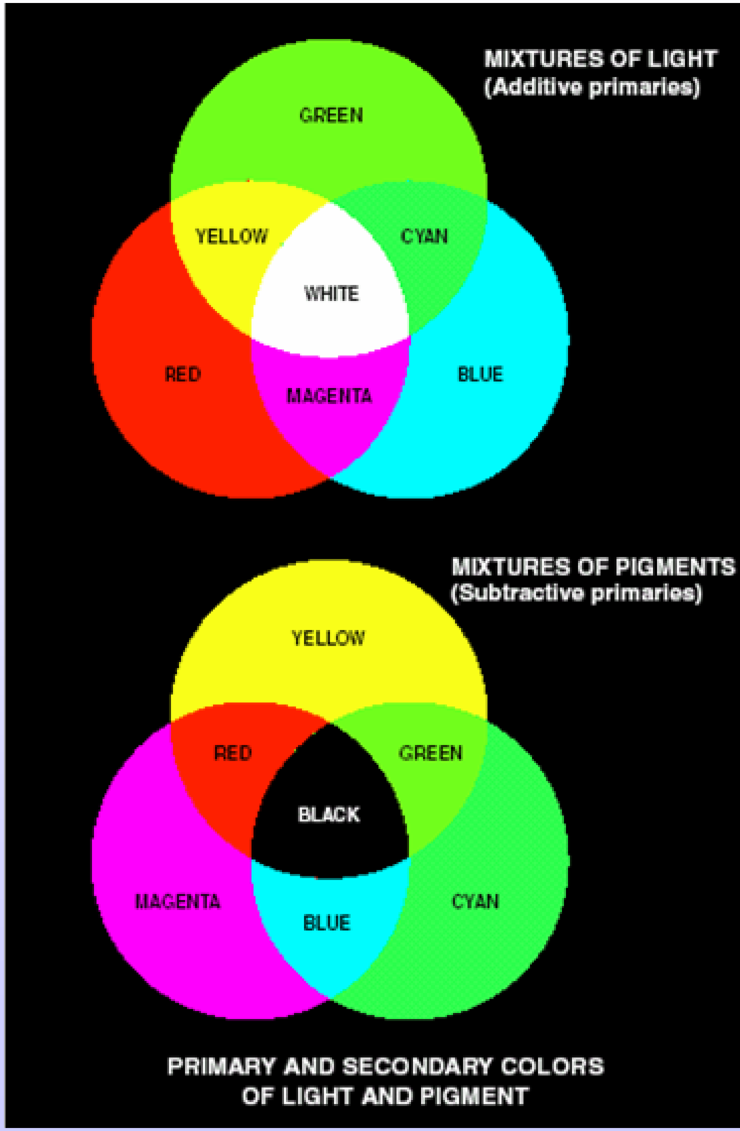
1. Image Fundamentals—Human Perception



Relative proportions of L (red), M (green), and S (blue) cones in the human retina.

1. Image Fundamentals—color space

- Additive primaries: Red(R), Green(G), Blue(B)
- Subtractive primaries: Cyan, Magenta, Yellow
- A color can be specified in terms of the amounts of three primaries required: $c = a\text{x}p1 + b\text{x}p2 + c\text{x}p3$, where ($p$1,$p$2,$p$3) is a particular set of primaries.
- A color space is a 3D space, defined to describe color in some standard way.

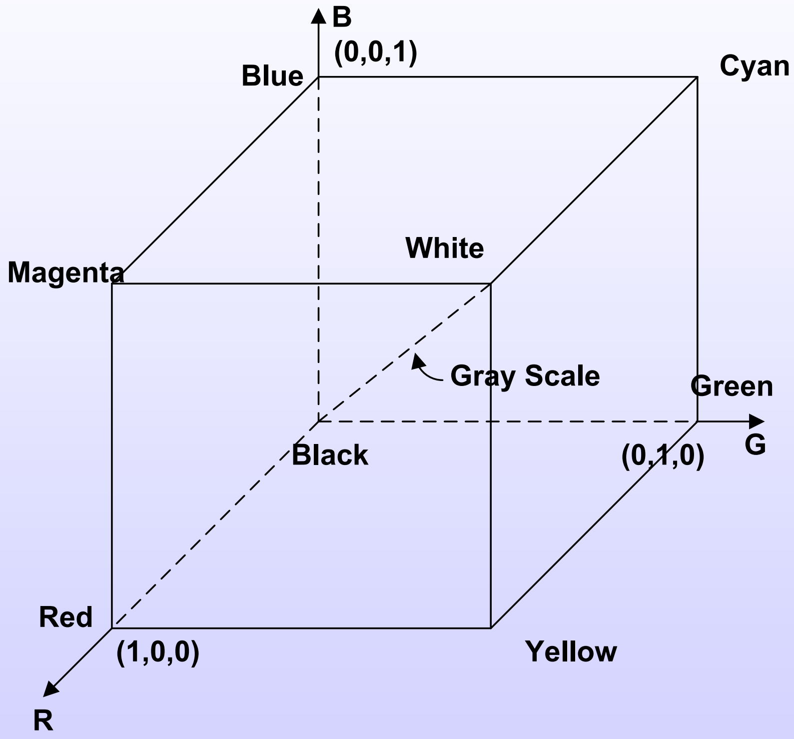


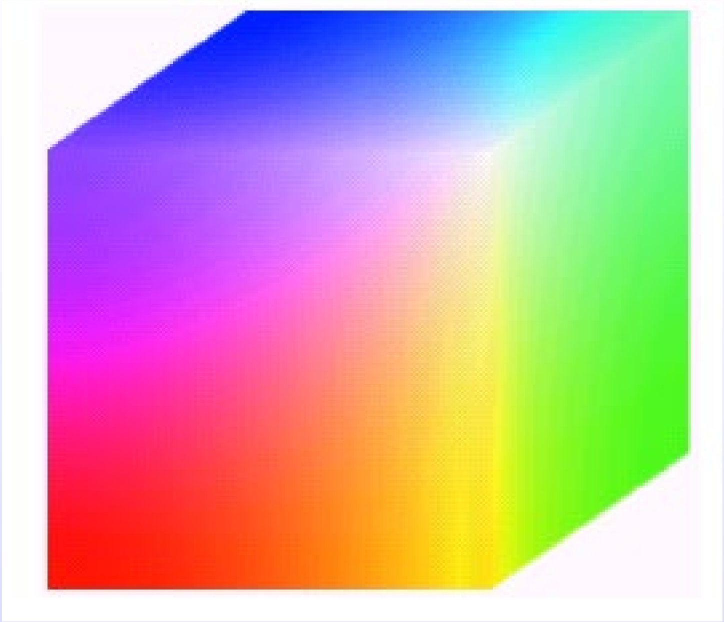
1. Image Fundamentals—color space

- Common color spaces:
  - RGB - hardware oriented, used for monitors, video cameras
  - rgb - Normalized RGB
  - CMY (Cyan-Magenta-Yellow) - used for color printer
  - YIQ (luminance, in-phase, quadrature.) - color TV broadcast
  - HSI (HSV) (Hue, saturation, intensive/value) - used for color manipulation
  - CIE-Luv, CIE-Lab (lightness, red-green, yellow-blue) - used for color differentiation.
  - sRGB – used for device independent digital image display.

1. Image Fundamentals—color space

- RGB color spaces:





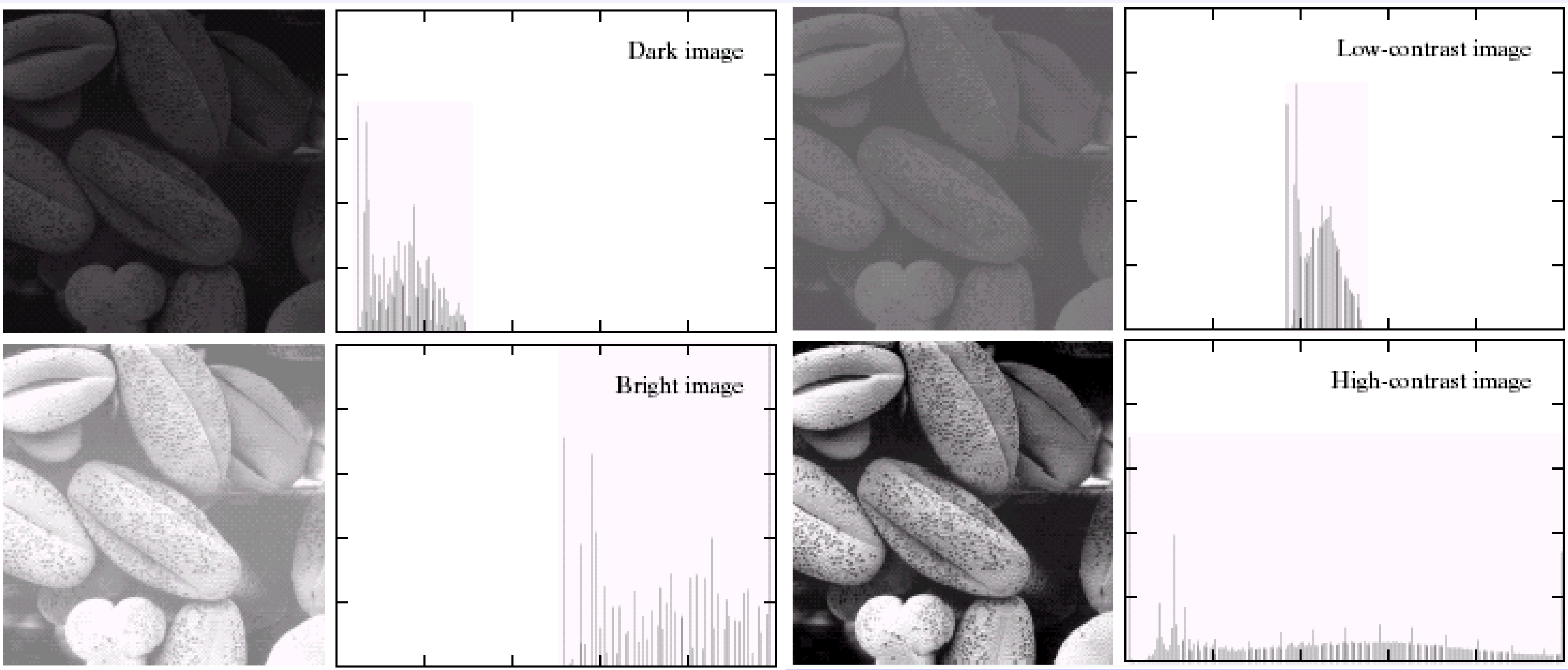
Z. Lu, X.D. Jiang and A. Kot, "A Color Channel Fusion Approach for Face Recognition," ***IEEE Signal Processing letters***, vol. 22, no. 11, pp. 1839 - 1843, Nov. 2015.

1. Image Fundamentals—image histogram

- The histogram of a digital image $f(x,y)$ with gray level range [0, L] is a discrete function
$$p_f(f) = \frac{n_f}{n}$$
- where f is the gray level, f = 0, 1, 2, …, L. n_f is the number of pixels with that gray level. n is the total number of pixels in the region of the image being processed.
- It is clear that the histogram $p_f(f)$ of a digital image is the frequency of occurrence of gray-level f in the image.
- Histogram shows the frequency distribution of gray-level f.
- Obviously, $$p_f(f) \geq 0, \quad \text{and} \quad \sum_{f=0}^{L} p_f(f) = 1$$
- If we treat the pixel gray-level of an image as a random number, the histogram $p_f(f)$ is an estimate of the probability of occurrence of gray-level f.

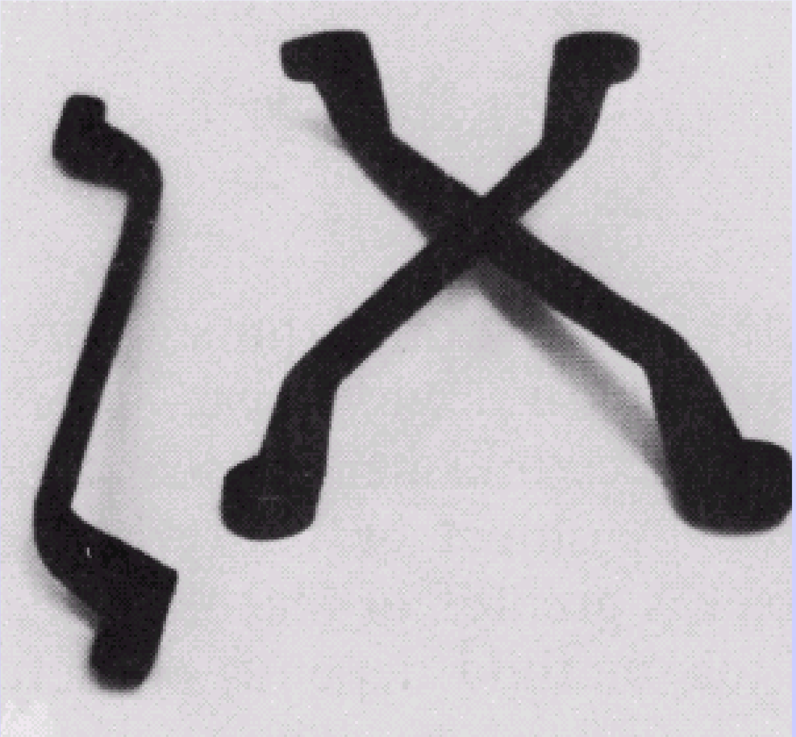
1. Image Fundamentals—image histogram

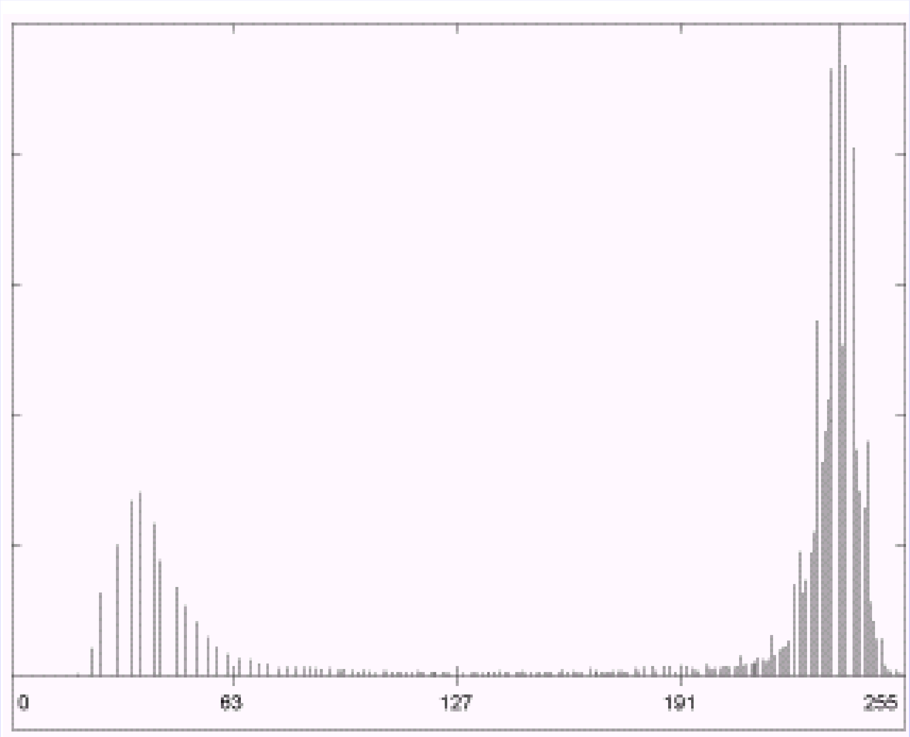
- Understand the histogram: The shape of image histogram provides many clues as to the characteristics of image. For example,
  - a narrowly distributed histogram indicates a low-contrast image



1. Image Fundamentals—image histogram

- Understand the histogram: The shape of image histogram provides many clues as to the characteristics of image. For example,
  - A bimodal histogram suggests that the image contains an object with a narrow amplitude range against a background of differing amplitude.





1. Image Fundamentals—image histogram

- Recent Journal papers exploring histogram features:

J. Ren, X. Jiang and J. Yuan, "LBP Encoding Schemes Jointly Utilizing the Information of Current Bit and Other LBP Bits," ***IEEE Signal Processing letters***, vol. 22, no. 12, pp. 2373 - 2377, Dec. 2015.

J. Ren, X. Jiang and J. Yuan, "A Chi-Squared-Transformed Subspace of LBP Histogram for Visual Recognition," ***IEEE Trans. Image Processing***, vol. 24, no. 6, pp. 1893-1904, June, 2015.

J. Ren, X. Jiang and J. Yuan, "Learning LBP Structure by Maximizing the Conditional Mutual Information," ***Pattern Recognition***, vol. 48, no. 10, pp. 3180 - 3190, Oct. 2015.

A. Satpathy, X. Jiang and H. Eng, "LBP Based Edge-Texture Features for Object Recognition," ***IEEE Trans. Image Processing***, vol. 23, no. 5, pp. 1953-1964, May, 2014.

J. Ren, X. Jiang, J. Yuan and W. Gang, "Optimizing LBP Structure for Visual Recognition Using Binary Quadratic Programming," ***IEEE Signal Processing letters***, vol. 21, no. 11, pp. 1346-1350, Nov. 2014.

A. Satpathy, X. Jiang and H. Eng, "Human Detection by Quadratic Classification on Subspace of Extended Histogram of Gradients," ***IEEE Trans. Image Processing***, vol. 23, no. 1, pp. 287-297, Jan, 2014.

J. Ren, X. Jiang and J. Yuan, "Noise-Resistant Local Binary Pattern with an Embedded Error-Correction Mechanism," ***IEEE Trans. Image Processing***, vol. 22, no. 10, pp. 4049-4060, Oct, 2013.

1. Image Fundamentals—what is image processing?

- An digital image is a two dimensional numerical representation (a 2-D function $f(x,y)$ or a mxn matrix) of a 3D scene or an object.

What is digital image processing?

- Digital image processing is a series of machine or computer operations leading to some desired results.
- The operations could be, should be, and are desired to be described by mathematics.
- Digital image processing, starting with an image or a set of images, produces a modified version of the image(s) or extract more "meaningful" information (features) from the image(s) or understand (recognize) the meaning of the image content.

1. Image Fundamentals—Why need image processing?

- **Why need image processing?**
  - Visualization :
    - Contrast enhancement, noise removal, visual quality improvement, pseudo colouring
  - Image understanding
    - Extraction of image properties such as colour, shape, texture, edges, lines, curves, corners.
  - Automated Guided vehicle
    - Identifying road, vehicle, pedestrian, traffic signs
  - Visual servicing
    - Automated robot control
  - Security
    - Intrusion detection, biometrics
  - Information retrieval
    - Image content based search

1. Image Fundamentals—image processing examples

- Contrast Enhancement

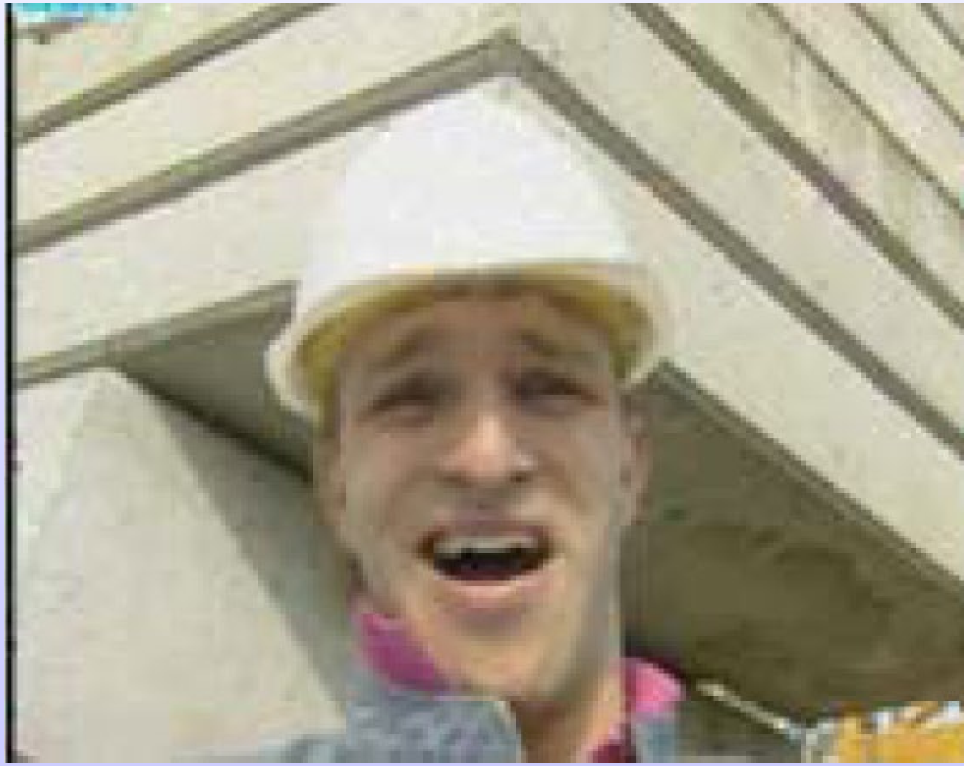
1. Image Fundamentals—image processing examples

- Deblurring

1. Image Fundamentals—image processing examples

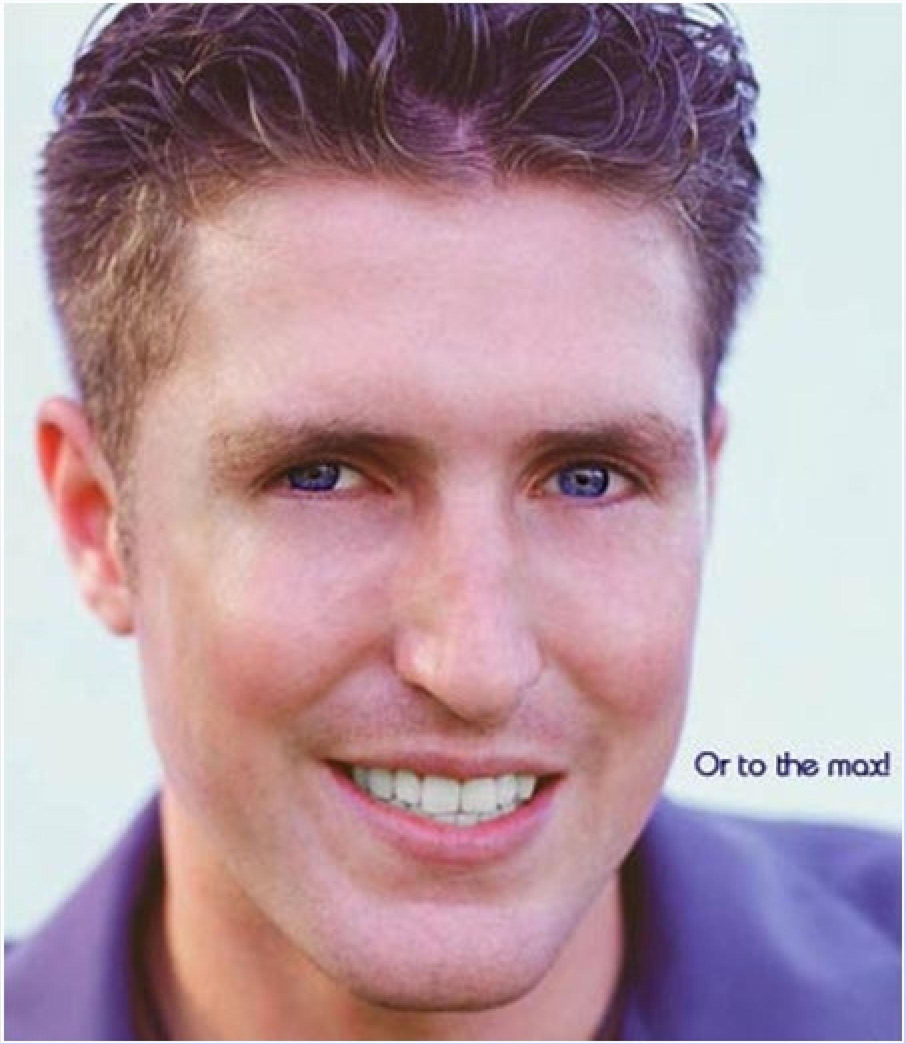
- Denoising: noise attenuation

Noise corrupted image and the image processed by DIP



1. Image Fundamentals—image processing examples

- Beautifying

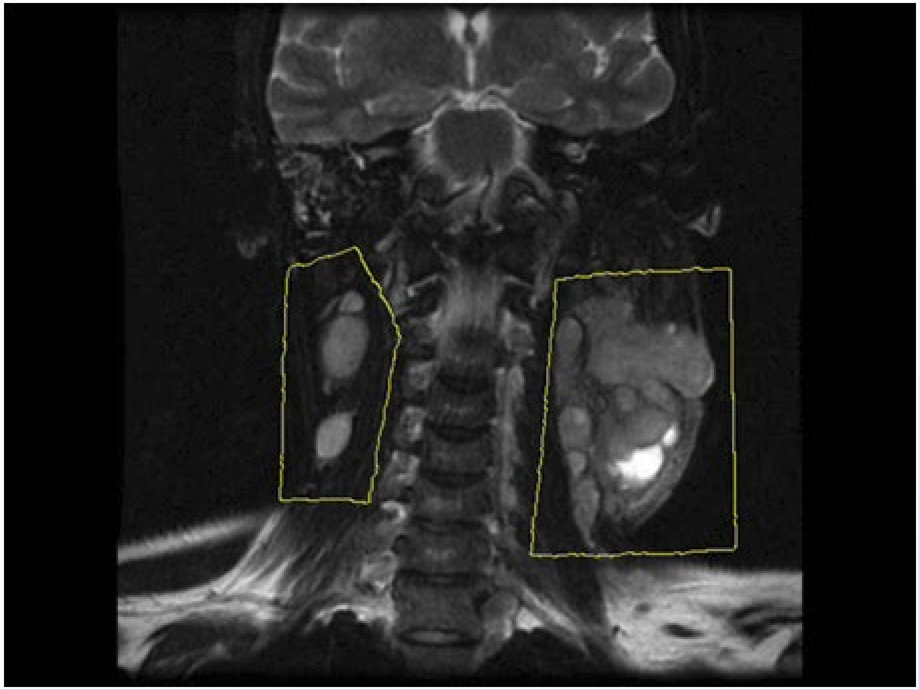


1. Image Fundamentals—image processing examples

- Restoration and Retouching

1. Image Fundamentals—image processing examples

- Segmentation



1. Image Fundamentals—image processing examples

Digital Watermarking

- **Traditional Watermark (bank notes)**

Watermark would appear when placed in the presence of ultra-violet light

- **Digital Watermark (digital images)**

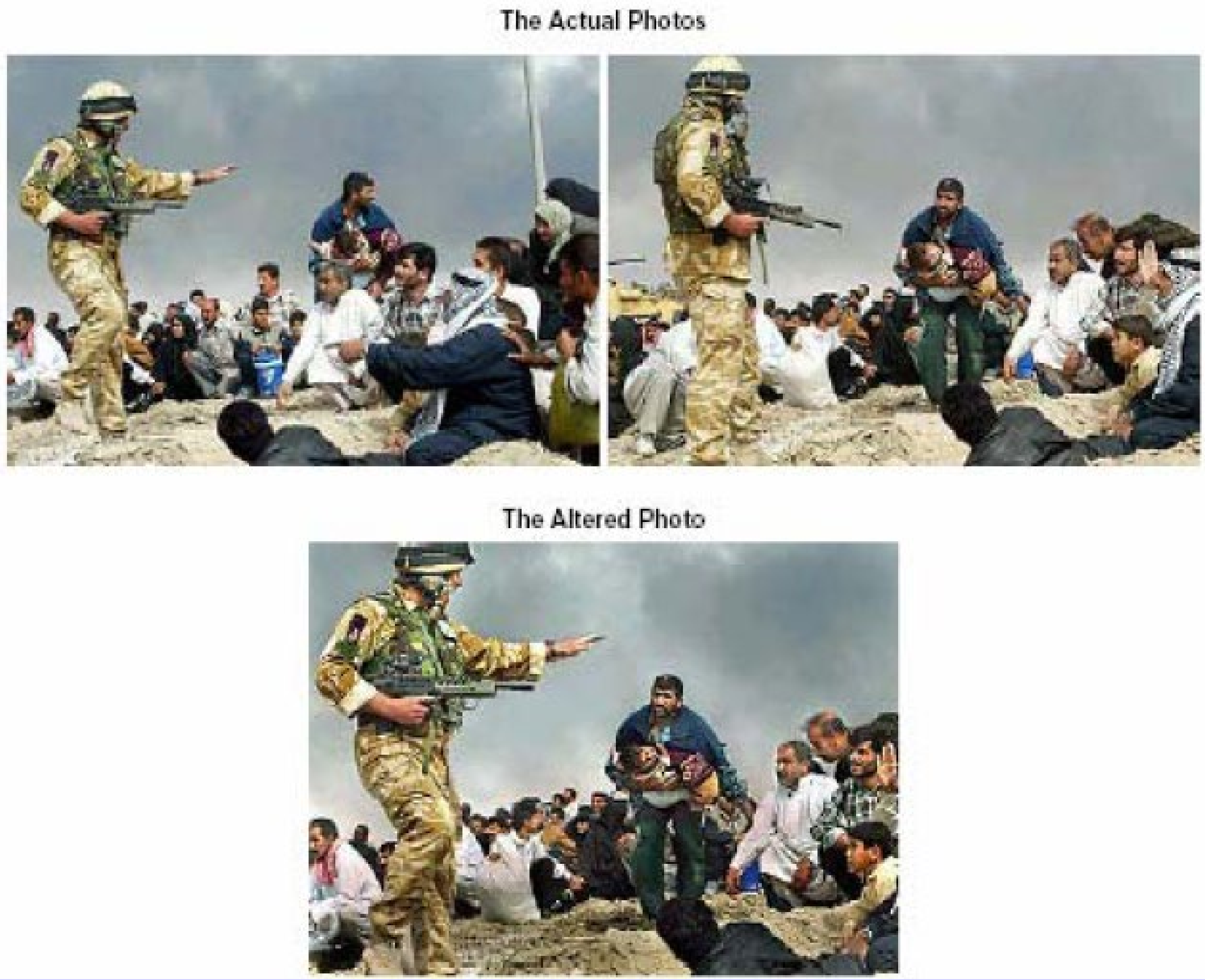
©

Digital watermark retrieved through an algorithm

1. Image Fundamentals—image processing examples

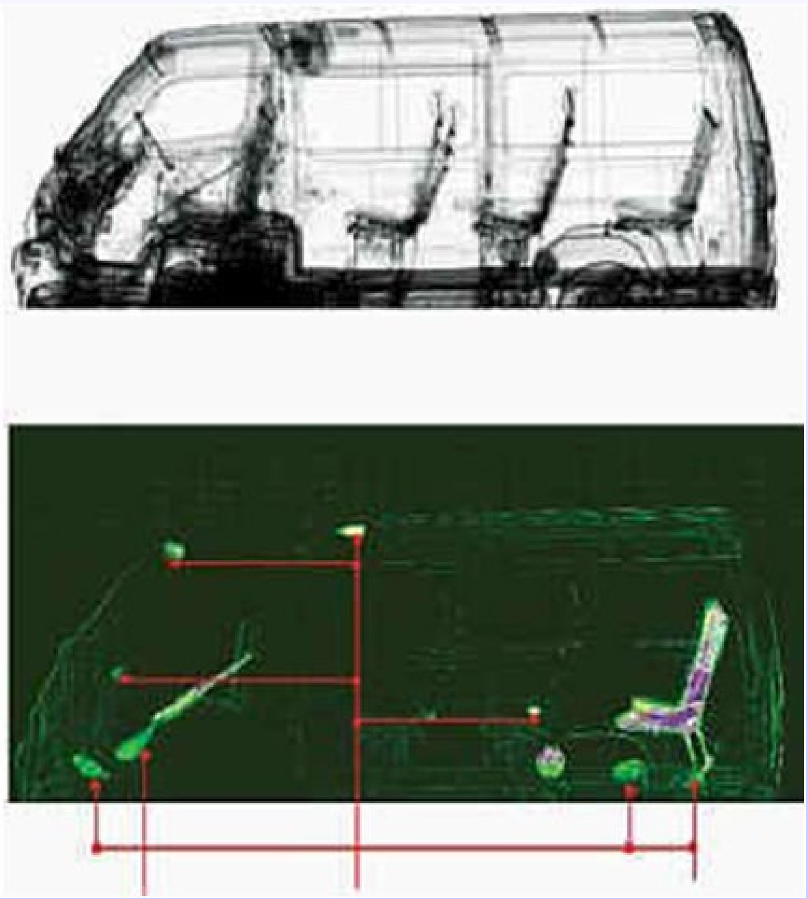
Digital content tampering detection

The photograph is a composite created by ex-composite LA Times photographer Brian Walski.
He was dismissed when the photograph was found be altered.



1. Image Fundamentals—image processing examples

- Objection Detection

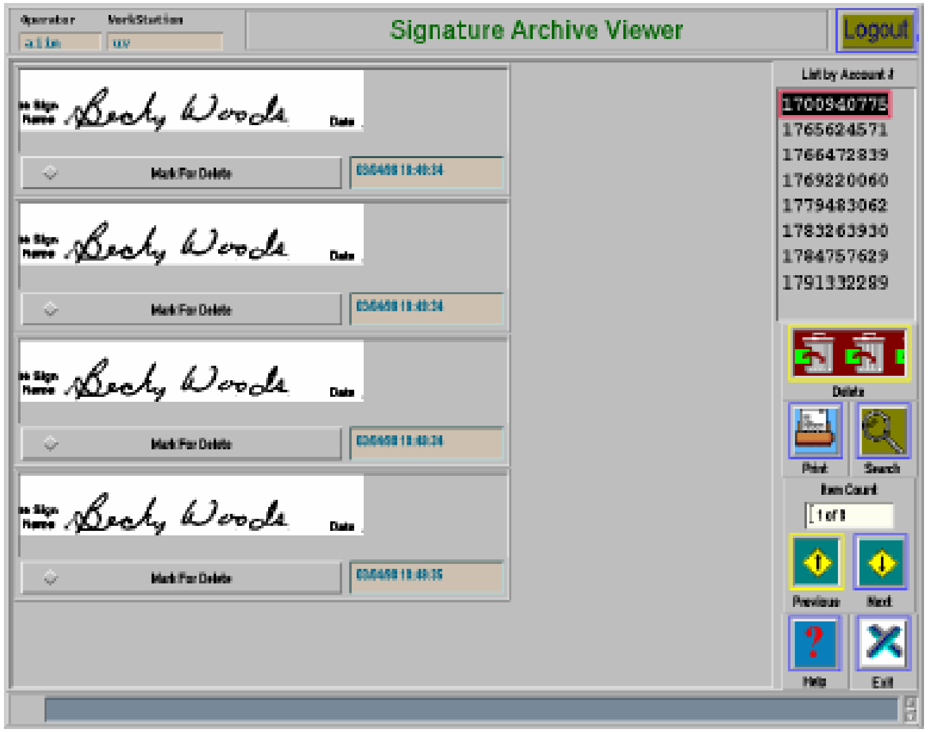


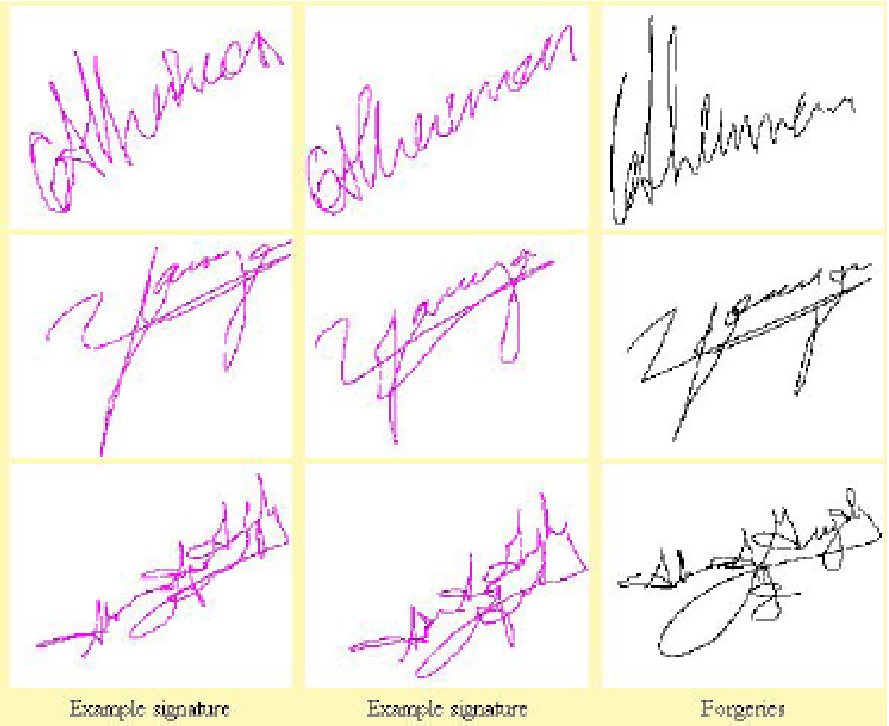
1. Image Fundamentals—image processing examples

- Face Detection

1. Image Fundamentals—image processing examples

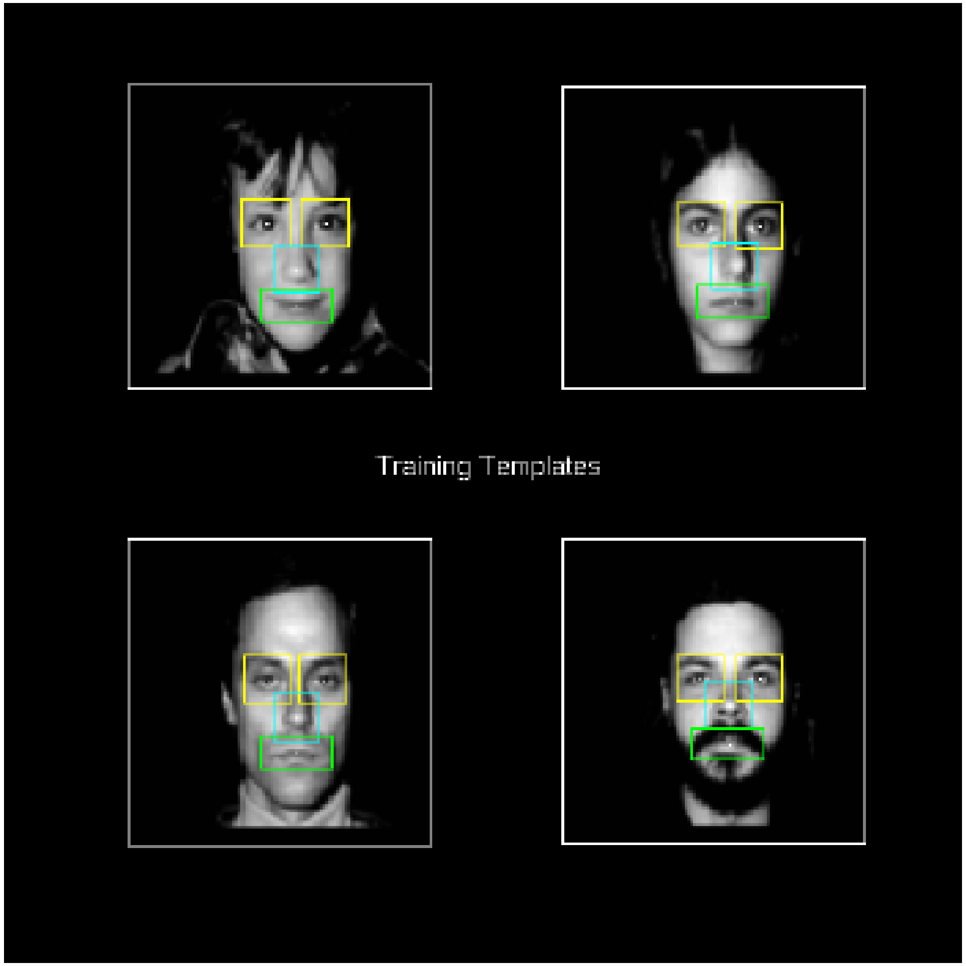
- Automatic signature verification and identification

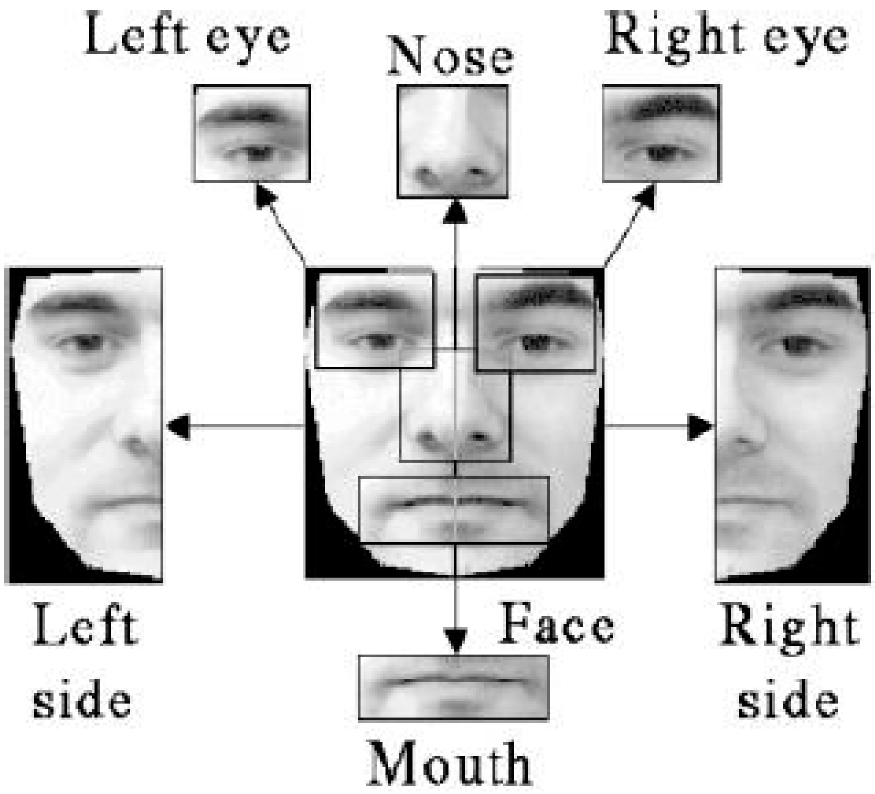




1. Image Fundamentals—image processing examples

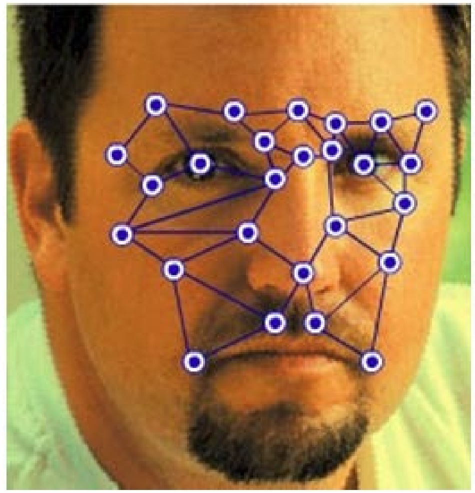
Are these feature effective for Automatic Face Recognition?

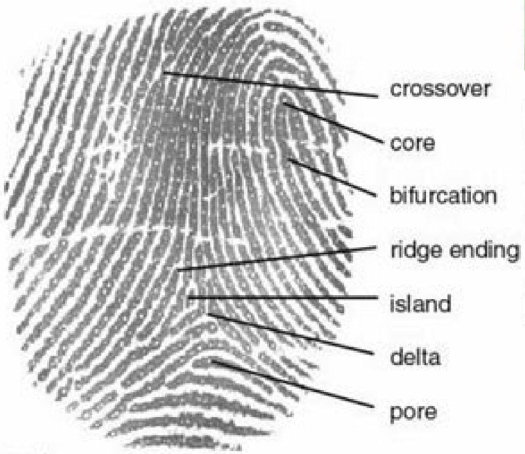


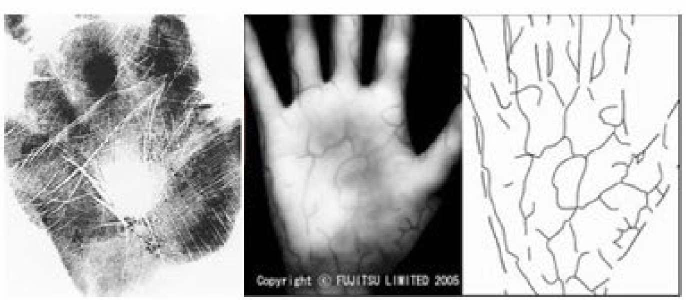


1. Image Fundamentals—image processing examples

Biometrics



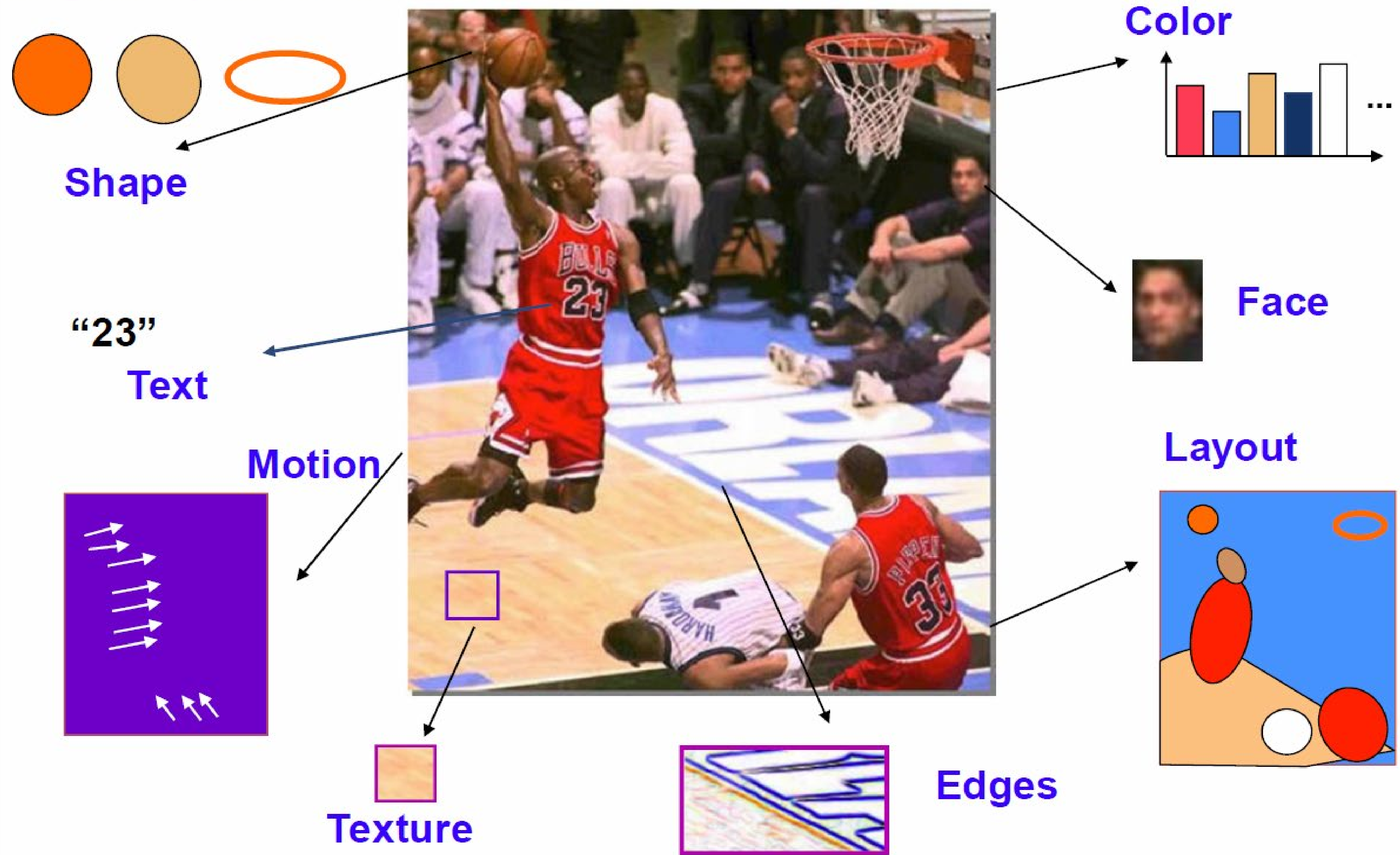




1. Image Fundamentals—image processing examples

1. Image Fundamentals—image processing examples

Image Analysis and Understanding



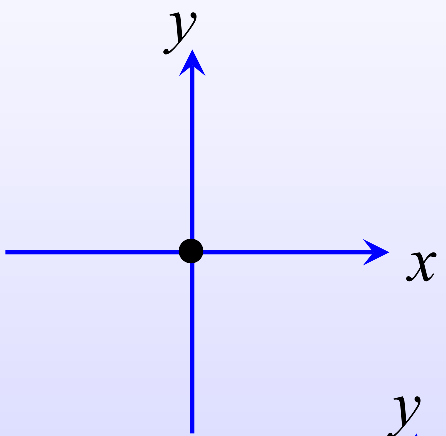
2. LSI Systems & Transforms—Outline

- Image Decomposition and Linear Shift Invariant Image Processing System
- Two-dimensional Convolution and its Properties
- Two-dimensional Fourier Transform and its Properties
- Two-dimensional Cosine transform.
- Image Sampling and Quantization

2. LSI Systems & Transforms—basic image element

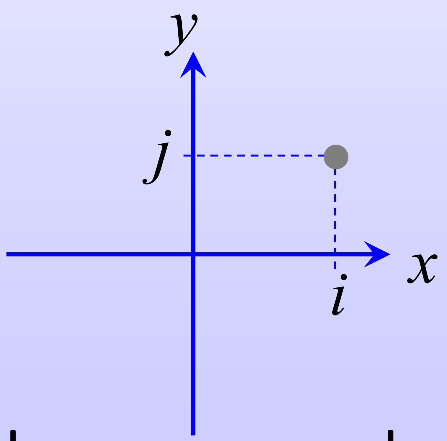
- A digital image can be represented by a 2-D function with two integer arguments, such as $f(x,y)$ where x and y are integers
- The basic element image is the impulse

$$\delta(x,y)=\begin{cases}1, & x=y=0\\ 0, & \text{otherwise}\end{cases}$$



- Shifted and scaled the impulse

$$f(x,y)=c\delta(x-i,y-j)=\begin{cases}c, & x=i, y=j\\ 0, & \text{otherwise}\end{cases}$$



- It describes any arbitrary pixel with all other pixels zero gray vale.

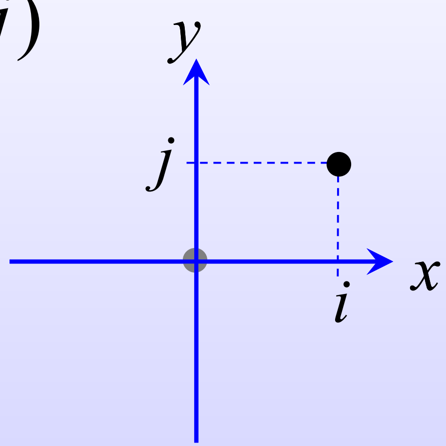
2. LSI Systems & Transforms—image decomposition

- Any image $f(x,y)$ then can be represented by the sum of a number of shifted and scaled impulses

$$f(x,y)=\sum_{j=-\infty}^{\infty}\sum_{i=-\infty}^{\infty}f(i,j)\delta(x-i,y-j)$$

Or

$$f(x,y)=\sum_{j=-n}^{n}\sum_{i=-m}^{m}f(i,j)\delta(x-i,y-j)$$



- For example, a constant gray level 215 square of size 11X11 centered at (0,0) is:

$$f(x,y)=\sum_{j=-5}^{5}\sum_{i=-5}^{5}215\delta(x-i,y-j)$$

2. LSI Systems & Transforms—2-D convolution

- A processing system relates any input image $f(x,y)$ to a unique output image $g(x,y)$, given by

$$g(x,y)=T\{f(x,y)\}=T\left\{\sum_{j=-\infty}^{\infty}\sum_{i=-\infty}^{\infty}f(i,j)\delta(x-i,y-j)\right\}$$

- If the processing system is linear, then

$$g(x,y)=T\{f(x,y)\}=\sum_{j=-\infty}^{\infty}\sum_{i=-\infty}^{\infty}f(i,j)T\{\delta(x-i,y-j)\}$$

- If define the output image of the input impulse image as impulse response of the system,

$h(x,y)\triangleq T\{\delta(x,y)\}$. Then for shift invariant system:

$$T\{\delta(x-i,y-j)\}=h(x-i,y-j)$$

2. LSI Systems & Transforms—2-D convolution

- Therefore, given an input image $f(x,y)$, a linear and shift-invariant (LSI) image processing system T produces the output image $g(x,y)$ by

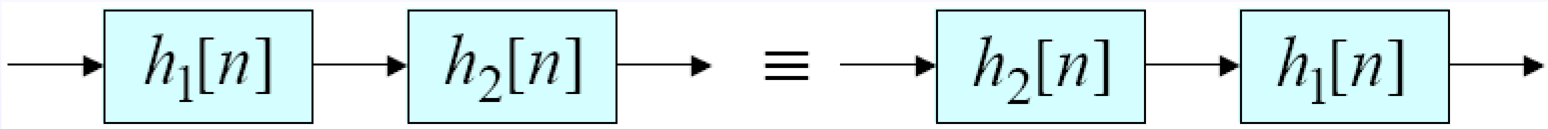
$$g(x,y) = T\{f(x,y)\} = \sum_{j=-\infty}^{\infty} \sum_{i=-\infty}^{\infty} f(i,j)h(x-i,y-j)$$

$$\triangleq f(x,y) * h(x,y) = \sum_{j=-\infty}^{\infty} \sum_{i=-\infty}^{\infty} h(i,j)f(x-i,y-j)$$

- A LSI system is completely characterized by its impulse response $h(x,y)$. $*$ is the convolution operator.
- For any LSI image processing system, the output image equals to the input image convolving with the impulse response of the system.

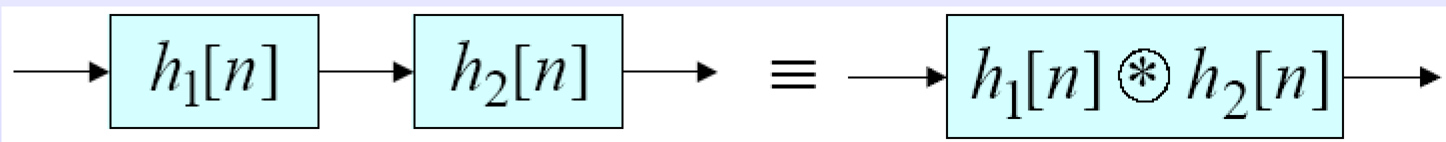
2. LSI Systems & Transforms—convolution property

- **Commutative:** $f(x,y)*h(x,y)=h(x,y)*f(x,y)$



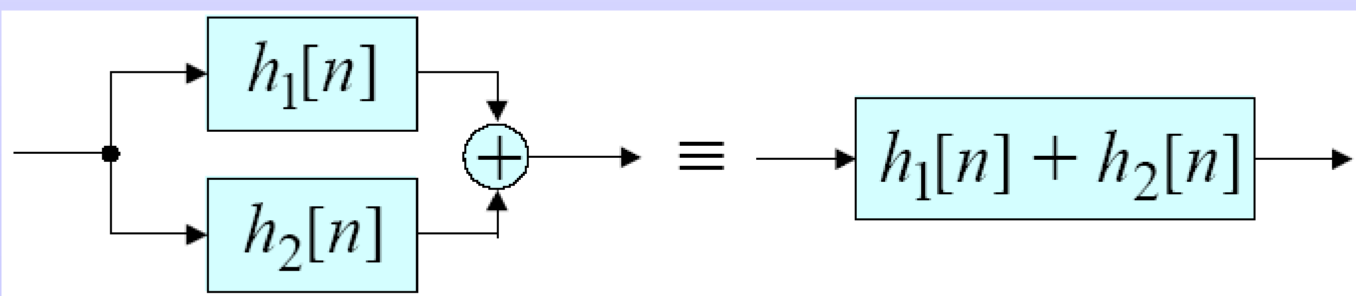
- **Associative:**

$$f(x,y)*\big(h_1(x,y)*h_2(x,y)\big)=\big(f(x,y)*h_1(x,y)\big)*h_2(x,y)$$



- **Distributive:**

$$f(x,y)*\big(h_1(x,y)+h_2(x,y)\big)=f(x,y)*h_1(x,y)+f(x,y)*h_2(x,y)$$



2. LSI Systems & Transforms—what is $h(x,y)$?

Understanding the impulse response:

- Impulse response $h(x,y)$ of an image processing system is the output image when a impulse image is inputted to the system.
- Therefore, the impulse response $h(x,y)$ is also an image, often called spatial representation of a filter or filter mask or filter coefficients or filter parameters.
- Although the impulse response $h(x,y)$ is basically an image, to speed up the image process, it is often a small size of image with size such as 3X3, 5X5, …11X11, comparing with a normal image of size 256X256.

$$g(x,y) = f(x,y) * h(x,y) = \sum_{j=-\infty}^{\infty} \sum_{i=-\infty}^{\infty} h(i,j) f(x-i, y-j)$$

$$= \sum_{j=-3}^{3} \sum_{i=-3}^{3} h(i,j) f(x-i, y-j), \quad \text{if } h(x,y) \neq 0 \ \text{ only when } -3 < x, y < 3$$

2. LSI Systems & Transforms—understand by example

- Given an input image $f(x,y)$, you want to suppress the pixel random noise or smooth the image so that the image looks "soft". So you do some local average that a pixel in the output image is produced by sum up the corresponding pixel and its 4 neighbor pixels in the input image.
- What is the mathematical expression of this very simple process?

$$g(x,y) = f(x,y) + f(x-1,y) + f(x+1,y) + f(x,y-1) + f(x,y+1)$$

- What is the convolution like? $g(x,y) = \sum_{j=-1}^{1}\sum_{i=-1}^{1} h(i,j) f(x-i,y-j)$
- What is the size of the impulse response? Is the impulse response a constant one within the filter window $h(x,y)=1$?

$$g(x,y) = \sum_{j=-1}^{1}\sum_{i=-1}^{1} f(x-i,y-j) \times$$

- What is the impulse response analytically?

$$h(x,y) = \delta(x,y) + \delta(x-1,y) + \delta(x+1,y) + \delta(x,y-1) + \delta(x,y+1)$$

- How to represent the impulse response by image, or filter mask or filter coefficient?

0	1	0
1	1	1
0	1	0

2. LSI Systems & Transforms—understand by example

- A numerical example of convolution:

$$g(x,y)=f(x,y)*h(x,y)=\sum_{j=-1}^{1}\sum_{i=-1}^{1}h(i,j)f(x-i,y-j)$$

$$f(x,y)=\begin{bmatrix} 2 & 4 & 66 & 7 & 8 & 9 \\ 34 & 5 & 6 & 7 & 2 & 4 \\ 4 & 44 & 5 & 6 & 9 & 10 \\ 3 & 6 & 7 & 8 & 9 & 7 \\ 5 & 7 & 3 & 5 & 7 & 5 \end{bmatrix}$$

$$h(x,y)=\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

$$g(4,4)=\sum_{j=-1}^{1}\sum_{i=-1}^{1}h(i,j)f(4-i,4-j)$$

- $g(4,4)$ = (7*1+5*2+3*3+9*4+8*5+7*6+9*7+6*8+5*9)
- At each point (x,y) the response of the filter at that point is calculated as a sum of products of the filter coefficients and the corresponding image pixels in the area spanned by the filter mask.

2. LSI Systems & Transforms—Fourier transform

- Let $f(x)$ be a continuous function of a single variable x and $F(u)$ be its Fourier transform, then

$$F(u) = \Im\{f(x)\} = \int_{-\infty}^{\infty} f(x)\exp(-j2\pi ux)dx$$

$$f(x) = \Im^{-1}\{F(u)\} = \int_{-\infty}^{\infty} F(u)\exp(j2\pi ux)du$$

whe

$$f(x) = A\cos(2\pi ux + \varphi)$$

$$= (A\cos\varphi)\cos 2\pi ux - (A\sin\varphi)\sin 2\pi ux$$

$$= \tfrac{1}{2}A\exp(j2\pi ux + j\varphi) + \tfrac{1}{2}A\exp(-j2\pi ux - j\varphi)$$

$$= (\tfrac{1}{2}A\exp j\varphi)\exp j2\pi ux + (\tfrac{1}{2}A\exp(-j\varphi))\exp(-j2\pi ux)$$

- T ... Fourier tr

$F($ ... $]dxdy$

$$f(x,y) = \Im^{-1}\{F(u,v)\} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F(u,v)\exp[j2\pi(ux+vy)]dudv$$

where u and v are 2 frequency variables.

2. LSI Systems & Transforms—Fourier transform

- 2-D Fourier transform is separable

$$F(u,v)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\exp[-j2\pi(ux+vy)]dxdy$$

$$=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\exp(-j2\pi ux)\exp(-j2\pi vy)dxdy$$

$$=\int_{-\infty}^{\infty}\left[\int_{-\infty}^{\infty} f(x,y)\exp(-j2\pi ux)dx\right]\exp(-j2\pi vy)dy$$

$$=\int_{-\infty}^{\infty} F_x(u,y)\exp(-j2\pi vy)dy$$

$$?=F_x(u)F_y(v) \quad \text{only if } f(x,y)=f_1(x)f_2(y)$$

- Note

$$\exp[-j2\pi(ux+vy)]=\cos[-2\pi(ux+vy)]+j\sin(-2\pi(ux+vy))$$

$$=\cos[2\pi(ux+vy)]-j\sin(2\pi(ux+vy))$$

2. LSI Systems & Transforms—Fourier transform

- Obviously, $F(u,v)$ is in general a complex function that can be represented by

$$F(u,v) = R(u,v) + jI(u,v) = |F(u,v)| \exp[j\varphi(u,v)]$$

where

$$|F(u,v)| = \sqrt{R^2(u,v) + I^2(u,v)}$$

$$\varphi(u,v) = \tan^{-1}\left[\frac{I(u,v)}{R(u,v)}\right]$$

$$R(u,v) = |F(u,v)| \cos\varphi(u,v)$$

$$I(u,v) = |F(u,v)| \sin\varphi(u,v)$$

2. LSI Systems & Transforms—definition of DFT

- The discrete Fourier transform (DFT) of a 2-D discrete function (or image) $f(x,y)$ of size mxn is defined by:

$$F(u,v) = \frac{1}{mn} \sum_{x=0}^{m-1} \sum_{y=0}^{n-1} f(x,y) \exp[-j2\pi(ux/m + vy/n)]$$

$$f(x,y) = \frac{1}{mn} \sum_{u=0}^{m-1} \sum_{v=0}^{n-1} F(u,v) \exp[j2\pi(ux/m + vy/n)]$$

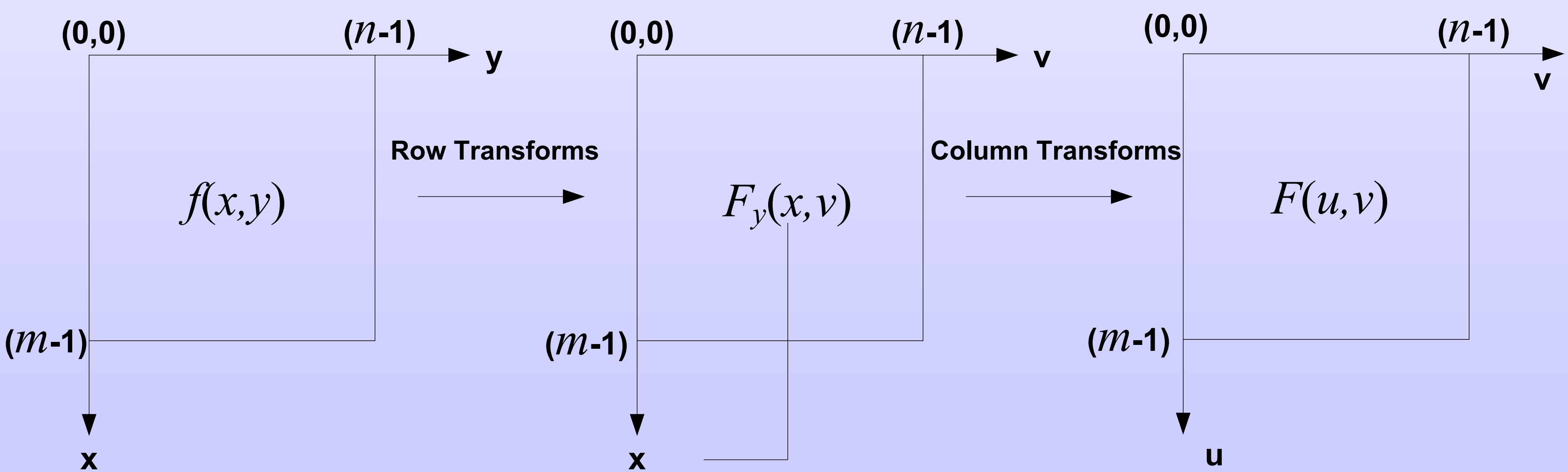
- wehere u and v are also discrete variable.

2. LSI Systems & Transforms—properties of DFT

- The 2-D DFT is also separable:

$$F(u,v)=\frac{1}{mn}\sum_{x=0}^{m-1}\exp[-j2\pi ux/m]\sum_{y=0}^{n-1}f(x,y)\exp[-j2\pi vy/n]$$

$$f(x,y)=\frac{1}{mn}\sum_{u=0}^{m-1}\exp[j2\pi ux/m]\sum_{v=0}^{n-1}F(u,v)\exp[j2\pi vy/n]$$



2. LSI Systems & Transforms—properties of DFT

- Periodicity:

$$F(u,v) = F(u+m,v) = F(u,v+n) = F(u+m,v+n)$$

- Conjugate symmetry for real image $f(x,y)$

$$F(u,v) = F^*(-u,-v), \qquad |F(u,v)| = |F(-u,-v)|$$

- Linearity and scaling:

$$\Im\{\alpha f_1(x,y) + \beta f_2(x,y) + \cdots\} = \alpha F_1(u,y) + \beta F_2(u,v) + \cdots$$

$$\Im\{f(\alpha x, \beta y)\} = \frac{1}{|\alpha\beta|} F(u/\alpha, y/\beta)$$

- Convolution theorem:

For contunuous function: $f(x,y) * g(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\alpha,\beta) g(x-\alpha, y-\beta) d\alpha d\beta$

$$f(x,y) * g(x,y) \Leftrightarrow F(u,v)G(u,v)$$

$$f(x,y)g(x,y) \Leftrightarrow F(u,v) * G(u,v)$$

- Be careful to apply this in digital image. Apply zero padding!

2. LSI Systems & Transforms—properties of DFT

- Translation

$$f(x-x_0, y-y_0) \Leftrightarrow F(u,v)\exp[-j2\pi(x_0u/m + y_0v/n)]$$

$$F(u-u_0, v-v_0) \Leftrightarrow f(x,y)\exp[j2\pi(u_0x/m + v_0y/n)]$$

- Rotation:

Let: $x = r\cos\theta, y = r\sin\theta, u = \omega\cos\varphi, v = \omega\sin\varphi$

$$f(r, \theta+\theta_0) \Leftrightarrow F(\omega, \varphi+\theta_0)$$

- Rotation Invariant Transform:

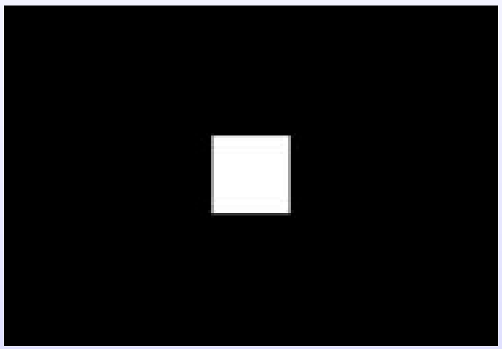
$$g(u,v) = \frac{1}{\pi}\int_0^{2\pi}\int_0^1 e^{-j(2\pi ur^2+v\theta)} f(r,\theta)drd\theta$$

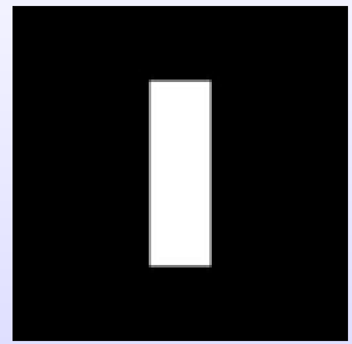
P. Yap, X.D. Jiang and A. Kot, "Two Dimensional Polar Harmonic Transforms for Invariant Image Representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 7, pp. 1259-1270, July 2010.

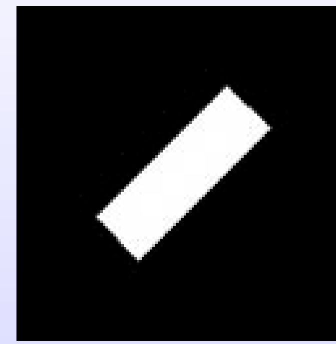
2. LSI Systems & Transforms—properties of DFT

- Examples:

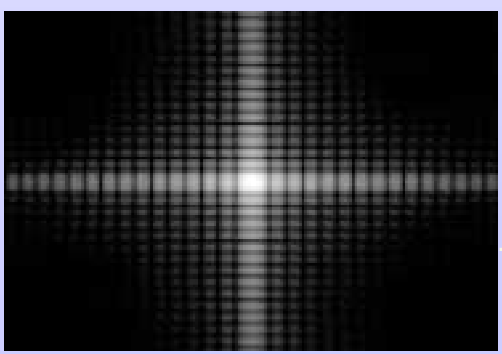
$f(x,y)$

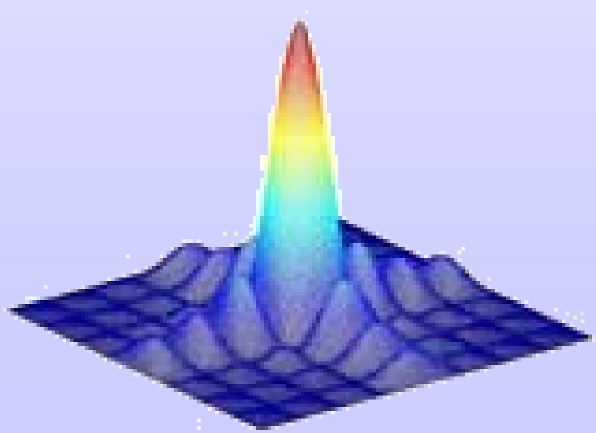


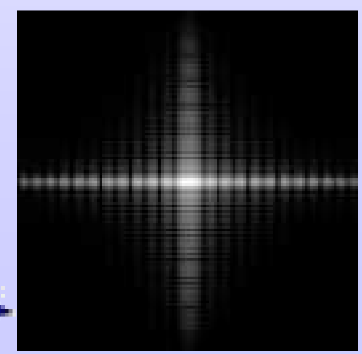


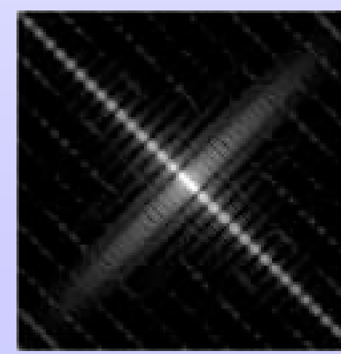


$|F(u,v)|$









2. LSI Systems & Transforms—properties of DFT

TABLE 4.1
Summary of some important properties of the 2-D Fourier transform.

Property	Expression(s)
Fourier transform	$F(u, v) = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x, y)e^{-j2\pi(ux/M+vy/N)}$
Inverse Fourier transform	$f(x, y) = \sum_{u=0}^{M-1}\sum_{v=0}^{N-1} F(u, v)e^{j2\pi(ux/M+vy/N)}$
Polar representation	$F(u, v) = \lvert F(u, v)\rvert e^{-j\phi(u,v)}$
Spectrum	$\lvert F(u, v)\rvert = \left[R^2(u, v) + I^2(u, v)\right]^{1/2}$, $R = \text{Real}(F)$ and $I = \text{Imag}(F)$
Phase angle	$\phi(u, v) = \tan^{-1}\left[\frac{I(u, v)}{R(u, v)}\right]$
Power spectrum	$P(u, v) = \lvert F(u, v)\rvert^2$
Average value	$\bar{f}(x, y) = F(0, 0) = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x, y)$
Translation	$f(x, y)e^{j2\pi(u_0x/M+v_0y/N)} \Leftrightarrow F(u - u_0, v - v_0)$ $f(x - x_0, y - y_0) \Leftrightarrow F(u, v)e^{-j2\pi(ux_0/M+vy_0/N)}$ When $x_0 = u_0 = M/2$ and $y_0 = v_0 = N/2$, then $f(x, y)(-1)^{x+y} \Leftrightarrow F(u - M/2, v - N/2)$ $f(x - M/2, y - N/2) \Leftrightarrow F(u, v)(-1)^{u+v}$

2. LSI Systems & Transforms—properties of DFT

Conjugate symmetry	$F(u,v) = F^*(-u,-v)$ $\lvert F(u,v)\rvert = \lvert F(-u,-v)\rvert$
Differentiation	$\frac{\partial^n f(x,y)}{\partial x^n} \Leftrightarrow (ju)^n F(u,v)$ $(-jx)^n f(x,y) \Leftrightarrow \frac{\partial^n F(u,v)}{\partial u^n}$
Laplacian	$\nabla^2 f(x,y) \Leftrightarrow -(u^2+v^2)F(u,v)$
Distributivity	$\Im[f_1(x,y)+f_2(x,y)] = \Im[f_1(x,y)] + \Im[f_2(x,y)]$ $\Im[f_1(x,y)\cdot f_2(x,y)] \neq \Im[f_1(x,y)]\cdot\Im[f_2(x,y)]$
Scaling	$af(x,y) \Leftrightarrow aF(u,v),\ f(ax,by) \Leftrightarrow \frac{1}{\lvert ab\rvert}F(u/a,v/b)$
Rotation	$x = r\cos\theta \quad y = r\sin\theta \quad u = \omega\cos\varphi \quad v = \omega\sin\varphi$ $f(r,\theta+\theta_0) \Leftrightarrow F(\omega,\varphi+\theta_0)$
Periodicity	$F(u,v) = F(u+M,v) = F(u,v+N) = F(u+M,v+N)$ $f(x,y) = f(x+M,y) = f(x,y+N) = f(x+M,y+N)$
Separability	See Eqs. (4.6-14) and (4.6-15). Separability implies that we can compute the 2-D transform of an image by first computing 1-D transforms along each row of the image, and then computing a 1-D transform along each column of this intermediate result. The reverse, columns and then rows, yields the same result.

2. LSI Systems & Transforms—properties of DFT

Property	Expression(s)
Computation of the inverse Fourier transform using a forward transform algorithm	$\frac{1}{MN} f^*(x, y) = \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F^*(u, v) e^{-j2\pi(ux/M + vy/N)}$ This equation indicates that inputting the function $F^*(u, v)$ into an algorithm designed to compute the forward transform (right side of the preceding equation) yields $f^*(x, y)/MN$. Taking the complex conjugate and multiplying this result by MN gives the desired inverse.
Convolution†	$f(x, y) * h(x, y) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) h(x - m, y - n)$
Correlation†	$f(x, y) \circ h(x, y) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f^*(m, n) h(x + m, y + n)$
Convolution theorem†	$f(x, y) * h(x, y) \Leftrightarrow F(u, v) H(u, v)$; $f(x, y) h(x, y) \Leftrightarrow F(u, v) * H(u, v)$
Correlation theorem†	$f(x, y) \circ h(x, y) \Leftrightarrow F^*(u, v) H(u, v)$; $f^*(x, y) h(x, y) \Leftrightarrow F(u, v) \circ H(u, v)$

2. LSI Systems & Transforms—some basic FT pairs

Some useful FT pairs:

Impulse	$\delta(x, y) \Leftrightarrow 1$
Gaussian	$A\sqrt{2\pi}\sigma e^{-2\pi^2\sigma^2(x^2+y^2)} \Leftrightarrow Ae^{-(u^2+v^2)/2\sigma^2}$
Rectangle	$\mathrm{rect}[a, b] \Leftrightarrow ab\frac{\sin(\pi ua)}{(\pi ua)}\frac{\sin(\pi vb)}{(\pi vb)}e^{-j\pi(ua+vb)}$
Cosine	$\cos(2\pi u_0 x + 2\pi v_0 y) \Leftrightarrow \frac{1}{2}\left[\delta(u + u_0, v + v_0) + \delta(u - u_0, v - v_0)\right]$
Sine	$\sin(2\pi u_0 x + 2\pi v_0 y) \Leftrightarrow j\frac{1}{2}\left[\delta(u + u_0, v + v_0) - \delta(u - u_0, v - v_0)\right]$

† Assumes that functions have been extended by zero padding.

2. LSI Systems & Transforms—discrete cosine transform

- Discrete Cosine Transform (DCT) is closely related to DFT and is widely used in image and video compression standards. The DCT pair of a 2-D discrete function (or image) $f(x,y)$ *of* size NXN can be defined as follows:

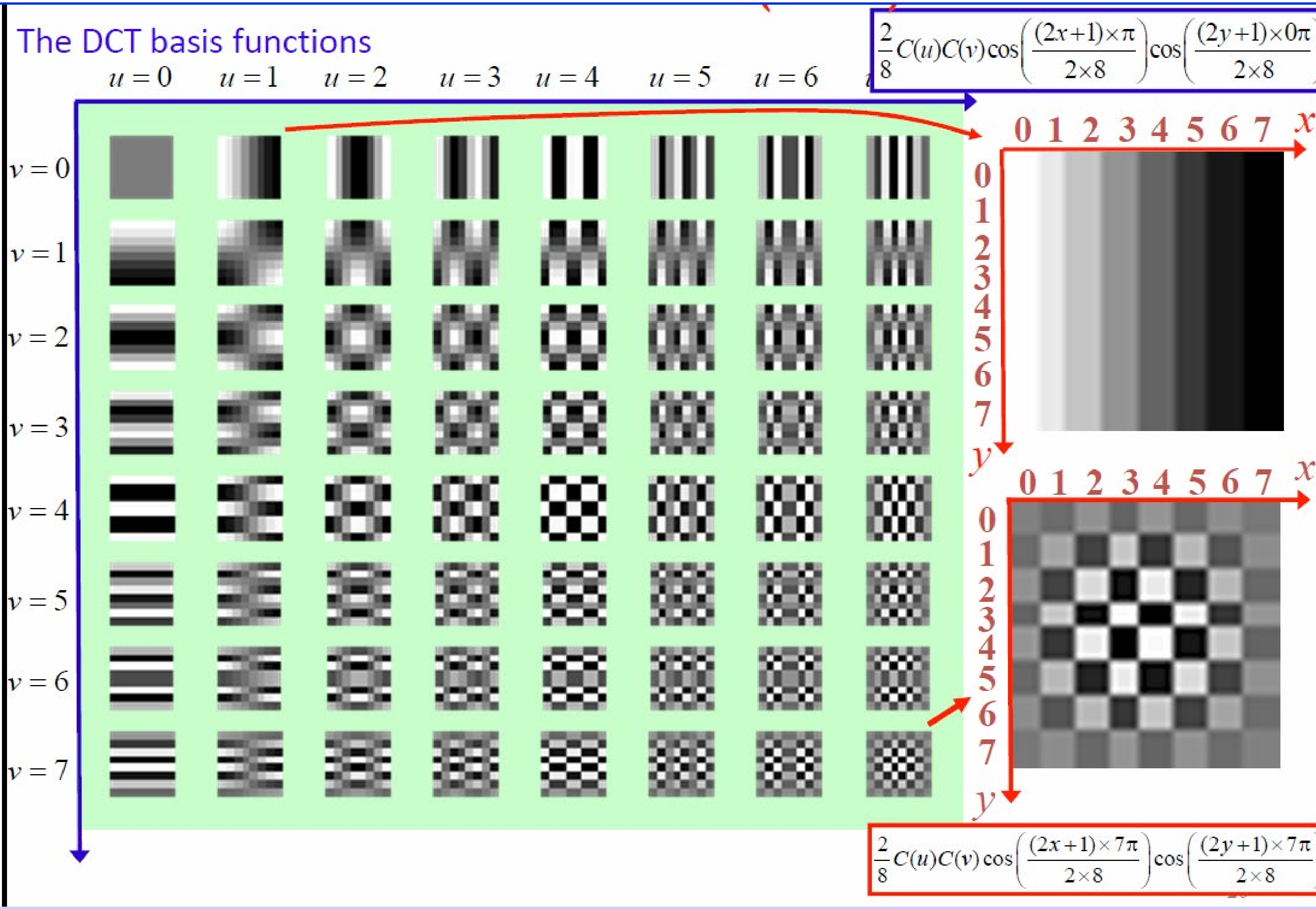
$$F(u,v)=\frac{2}{N}C(u)C(v)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}f(x,y)\cos\left(\frac{(2x+1)u\pi}{2N}\right)\cos\left(\frac{(2y+1)v\pi}{2N}\right)$$

$$f(x,y)=\frac{2}{N}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}C(u)C(v)F(u,v)\cos\left(\frac{(2x+1)u\pi}{2N}\right)\cos\left(\frac{(2y+1)v\pi}{2N}\right)$$

where

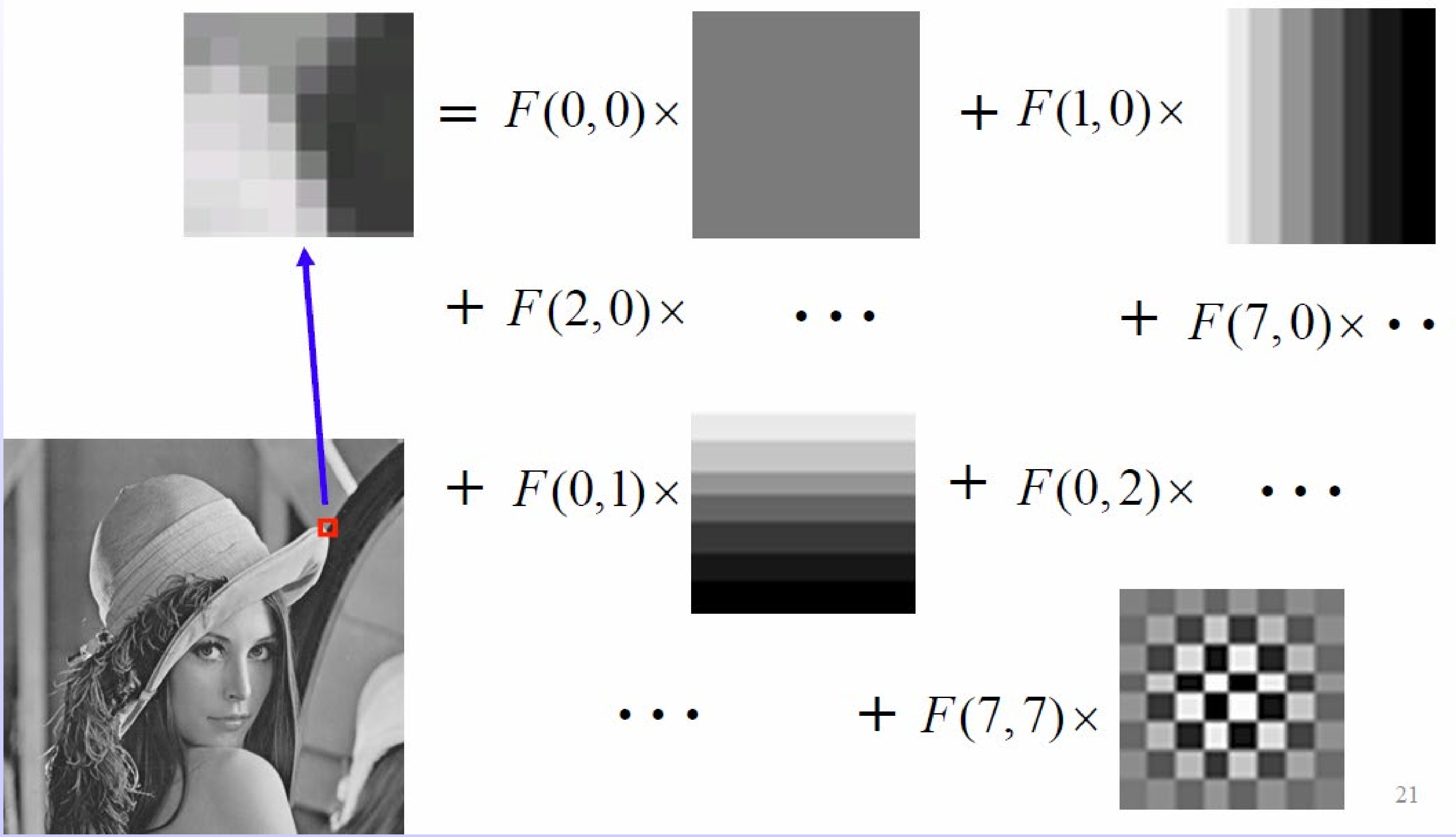
$$C(u),C(v)=\begin{cases}\frac{1}{\sqrt{2}} & u,v=0\\ 1 & u,v=1,2,\ldots,N-1\end{cases}$$

2. LSI Systems & Transforms—discrete cosine transform



2. LSI Systems & Transforms—discrete cosine transform

$$f(x,y)=\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}F(u,v)\left(\frac{2}{N}C(u)C(v)\cos\left(\frac{(2x+1)u\pi}{2N}\right)\cos\left(\frac{(2y+1)v\pi}{2N}\right)\right)$$



2. LSI Systems & Transforms—image sampling

- An digital image $f_d(m,n)$ is obtained by sampling and quantizing a continuous analogy image $f_c(x,y)$. Here the sampling converts the continuous variable x, y into integer m, n and the quantization convert the continuous variable f_c into a finite set of numbers f_d . As the quantization has less impact than sampling to the image processing, we will focus more on the sampling process and theory.
- Given a continuous analogy image $f_c(x,y)$, it is very simple to get its discrete image $f_d(m,n)$ mathematically by

$$f_d(m,n) = f_c(m\Delta x, n\Delta y) = f_c(x,y)\Big|_{\text{Let } x=m\Delta x, y=n\Delta y}$$

- Does $f_d(m,n)$ contain the same information as $f_c(x,y)$? Under what conditions is it yes? Mathematically analyzing this, however, needs abstract mathematical tools.

2. LSI Systems & Transforms—image sampling

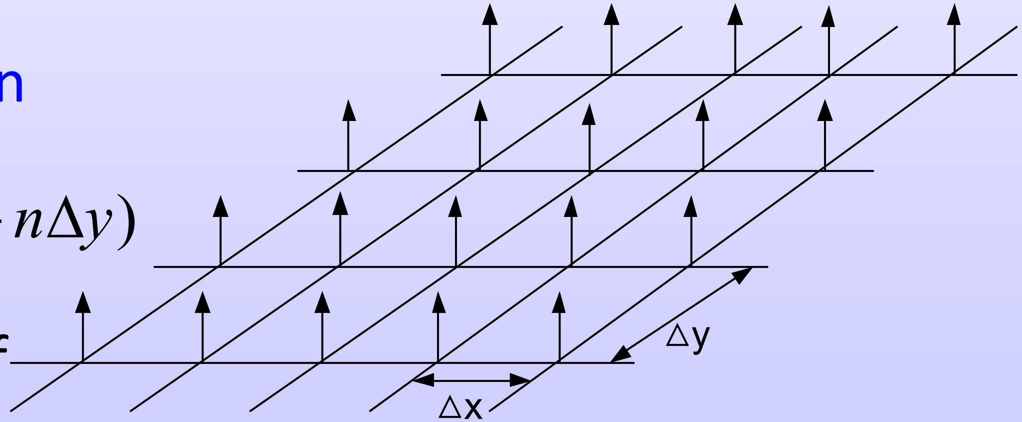
- A 2-D function $f_c(x,y)$ is band-limited if its Fourier transform $F_c(u,v)$ is zero outside a bounded spatial frequency support; e.g.,

$$F_c(u,v) = 0, \quad \text{for } |u| > U_0, |v| > V_0$$

- where $2U_0$ and $2V_0$ are referred to as the x and y bandwidths of the 2-D function. Why $2U_0$ and $2V_0$?
- In practice, real-world images can be well approximated by band-limited signals.
- Define a 2-D sampling function

$$s(x,y) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} \delta(x - m\Delta x, y - n\Delta y)$$

- It has the Fourier transform of



$$S(u,v) = \frac{1}{\Delta x \Delta y} \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} \delta(u - m/\Delta x, v - n/\Delta y)$$

2. LSI Systems & Transforms—image sampling

- Multiply the continuous image $f_c(x,y)$ with the sampling image $s(x,y)$ yield

$$f_d(x,y) = f_c(x,y)s(x,y) = \sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty} f_c(m\Delta x, n\Delta y)\delta(x - m\Delta x, y - n\Delta y)$$

- The Fourier transform of the sampled image $f_d(x,y)$ in continuous domain is:

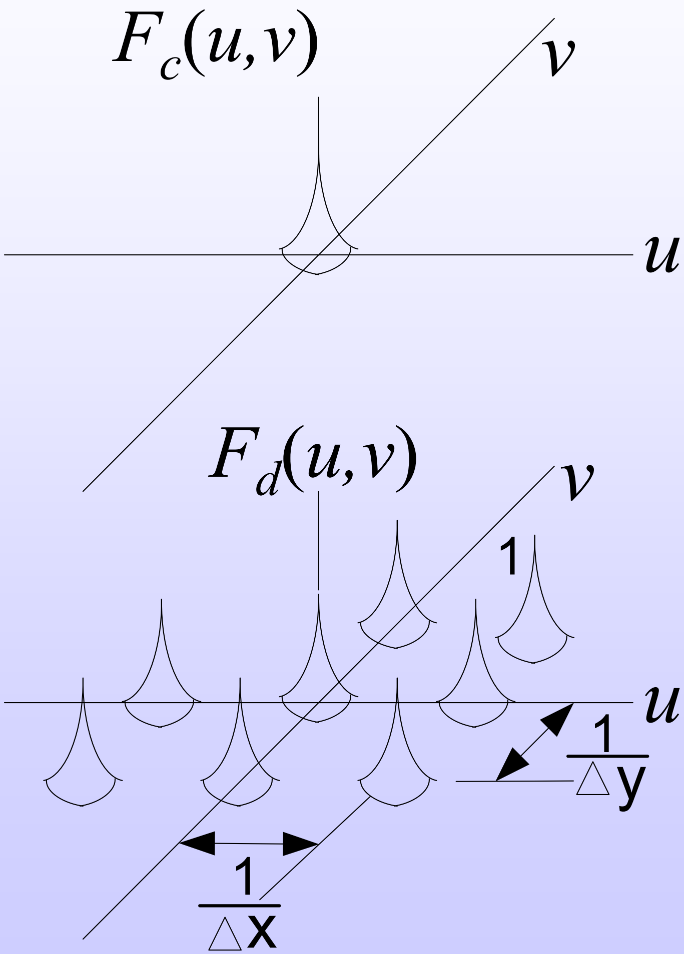
$$F_d(u,v) = F_c(u,v) * S(u,v) = \frac{1}{\Delta x \Delta y}\sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty} F_c(u,v) * \delta(u - m/\Delta x, y - n/\Delta y)$$

$$= \frac{1}{\Delta x \Delta y}\sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty} F_c(u - m/\Delta x, y - n/\Delta y) = \frac{1}{\Delta x \Delta y}\sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty} F_c(u - mf_{xs}, y - nf_{ys})$$

- It is a periodic replication of $F_c(u,v)$, on a rectangular grid with spacing $(1/\Delta x, 1/\Delta y)$.

2. LSI Systems & Transforms—image sampling

- The spectrum of the sampled image $f_d(x,y)$ consists of the spectrum of the continuous $f_c(x,y)$ image (top) infinitely repeated over the frequency plane in a rectangular grid with spacing $(1/\Delta x,\ 1/\Delta y)$. It is a periodic replication of $F_c(u,v)$, on a rectangular grid with spacing $(1/\Delta x,\ 1/\Delta y)$.



2. LSI Systems & Transforms—image sampling

- If the x, y sampling frequencies are greater than the bandwidths, or if the sampling intervals are smaller than the reciprocal of bandwidths, i.e.,

$$f_{xs} = \frac{1}{\Delta x} \geq 2U_0 \quad \& \quad f_{ys} = \frac{1}{\Delta y} \geq 2V_0 \quad \text{or} \quad \Delta x \leq \frac{1}{2U_0} \quad \& \quad \Delta y \leq \frac{1}{2V_0}$$

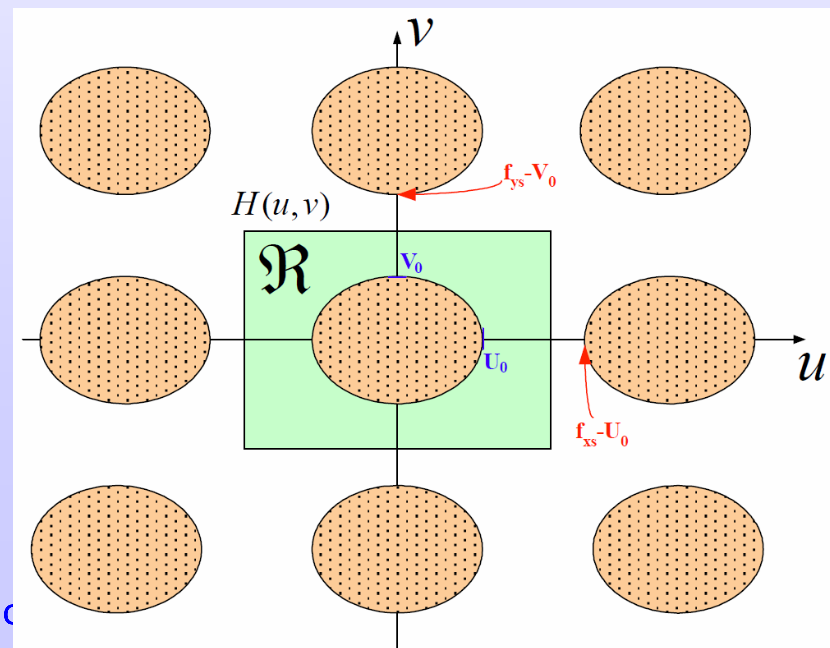
- Then $F_c(u,v)$ can be recovered from $F_d(u,v)$ by using a low-pass filter with frequency response

$$H(u,v) = \begin{cases} \Delta x \Delta y, & (u,v) \in \Re \\ 0, & \text{otherwise} \end{cases}$$

- That is

$$F_c(u,v) = F_d(u,v)H(u,v)$$

$$f_c(x,y) = f_d(x,y) * h(x,y)$$



2. LSI Systems & Transforms—image sampling

$$f_c(x,y) = h(x,y) * f_d(x,y)$$

$$= h(x,y) * \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f_c(m\Delta x, n\Delta y)\delta(x - m\Delta x, y - n\Delta y)$$

$$= \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f_c(m\Delta x, n\Delta y) h(x,y) * \delta(x - m\Delta x, y - n\Delta y)$$

$$= \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f_c(m\Delta x, n\Delta y) h(x - m\Delta x, y - n\Delta y)$$

$$= \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f_d(m,n) h(x - m\Delta x, y - n\Delta y)$$

- Therefore, all information of the continuous image $f_c(x,y)$ can be recovered from the discrete image $f_d(m,n)$.

2. LSI Systems & Transforms—image sampling

Sampling Theorem:

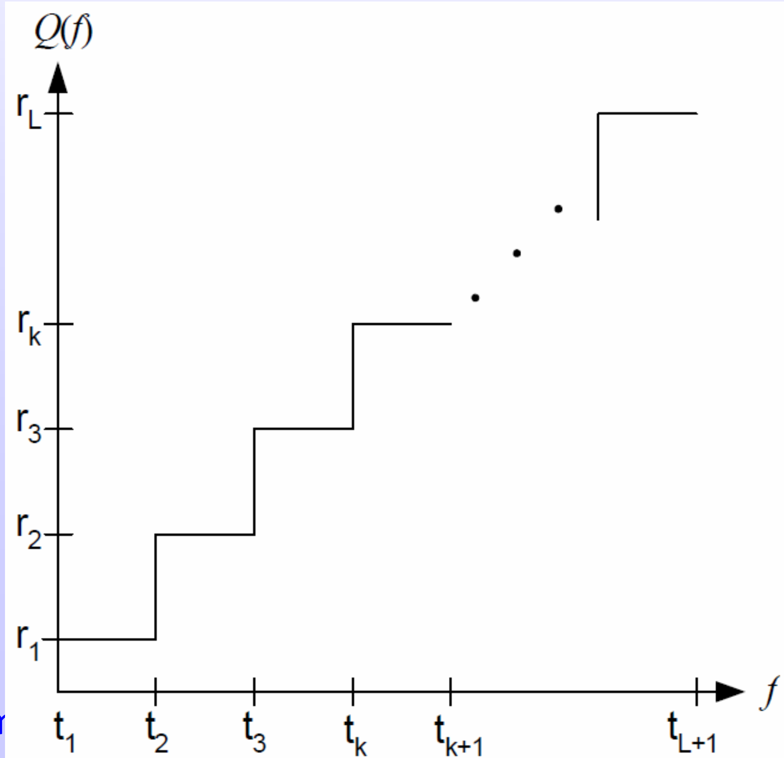
- A band-limited image $f_c(x,y)$ with bandwidths $(2U_0, 2V_0)$ sampled uniformly on a rectangular grid with spacing $(\Delta x,\Delta y)$ can be recovered without error from the sampled values $f_c(m\Delta x,n\Delta y)= f_d(m,n)$ provided that the sampling rates (f_{xs}, f_{ys}) are greater than the Nyquist rates.
- The lower bounds of the required sampling rates, the band width, are known as the Nyquist rates or Nyquist frequencies. Their reciprocals are known as the Nyquist intervals.
- Sampling below the Nyquist rates will cause the periodic replications of $F_c(u,v)$ to overlap, resulting in a distorted spectrum $F_d(u,v)$, in which $F_c(u,v)$ is irrevocably lost—a phenomenon that is known as aliasing.
- Aliasing can be avoided or reduced by low-pass filtering the image $f_c(x,y)$ before sampling so that its bandwidth is less than the sampling frequency. (at the expense of what?)

2. LSI Systems & Transforms—image quantization

- An image (scalar) quantizer Q maps a continuous intensity variable f into a discrete variable f_q, which takes values from a finite set $\{r_1, \ldots, r_L\}$ of L numbers.
- A common L-level image quantizer has the following form:

$$f_q = Q(f) = r_k, \quad \text{if } t_k \le f < t_{k+1} \quad \text{for} \quad k = 1, \ldots, L$$

- where $\{t_1, \ldots, t_{L+1}\}$ is a set of increasing values known as decision levels, and r_k is the kth reconstruction level corresponding to the decision interval $[t_k, t_{k+1})$.
- As such a quantizer performs many-(much)-to-1 mapping, it introduces distortions or errors, known as quantization errors.



2. LSI Systems & Transforms—image quantization

- An image (scalar) quantizer Q maps a continuous intensity variable f *into a* discrete variable f_q, which takes values from a finite set $\{r_1, \ldots, r_L\}$ of L numbers.
- The choice of quantized levels should minimize the quantization error between f and f_q, . For example, Lloyd-Max quantizer minimizes the mean squared error:

$$\varepsilon = E\{(f - f_d)^2\} = \int_{t_1}^{t_{L+1}} (f - Q(f))^2 p(f)df = \sum_{k=1}^{L} \int_{t_k}^{t_{k+1}} (f - r_k)^2 p(f)df$$

- Differentiating the above equation with respect to r_k and t_k, and setting the results to zero, it can be shown that

$$t_k = \frac{r_k + r_{k-1}}{2}, \quad \text{for } k = 2, \ldots, L \qquad r_k = \frac{\int_{t_k}^{t_{k+1}} fp(f)df}{\int_{t_k}^{t_{k+1}} p(f)df}, \quad \text{for } k = 1, \ldots, L$$

- These equations are to be solved simultaneously— a non-trivial problem that usually calls for an iterative scheme such as Newton method.

2. LSI Systems & Transforms—image quantization

- If $p(f)$ is uniform, the Lloyd-Max quantizer becomes a linear quantizer with equal intervals between the decision levels and the reconstruction levels, given as

$$t_k = t_{k-1} + q, \quad r_k = t_k + \frac{q}{2}, \quad q = \frac{t_{L+1} + t_1}{L}$$

- and the mean square error can be shown as

$$\varepsilon_{linear} = q^2 / 12$$

- It is easy to show that the signal-to-noise ratio (SNR) achievable by a B-bit (2^B-level) linear quantizer for a uniform distributed function is given by

$$SNR = 10\log_{10}\frac{\sigma^2}{\varepsilon} = 10\log_{10} 2^{2B} \approx 6B\text{dB}$$

where σ^2 is the variance of the f.

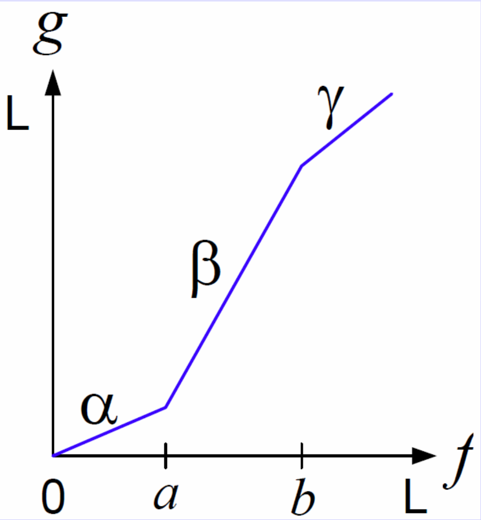
3. Image Enhancement—Outline

- Simple Point Processing Approaches
- Histogram Equalization
- Image Smoothing
- Image Sharpening
- Nonlinear Image Processing

3. Image Enhancement—point processing

- Image processing operations are designed to enhance image content or features so that they are more suitable for display or analysis.
- Many image enhancement processes are point and memoryless operations which map input image gray-level to output gray-level according to a transformation g=$T(f)$. For example:
  - Power Transformation (gamma correction): $g=c\,f^{\gamma}$
  - Log Transformation: $g=c\log(1+f)$
  - Piecewise Linear Transformation

$$g = T(f) = \begin{cases} \alpha f, & 0 \le f < a \\ \beta(f-a)+T(a), & a \le f < b \\ \gamma(f-b)+T(b), & b \le f < L \end{cases}$$



  - Histogram equalization

3. Image Enhancement—simple point processing

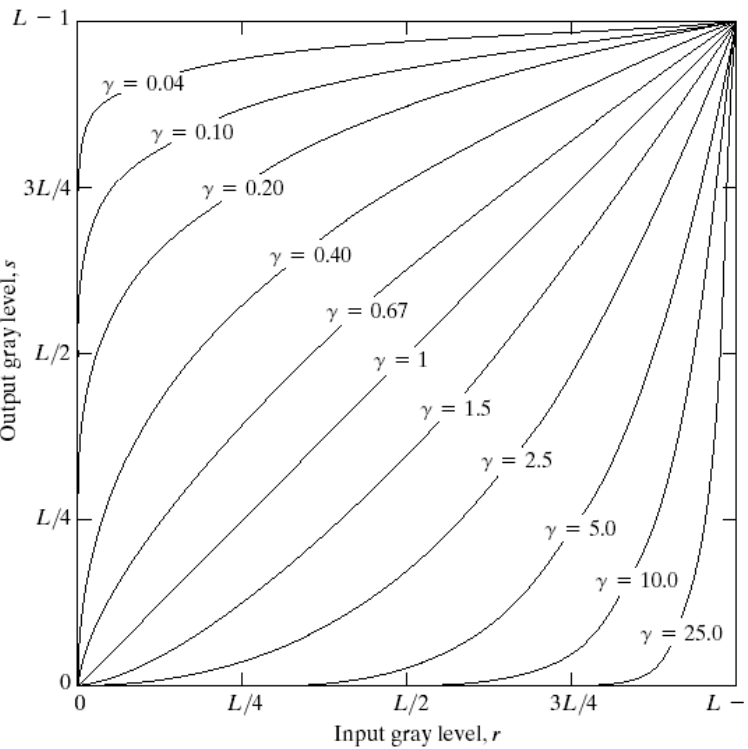
Power Transformation

gamma correction

$g = c f^{\gamma}$

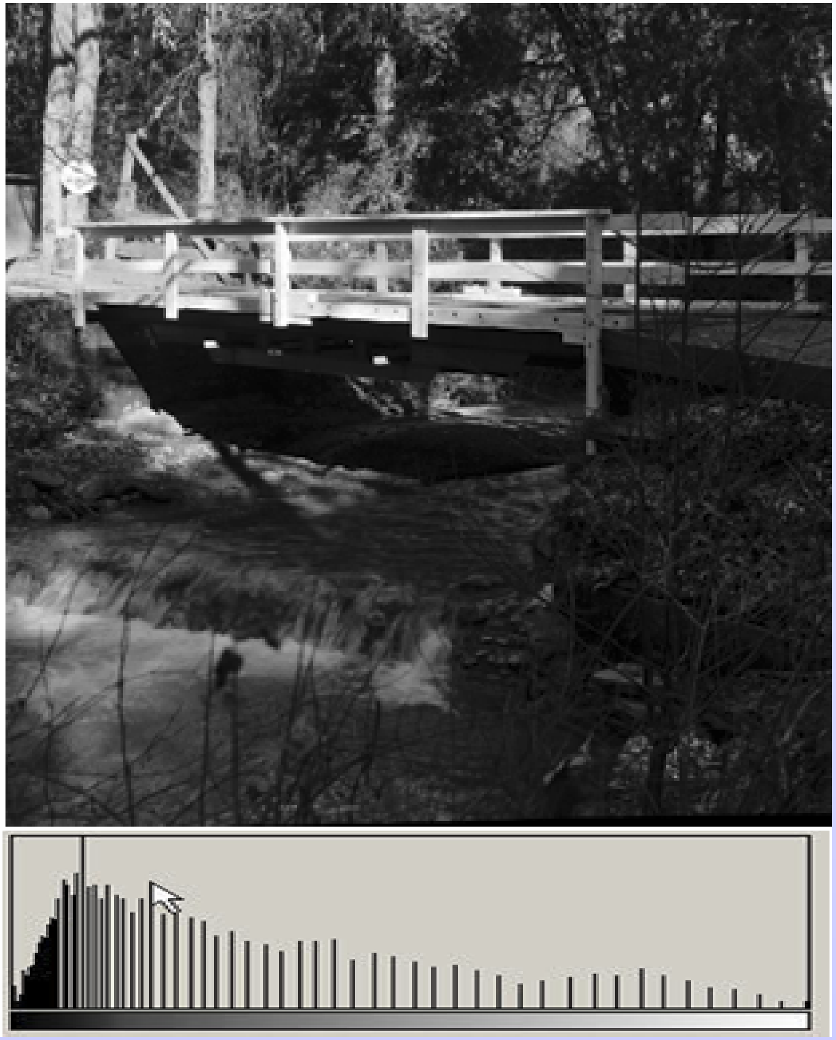
a b
c d

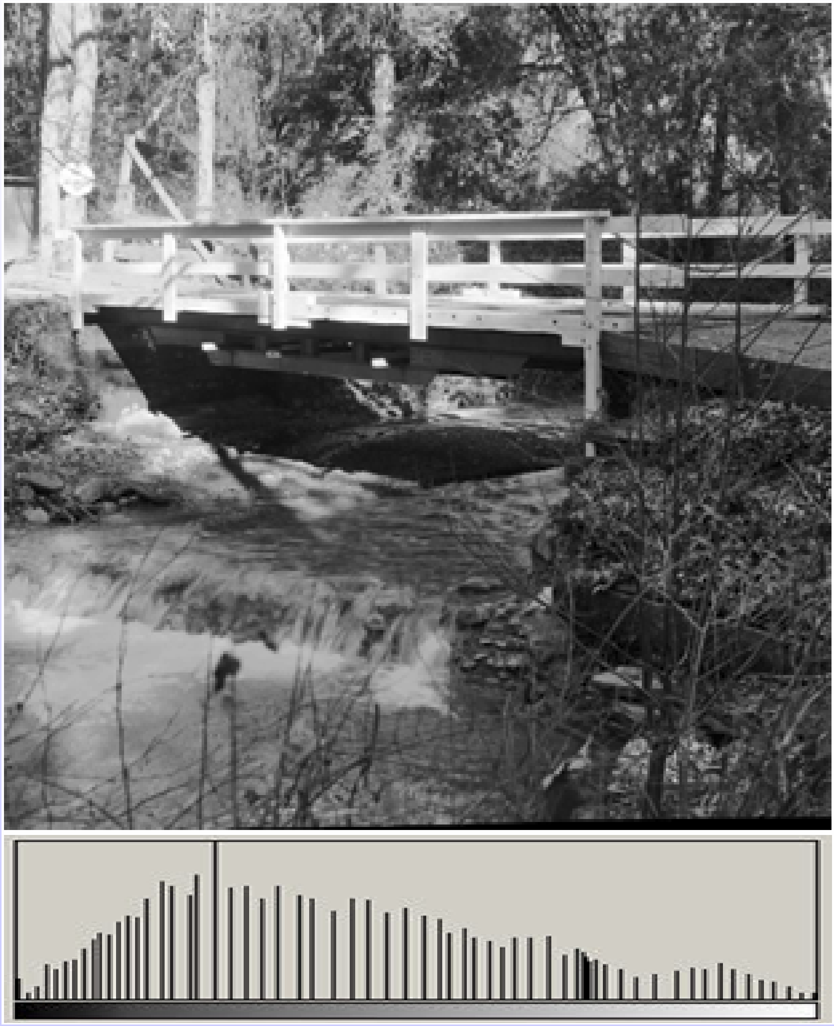
FIGURE 3.9 (a) Aerial image. (b)–(d) Results of applying the transformation in Eq. (3.2-3) with $c = 1$ and $\gamma = 3.0, 4.0,$ and 5.0, respectively.



3. Image Enhancement—simple point processing

Log Transformation: $g = c\log(1+f)$





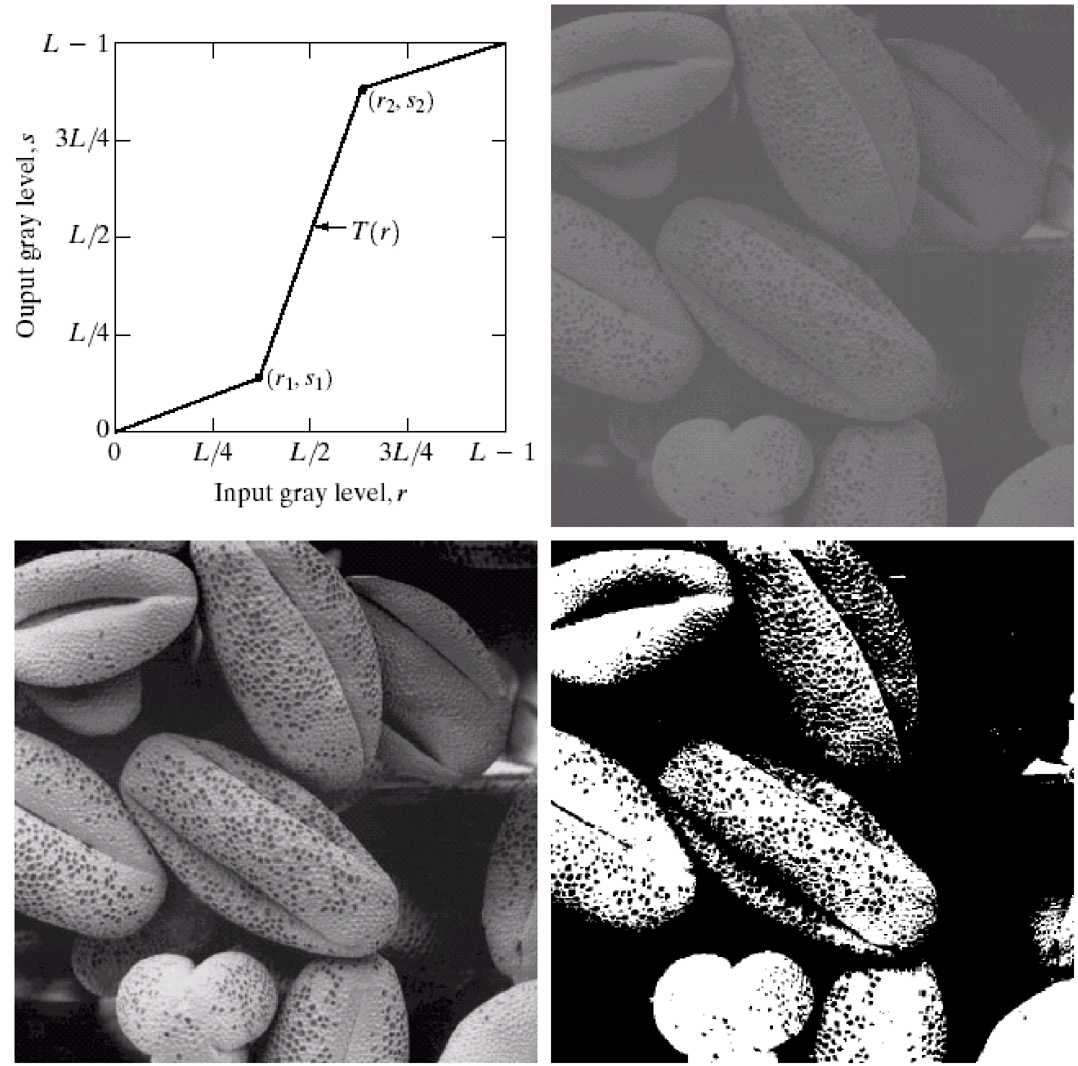
3. Image Enhancement—simple point processing

Piecewise Linear Transformation:

Contrast stretching:

$$g = T(f)$$

$$= \begin{cases} \alpha f, & 0 \le f < a \\ \beta(f-a) + T(a), & a \le f < b \\ \gamma(f-b) + T(b), & b \le f < L \end{cases}$$



3. Image Enhancement—histogram equalization

- Histogram equalization aims to obtain a uniform histogram for the output image $g(x,y)$ by transforming the gray-level f of the input image $f(x,y)$ into g

$$g = T(f)$$

- Histogram equalization algorithm:

$$c(f) = \sum_{t=0}^{f} p_f(t) = \sum_{t=0}^{f} \frac{n_t}{n}, \qquad f = 0, 1, \ldots, L$$

$$g = T(f) = round\left[\frac{c(f) - c_{\min}}{1 - c_{\min}} L\right], \quad c(f) \geq c_{\min}$$

- where t is a dummy variable of the summation. c_{min} is the smallest positive value of all $c(f)$ obtained, round[ ] rounds a real number to an integer. g is approximately uniformly distributed in [0, L] .

3. Image Enhancement—histogram equalization

- Theoretical analysis of the histogram equalization can only be done for continuous variable.
- Let f be the continuous gray value normalize to [0,1]
- Let the transform $g=T(f)$ single-valued, monotonically increasing in $0 \leq g=T(f) \leq 1$.
- The inverse transform is $f=T^{-1}(g)$ should also be single-valued and monotonically increasing.
- From probability theory, if original gray level pdf $p_f(f)$ and $T(f)$ are known, $T^{-1}(g)$ satisfies the above condition, then the transformed gray level pdf $p_g(g)$ is

$$p_g(g) = p_f(f)\frac{df}{dg}$$

3. Image Enhancement—histogram equalization

- Consider that $$g = T(f) = \int_0^f p_f(t)dt$$
- The above function is the cumulative distribution function (cdf) of f. cdf is single-valued and monotonically increasing.

$$\because \frac{dg}{df} = p_f(f)$$

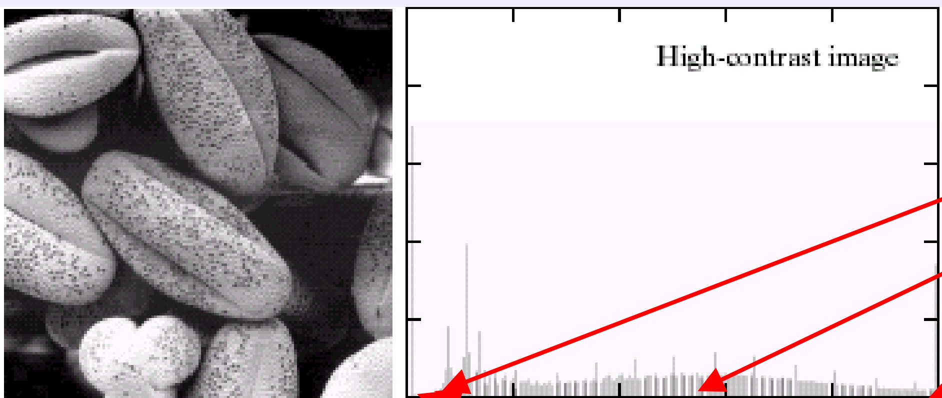
$$\therefore p_g(g) = p_f(f)\frac{df}{dg} = p_f(f)\frac{1}{p_f(f)} = 1$$

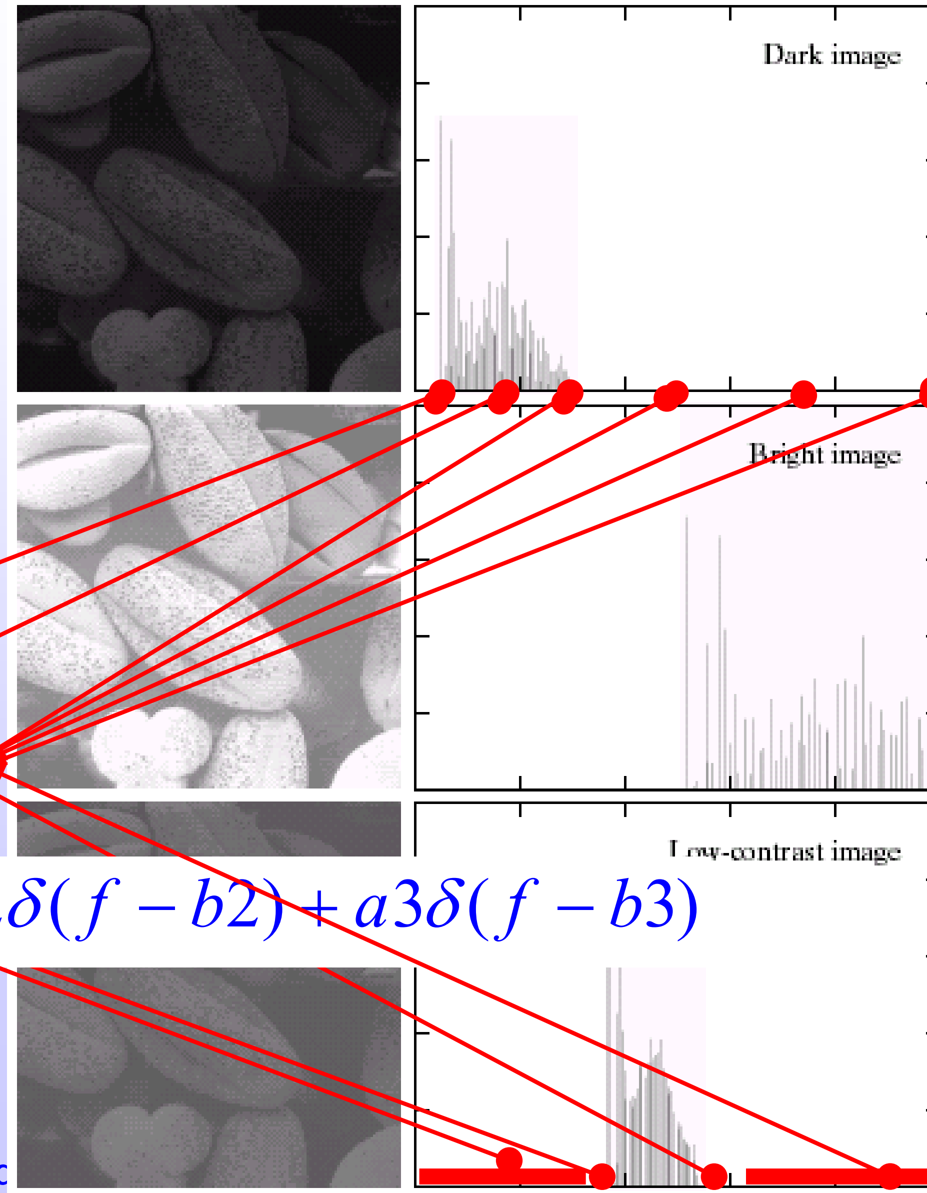
- Therefore, the transformed gray value has uniform distribution.
- The histogram equalization $$c(f) = \sum_{t=0}^{f} p_f(t) = \sum_{t=0}^{f} \frac{n_t}{n}$$
- is the discrete version of $$g = T(f) = \int_0^f p_f(t)dt$$

3. Image Enhancement—histogram equalization

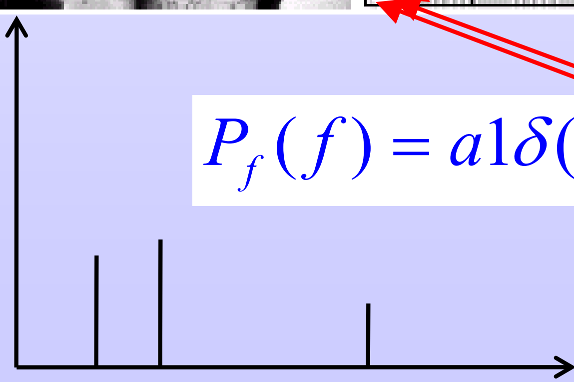
Understand the histogram equalization and its effect on image.

$$c(f) = \sum_{t=0}^{f} p_f(t) = \sum_{t=0}^{f} \frac{n_t}{n}$$



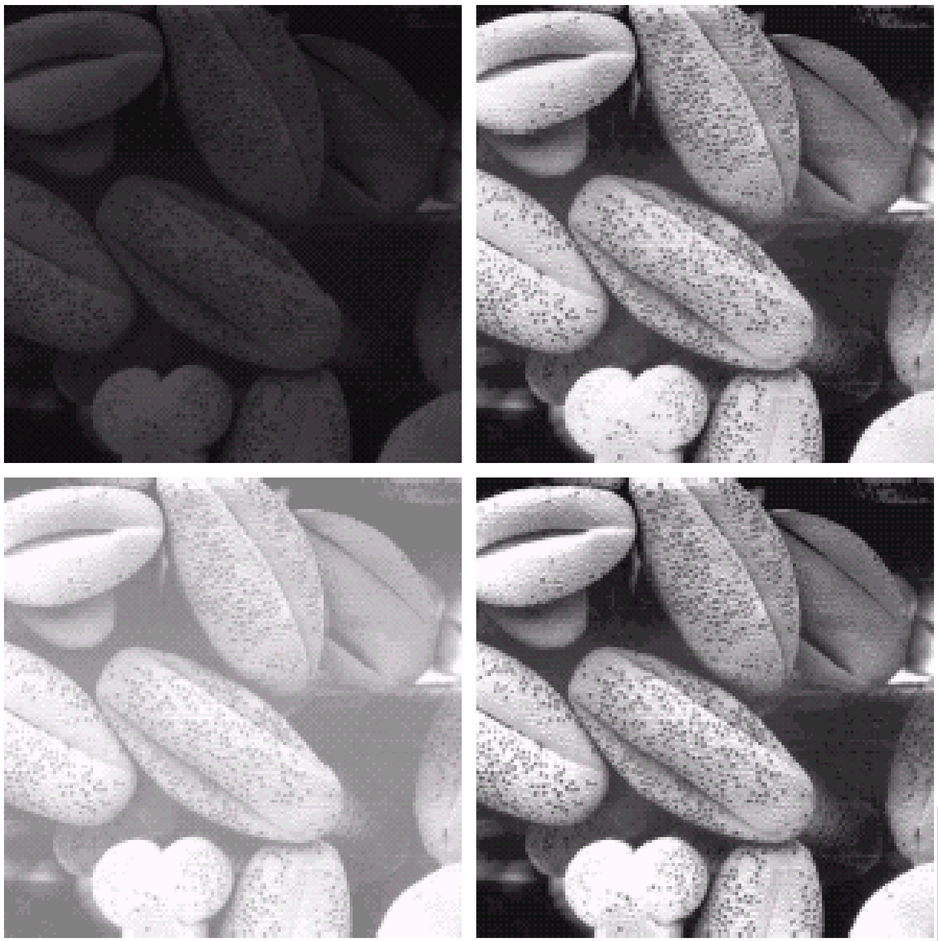


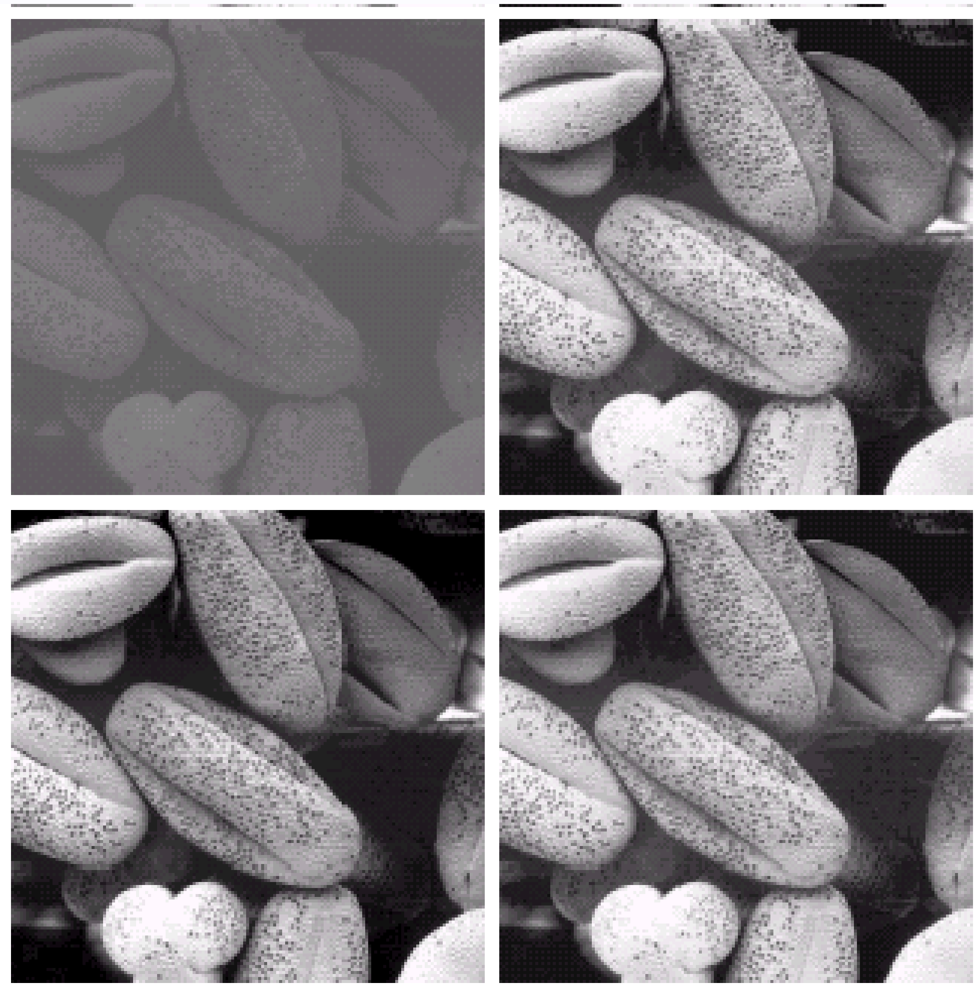
$$P_f(f) = a1\delta(f - b1) + a2\delta(f - b2) + a3\delta(f - b3)$$



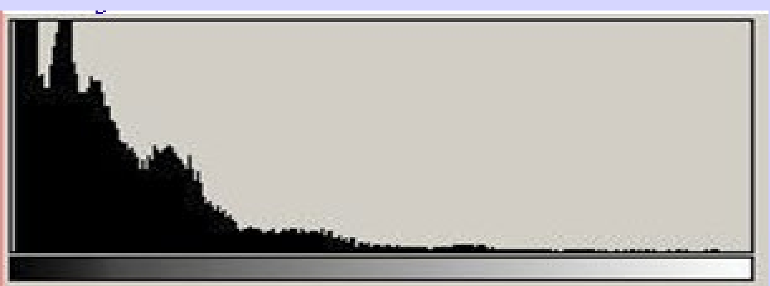
3. Image Enhancement—histogram equalization

- histogram equalization





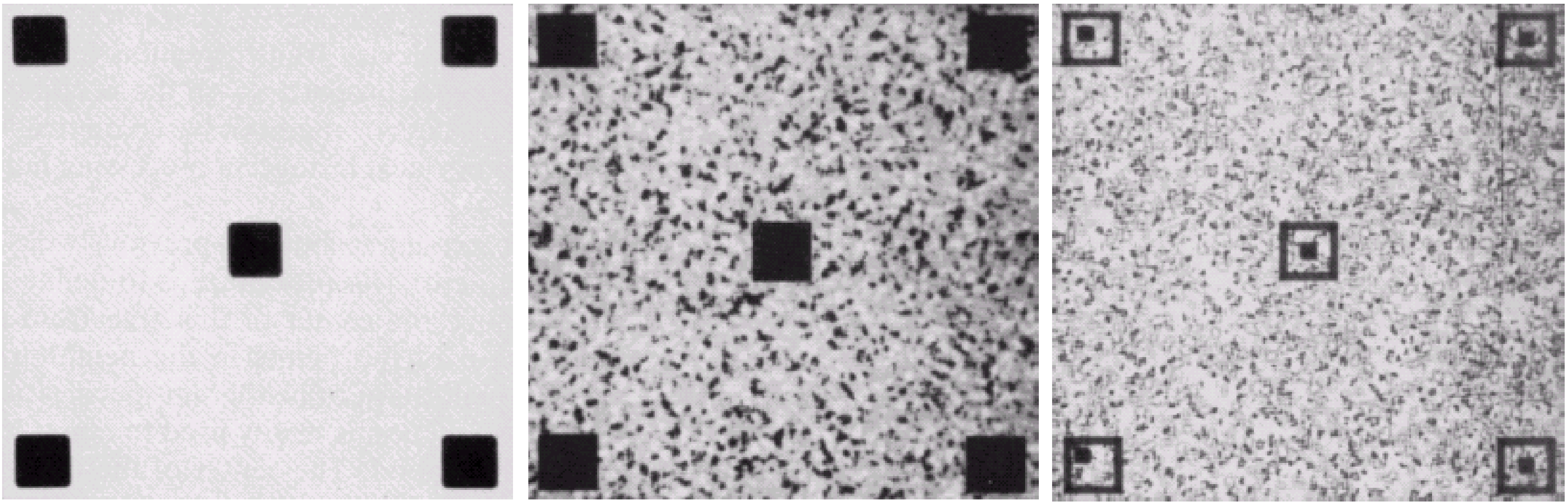
3. Image Enhancement—histogram equalization



Histogram Equalization

3. Image Enhancement—histogram equalization

- Local histogram equalization



a b c

FIGURE 3.23 (a) Original image. (b) Result of global histogram equalization. (c) Result of local histogram equalization using a 7 × 7 neighborhood about each pixel.

3. Image Enhancement—image smoothing

- The basic form of image filtering in the spatial and frequency domain can be given as

$$
\begin{array}{ccccc}
g(x,y) & = & f(x,y) & * & h(x,y) \\
\Updownarrow & & \Updownarrow & & \Updownarrow \\
G(u,v) & = & F(u,v) & & H(u,v)
\end{array}
$$

$$g(x,y) = f(x,y) * h(x,y) = \sum_{j=-\infty}^{\infty} \sum_{i=-\infty}^{\infty} h(i,j) f(x-i, y-j)$$

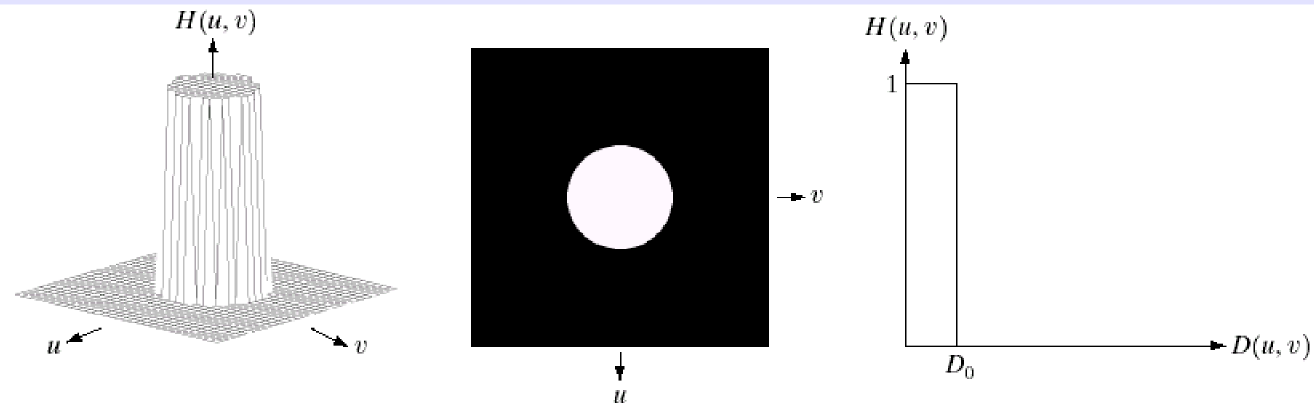
$$= \sum_{j=-3}^{3} \sum_{i=-3}^{3} h(i,j) f(x-i, y-j), \ \text{ if } h(x,y) = 0 \ \text{ for } -3 < x, y < 3$$

- At each point (x,y) the response of the filter at that point is calculated as a sum of products of the filter coefficients and the corresponding image pixels in the area spanned by the filter mask centered at (x,y).

3. Image Enhancement—image smoothing

- Image smoothing filters are used for blurring and for noise reduction.
- These filters are also known as averaging or low pass filters.
- Ideal lowpass filter

$$H(u,v) = \begin{cases} 1, & \text{if } D(u,v) \le D_0 \\ 0, & \text{if } D(u,v) > D_0 \end{cases} \qquad D(u,v) = \sqrt{u^2 + v^2}$$



a b c

FIGURE 4.10 (a) Perspective plot of an ideal lowpass filter transfer function. (b) Filter displayed as an image. (c) Filter radial cross section.

3. Image Enhancement—image smoothing

- Ideal lowpass filtering

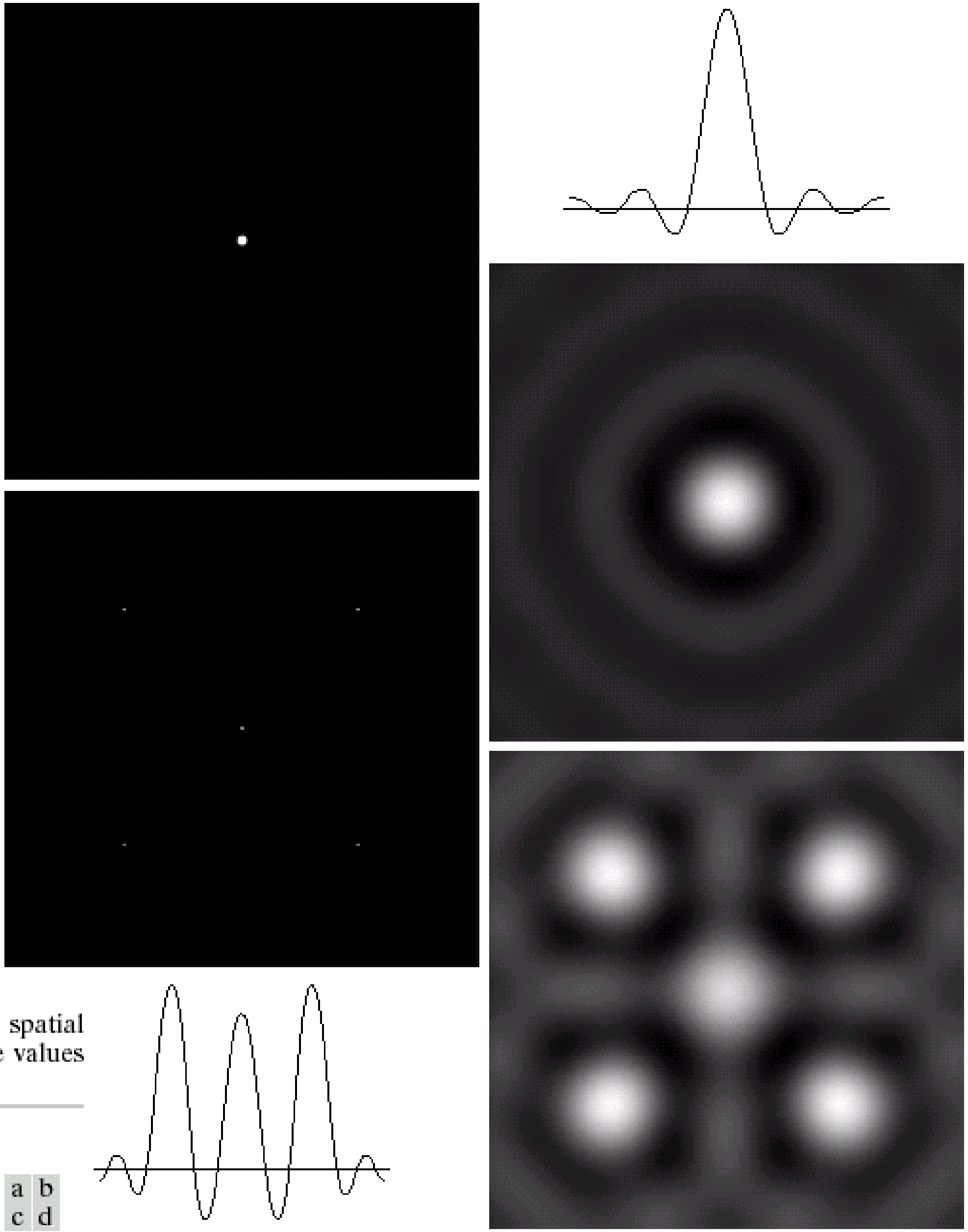
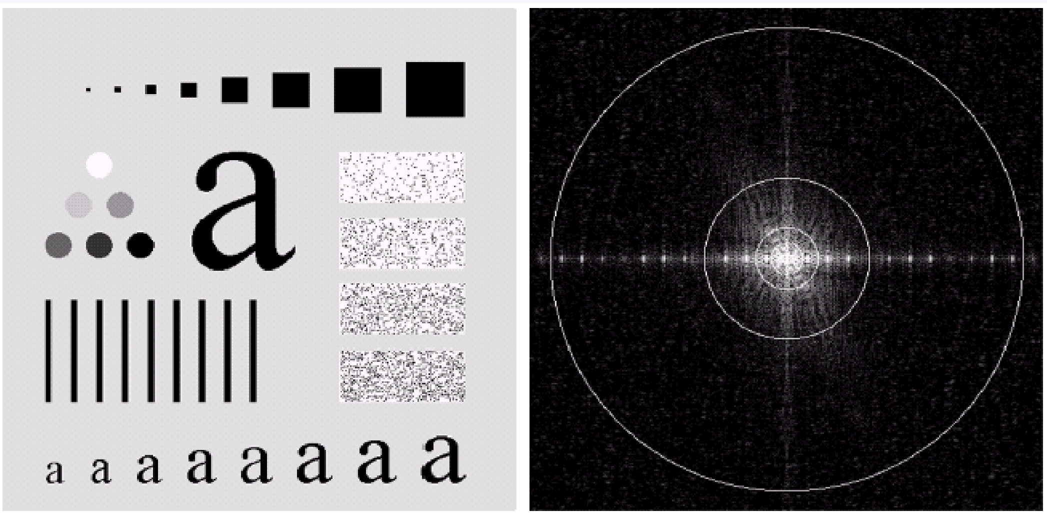


FIGURE 4.13 (a) A frequency-domain ILPF of radius 5. (b) Corresponding spatial filter (note the ringing). (c) Five impulses in the spatial domain, simulating the values of five pixels. (d) Convolution of (b) and (c) in the spatial domain.

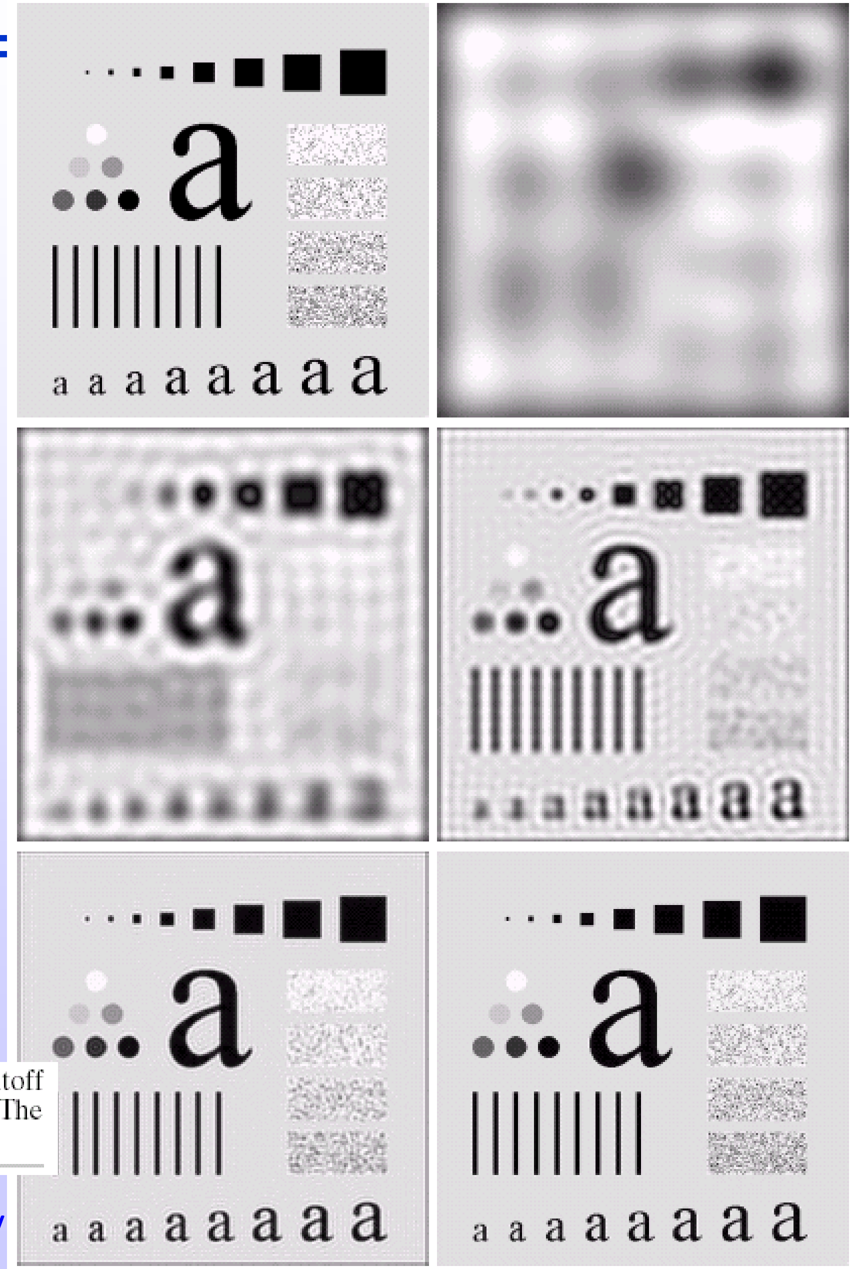
3. Image Enhancement—image smoothing

- Ideal lowpass filtering



a b

FIGURE 4.11 (a) An image of size 500 × 500 pixels and (b) its Fourier spectrum. The superimposed circles have radii values of 5, 15, 30, 80, and 230, which enclose 92.0, 94.6, 96.4, 98.0, and 99.5% of the image power, respectively.



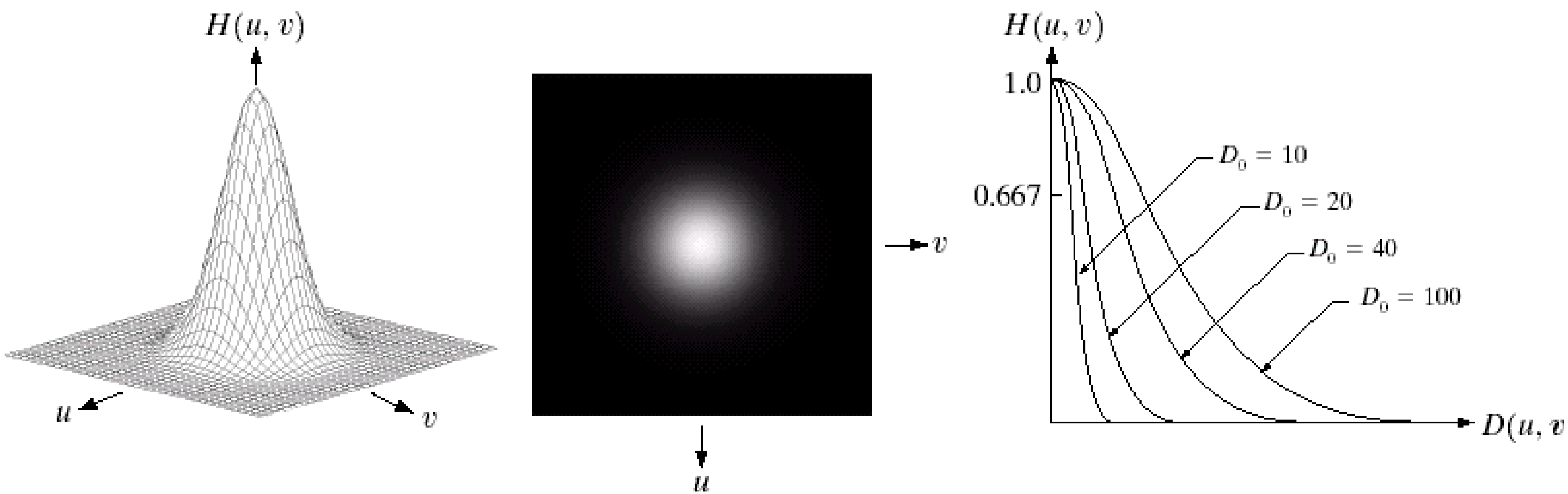
a b
c d
e f

FIGURE 4.12 (a) Original image. (b)–(f) Results of ideal lowpass filtering with cutoff frequencies set at radii values of 5, 15, 30, 80, and 230, as shown in Fig. 4.11(b). The power removed by these filters was 8, 5.4, 3.6, 2, and 0.5% of the total, respectively.

3. Image Enhancement—image smoothing

- Gaussian lowpass filtering (GLPF)

$$G(u,v) = \frac{1}{2\pi\sigma^2} e^{-\frac{u^2+v^2}{2D_0}}$$



a b c

FIGURE 4.17 (a) Perspective plot of a GLPF transfer function. (b) Filter displayed as an image. (c) Filter radial cross sections for various values of D_0.

3. Image Enhancement—image smoothing

- Gaussian lowpass filtering (GLPF)

$$G(u,v) = \frac{1}{2\pi\sigma^2} e^{-\frac{u^2+v^2}{2D_0}}$$

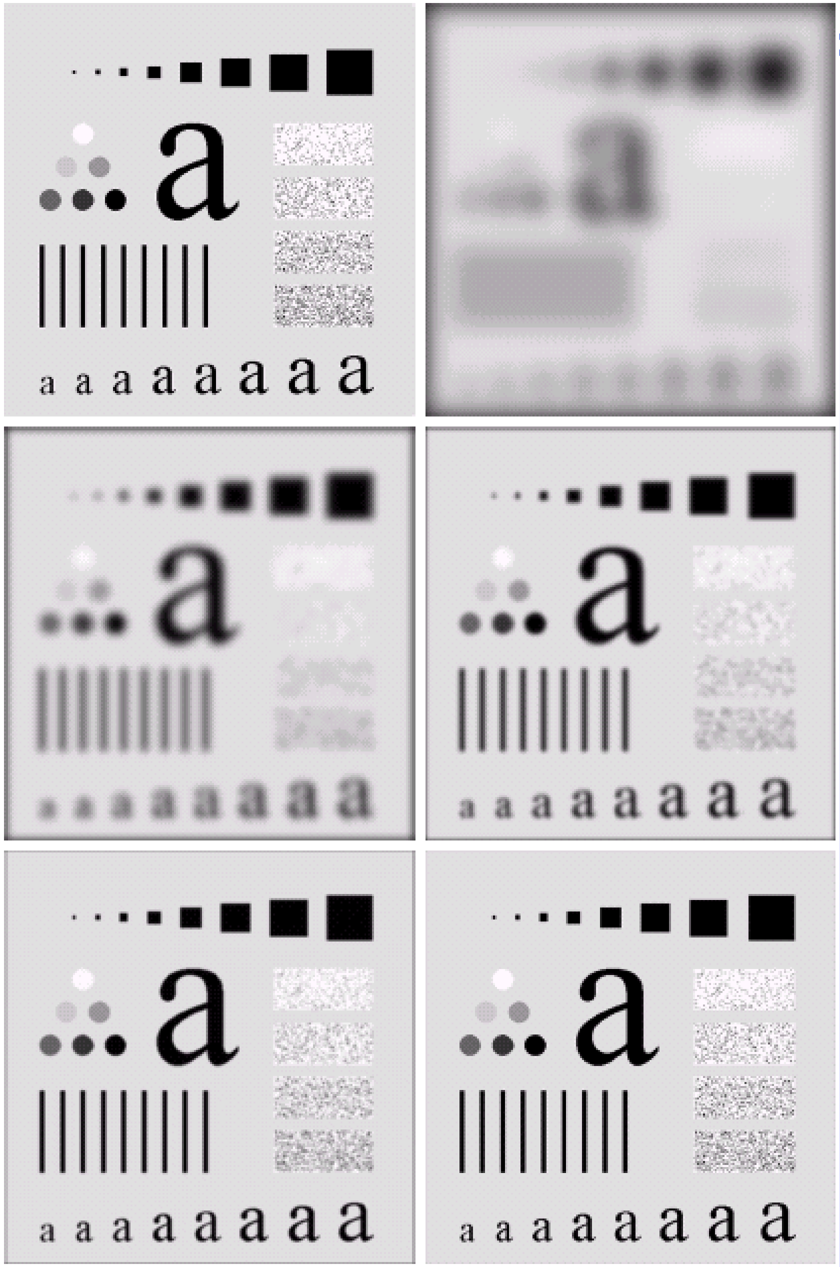


FIGURE 4.18 (a) Original image. (b)–(f) Results of filtering with Gaussian lowpass filters with cutoff frequencies set at radii values of 5, 15, 30, 80, and 230, as shown in Fig. 4.11(b). Compare with Figs. 4.12 and 4.15.

a	b
c	d
e	f

3. Image Enhancement—image smoothing

a b c

FIGURE 4.21 (a) Image showing prominent scan lines. (b) Result of using a GLPF with $D_0 = 30$. (c) Result of using a GLPF with $D_0 = 10$. (Original image courtesy of NOAA.)

3. Image Enhancement—image smoothing

FIGURE 4.20 (a) Original image (1028 × 732 pixels). (b) Result of filtering with a GLPF with $D_0 = 100$. (c) Result of filtering with a GLPF with $D_0 = 80$. Note reduction in skin fine lines in the magnified sections of (b) and (c).

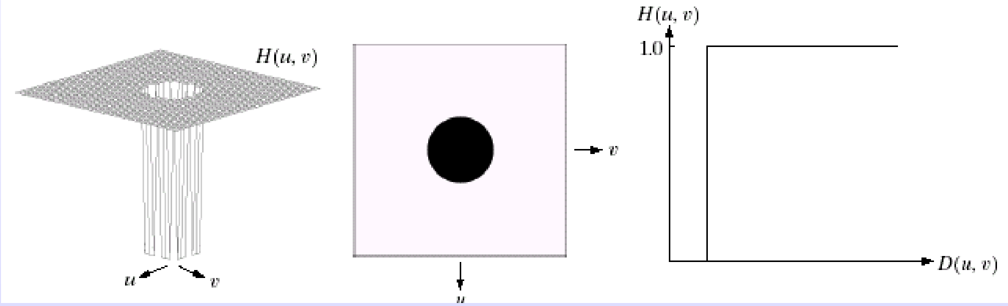
3. Image Enhancement—image sharpening

- Highpass filtering

$$H_{hp}(u,v) = 1 - H_{lr}(u,v)$$

- Ideal highpass filter

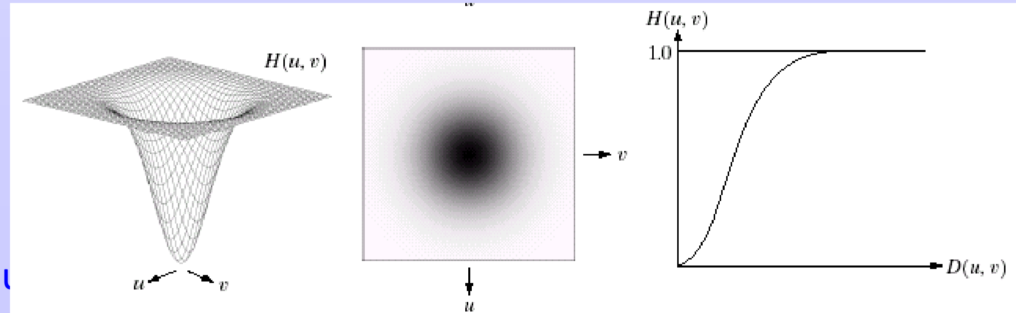
$$H(u,v) = \begin{cases} 0, & \text{if } D(u,v) \le D_0 \\ 1, & \text{if } D(u,v) > D_0 \end{cases}$$



$$D(u,v) = \sqrt{u^2 + v^2}$$

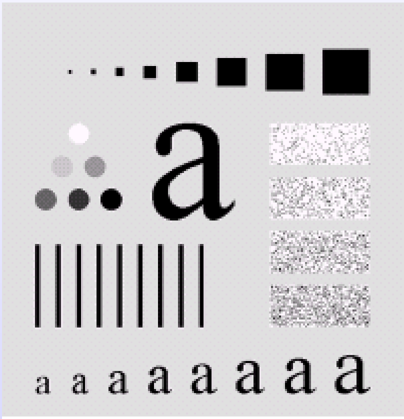
- Gaussian highpass filter (GHPF)

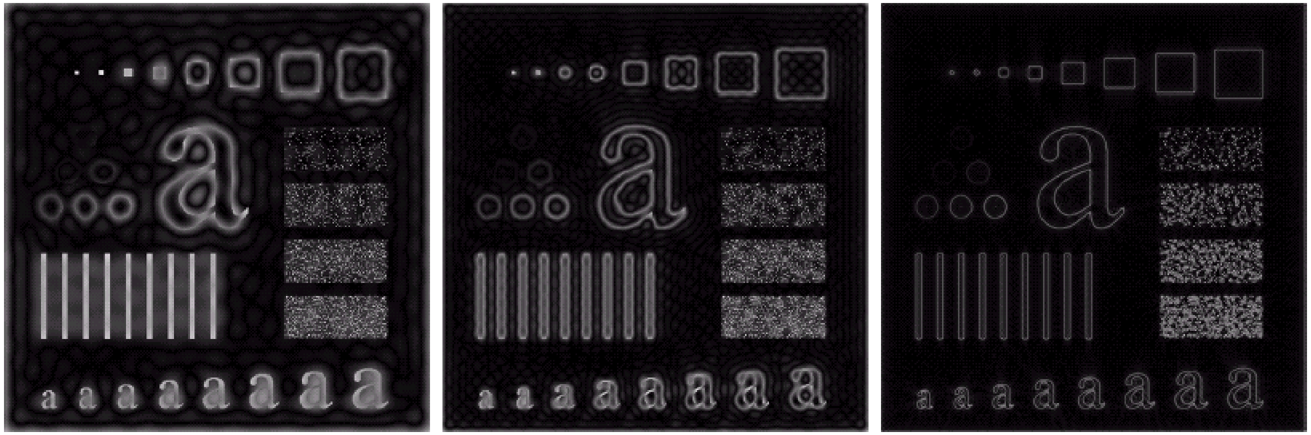
$$G(u,v) = 1 - e^{-\frac{u^2+v^2}{2D_0}}$$



3. Image Enhancement—image sharpening

Ideal highpass filter

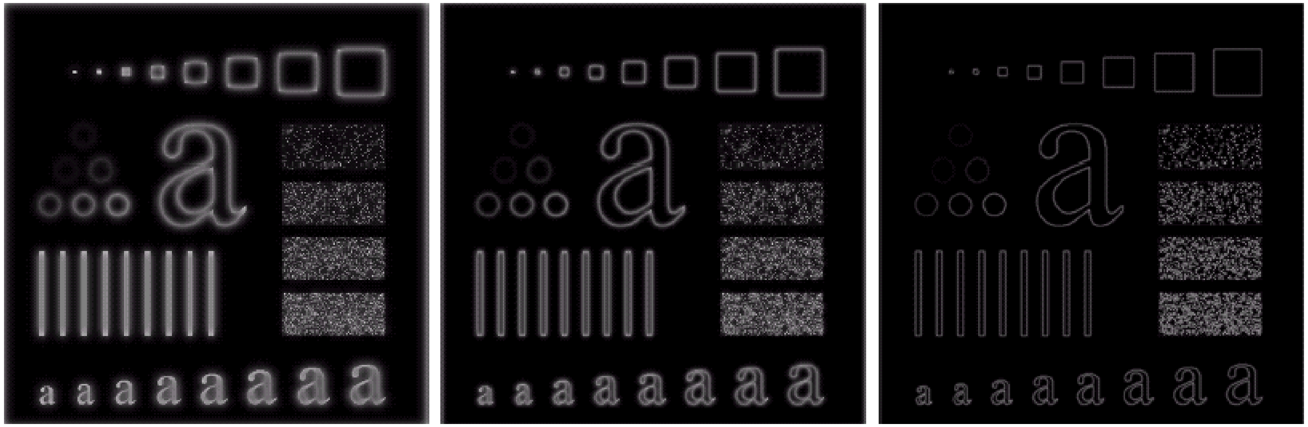




a b c

FIGURE 4.24 Results of ideal highpass filtering the image in Fig. 4.11(a) with $D_0 = 15$, 30, and 80, respectively. Problems with ringing are quite evident in (a) and (b).

Gaussian highpass filter



a b c

FIGURE 4.26 Results of highpass filtering the image of Fig. 4.11(a) using a GHPF of order 2 with $D_0 = 15$, 30, and 80, respectively. Compare with Figs. 4.24 and 4.25.

exdjiang@ntu.edu.sg

3. Image Enhancement—image sharpening

- High-boost filtering

$$f_{hb}(x, y) = Af(x, y) - f_{lp}(x, y)$$

where $f_{lp}(x, y)$ is a smoothed version of $f(x, y)$

by a lowpass filter,

$A \geq 1$

$$f_{hb}(x, y) = (A-1)f(x, y) + f(x, y) - f_{lp}(x, y)$$

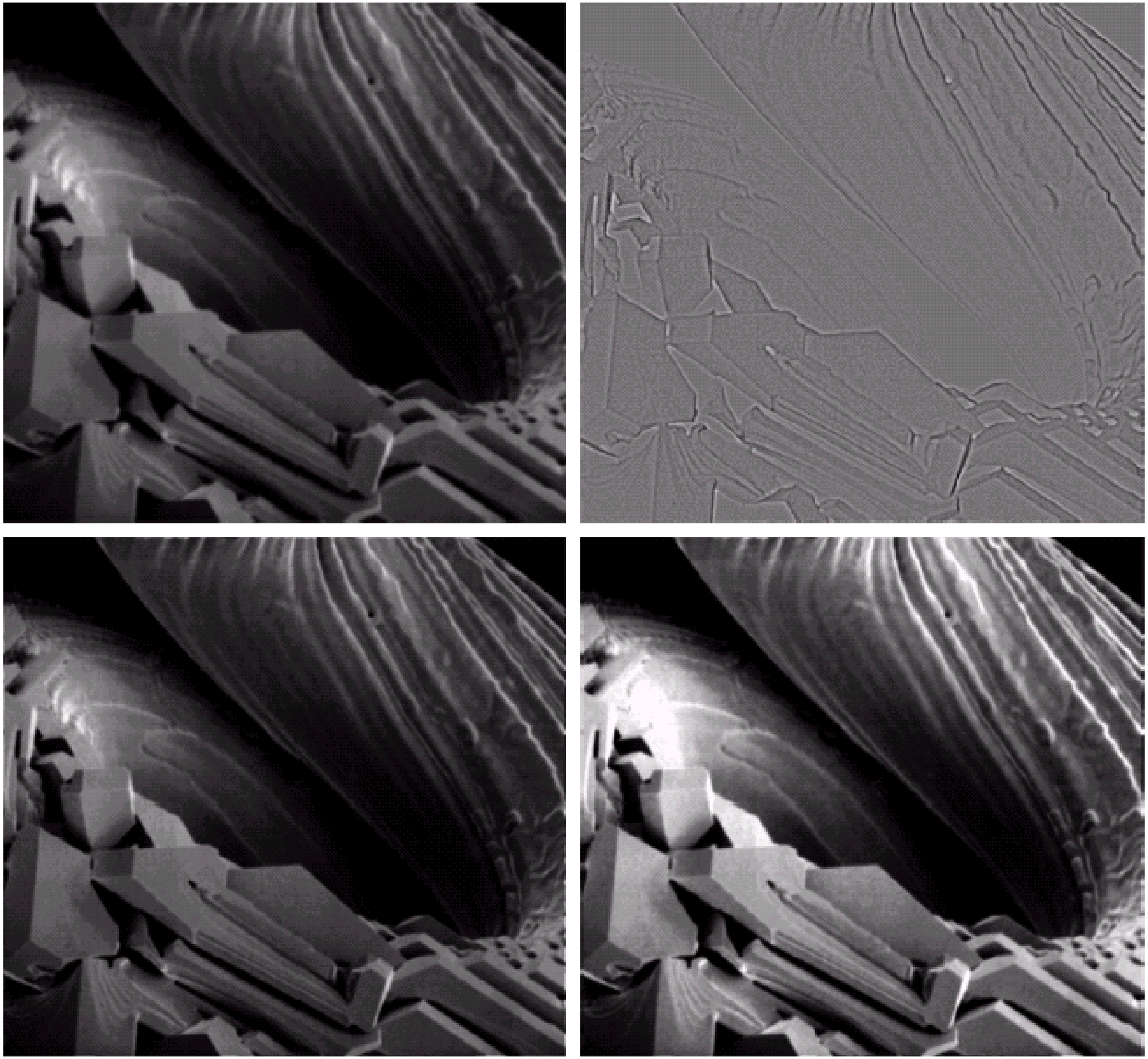
$$f_{hb}(x, y) = (A-1)f(x, y) + f_{hp}(x, y)$$

3. Image Enhancement—image sharpening

High-boost filtering

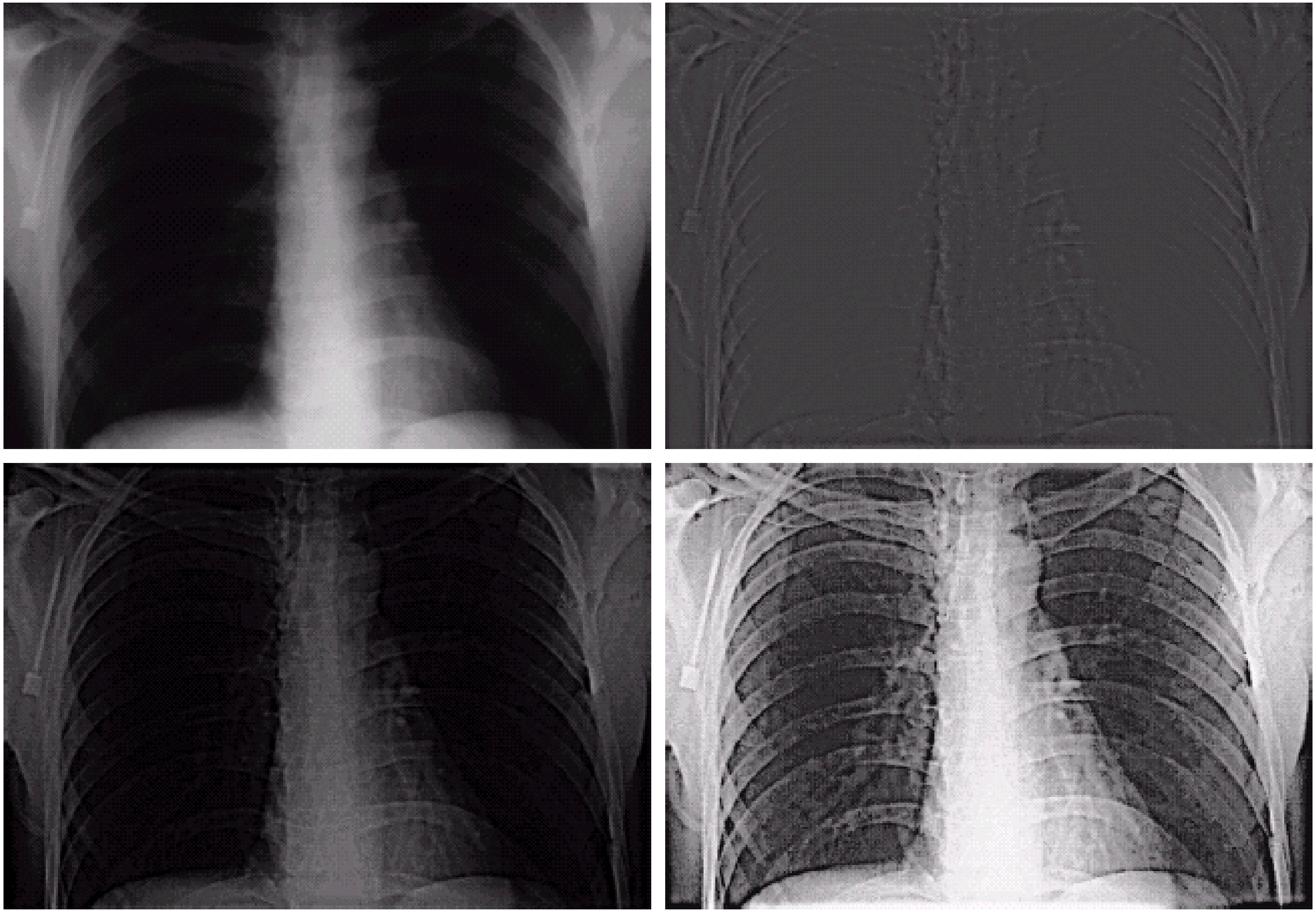
a b
c d

FIGURE 4.29 Same as Fig. 3.43, but using frequency domain filtering. (a) Input image. (b) Laplacian of (a). (c) Image obtained using Eq. (4.4-17) with $A = 2$. (d) Same as (c), but with $A = 2.7$. (Original image courtesy of Mr. Michael Shaffer, Department of Geological Sciences, University of Oregon, Eugene.)



3. Image Enhancement—image sharpening

Examples of combining image enhancement

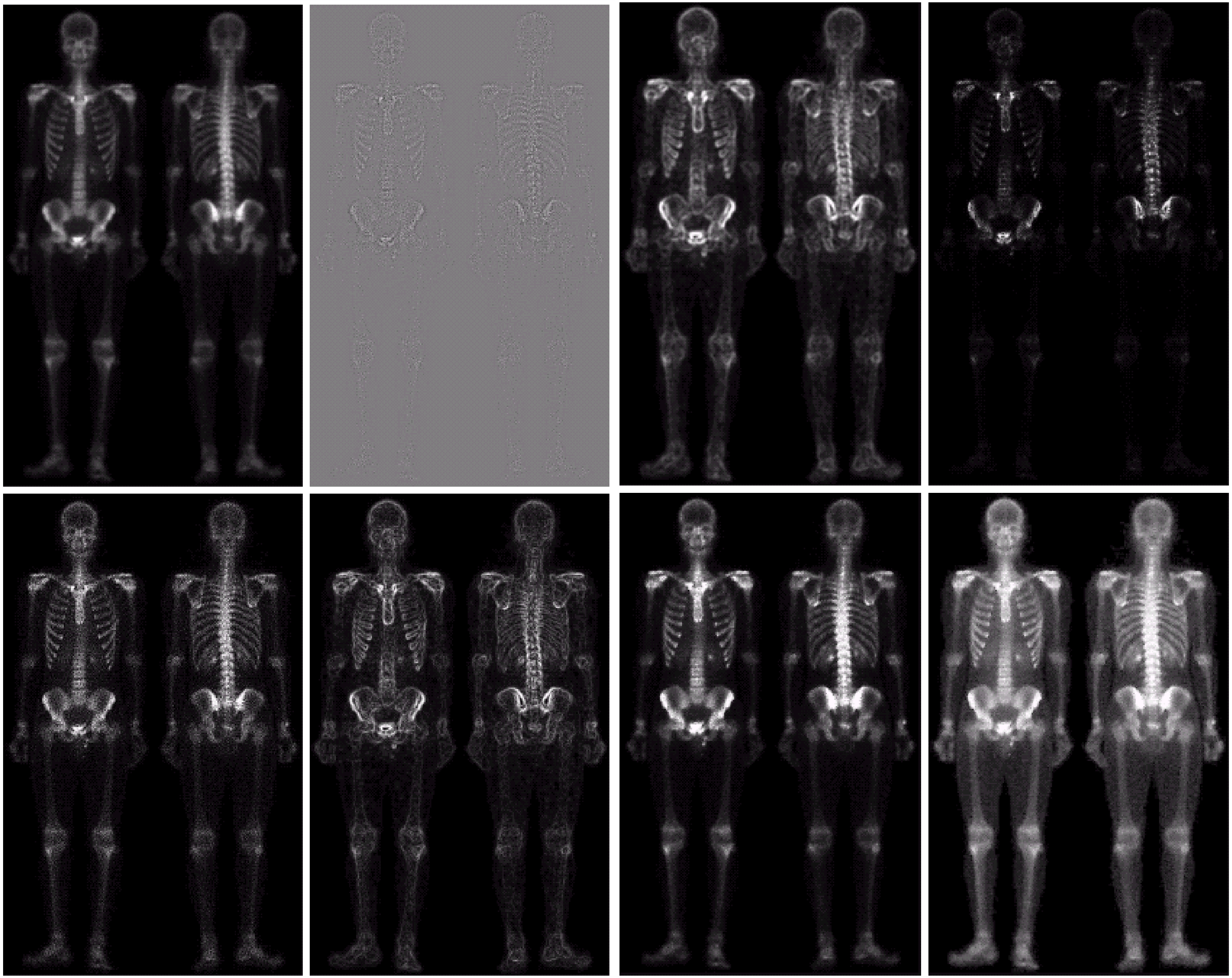


a	b
c	d

FIGURE 4.30 (a) A chest X-ray image. (b) Result of Butterworth highpass filtering. (c) Result of high-frequency emphasis filtering. (d) Result of performing histogram equalization on (c). (Original image courtesy Dr. Thomas R. Gest, Division of Anatomical Sciences, University of Michigan Medical School.)

3. Image Enhancement—image sharpening

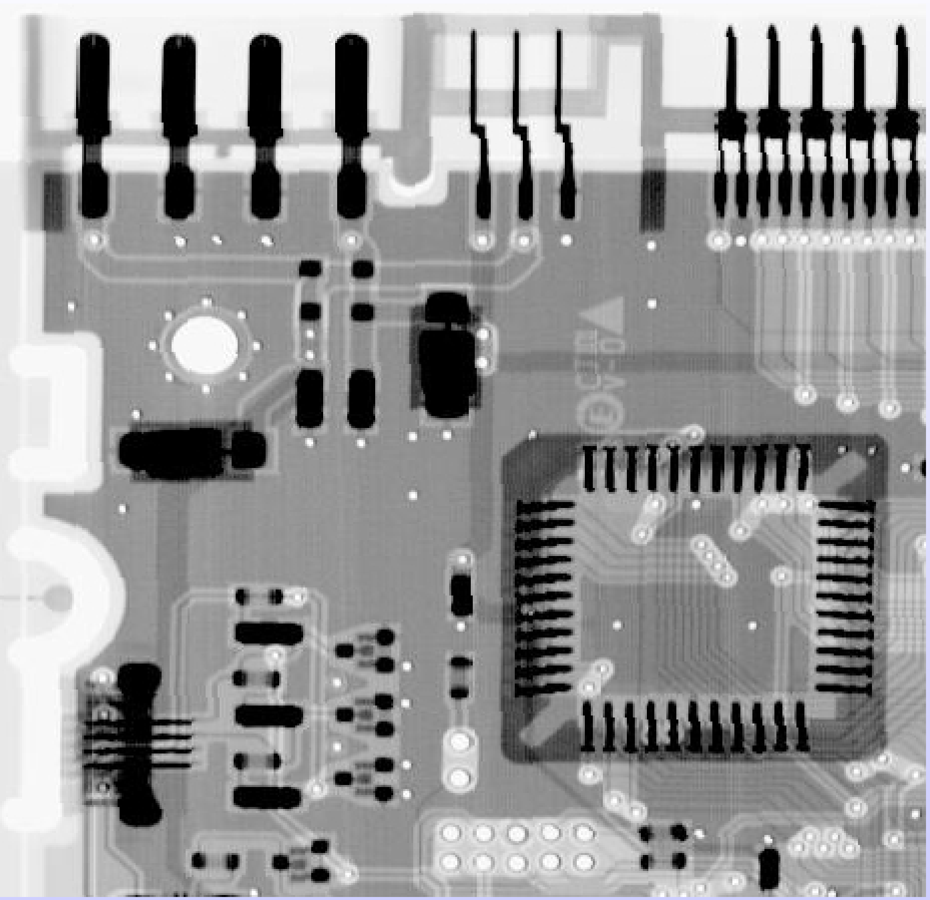
Examples of combining image enhancement

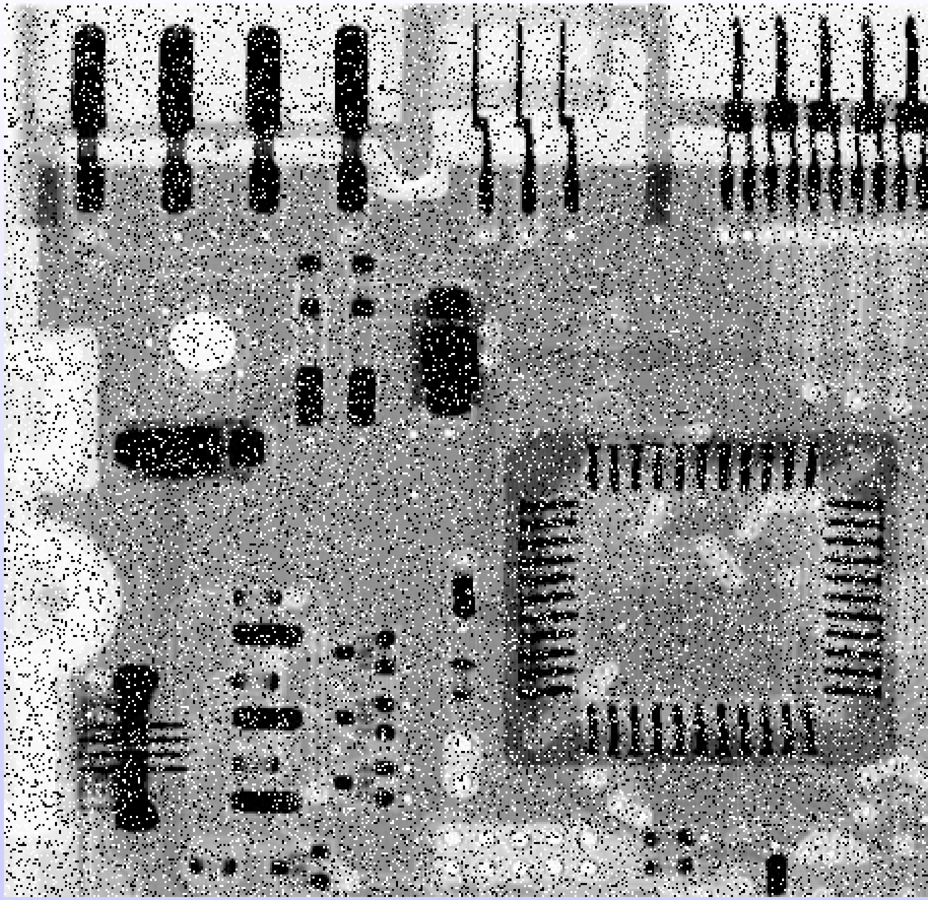


exdjiang@ntu

3. Image Enhancement—nonlinear processing

Observe an image and its noise contaminated version

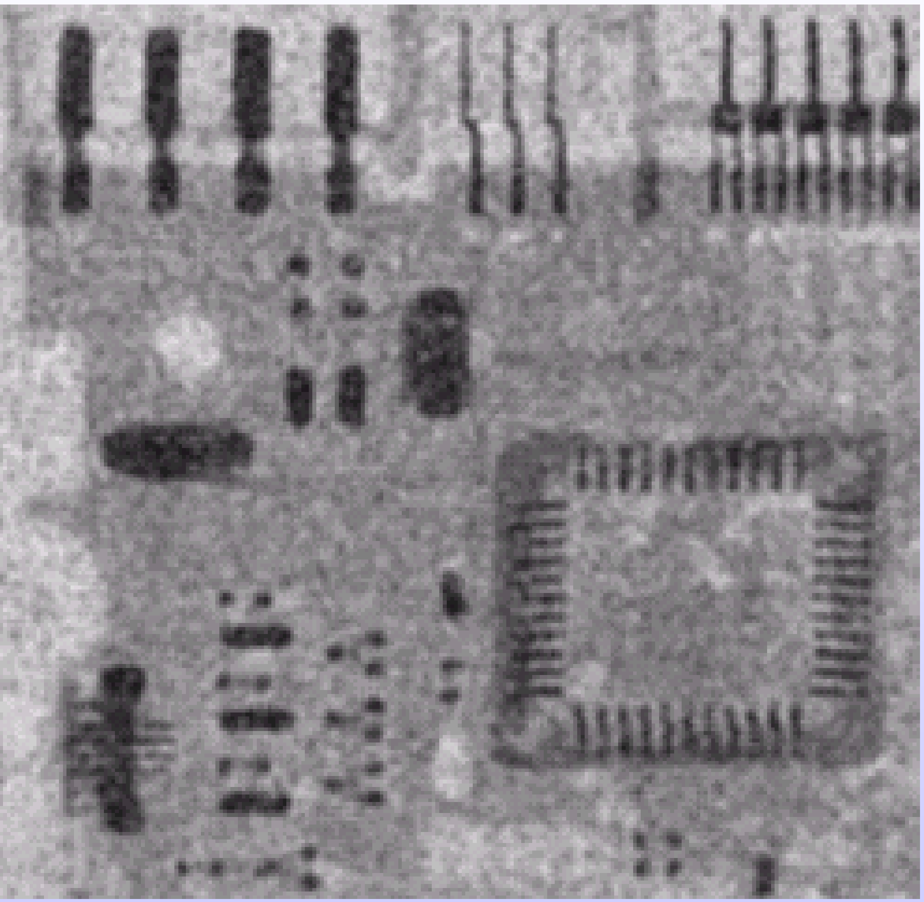


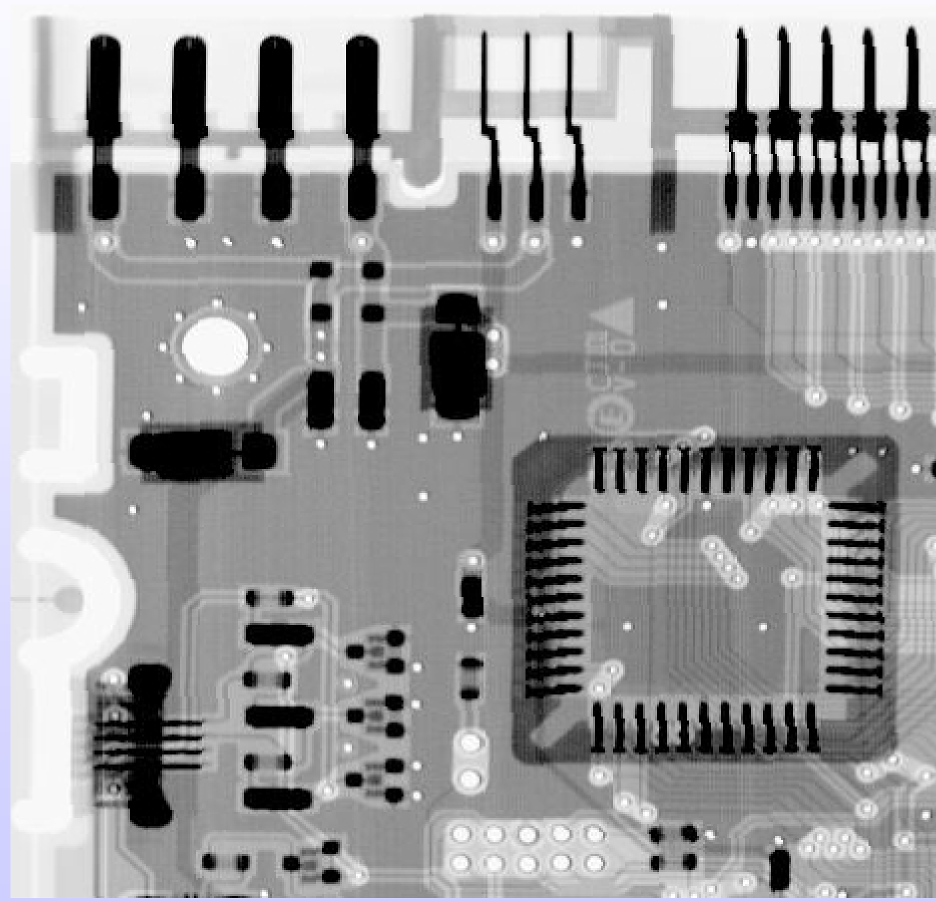


What are the noise characteristics? How to remove such noise?

3. Image Enhancement—Problems of Linear Filter

Observe the Image smoothed by a linear low pass filter

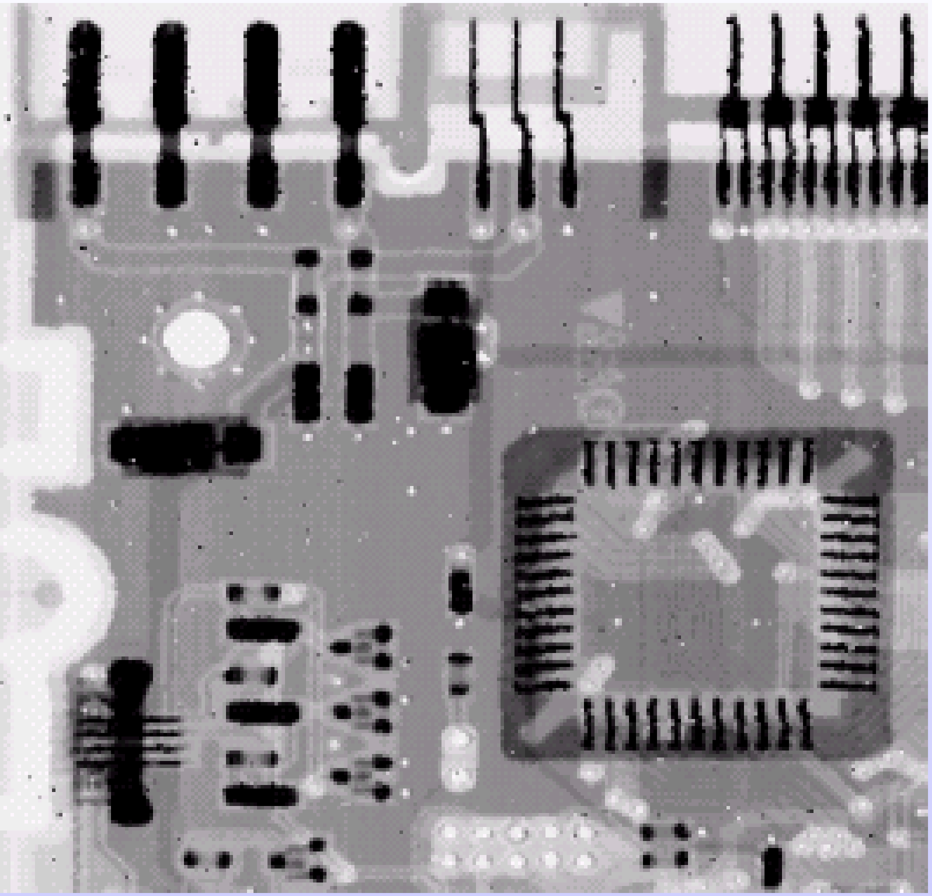


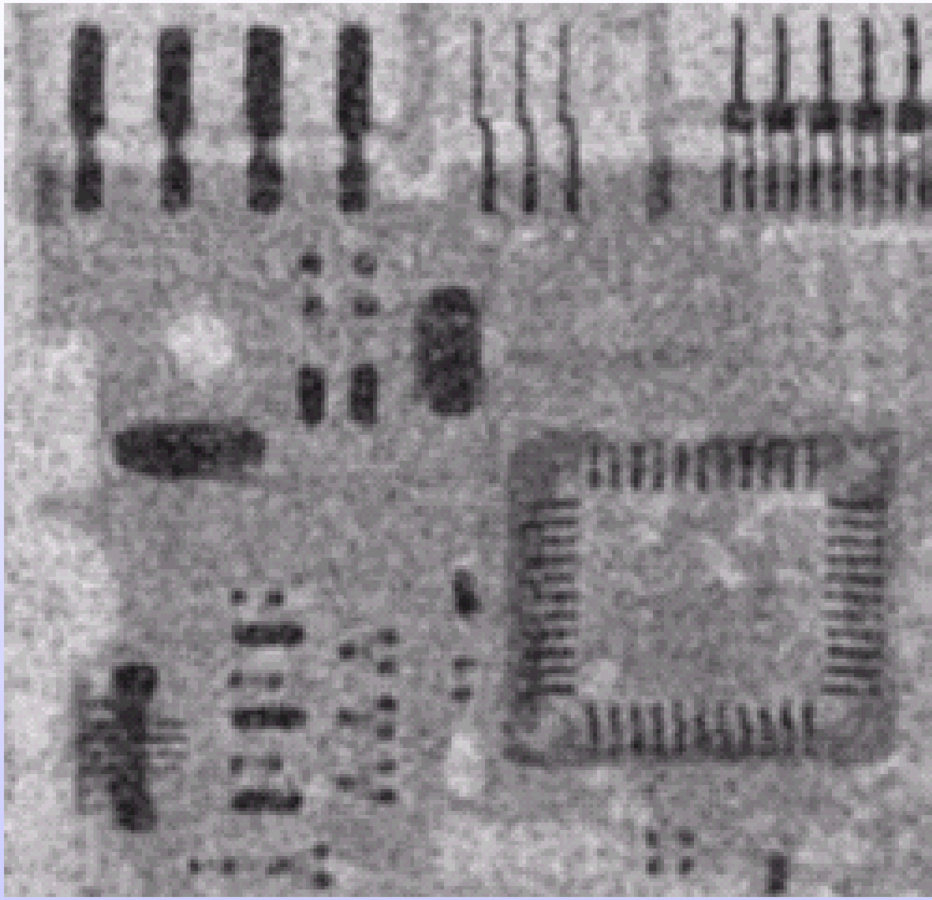


What are its problems comparing to the original image? Why?

3. Image Enhancement—Problems of Linear Filter

See another smoothed image comparing to the previous one





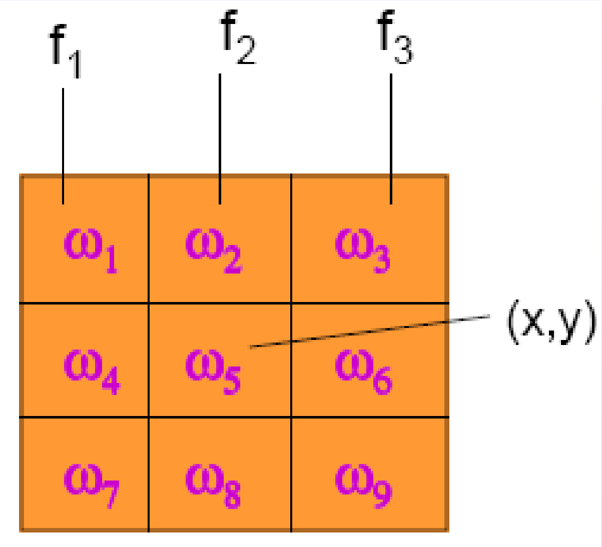
How is this smoothed image much better than the previous one?

3. Image Enhancement—Problems of Linear Filter

- Any linear filter output is a weighted average of the input pixels

$$\hat{f}(x,y) = h(x,y) * f(x,y) = \sum_{i=-a}^{a} \sum_{j=-b}^{b} h(i,j) f(x-i, y-j)$$

$$= \sum_{(s,t)\in S_{xy}} \omega(s,t) f(s,t)$$



➢ What are problems of the average of pixel grey values?

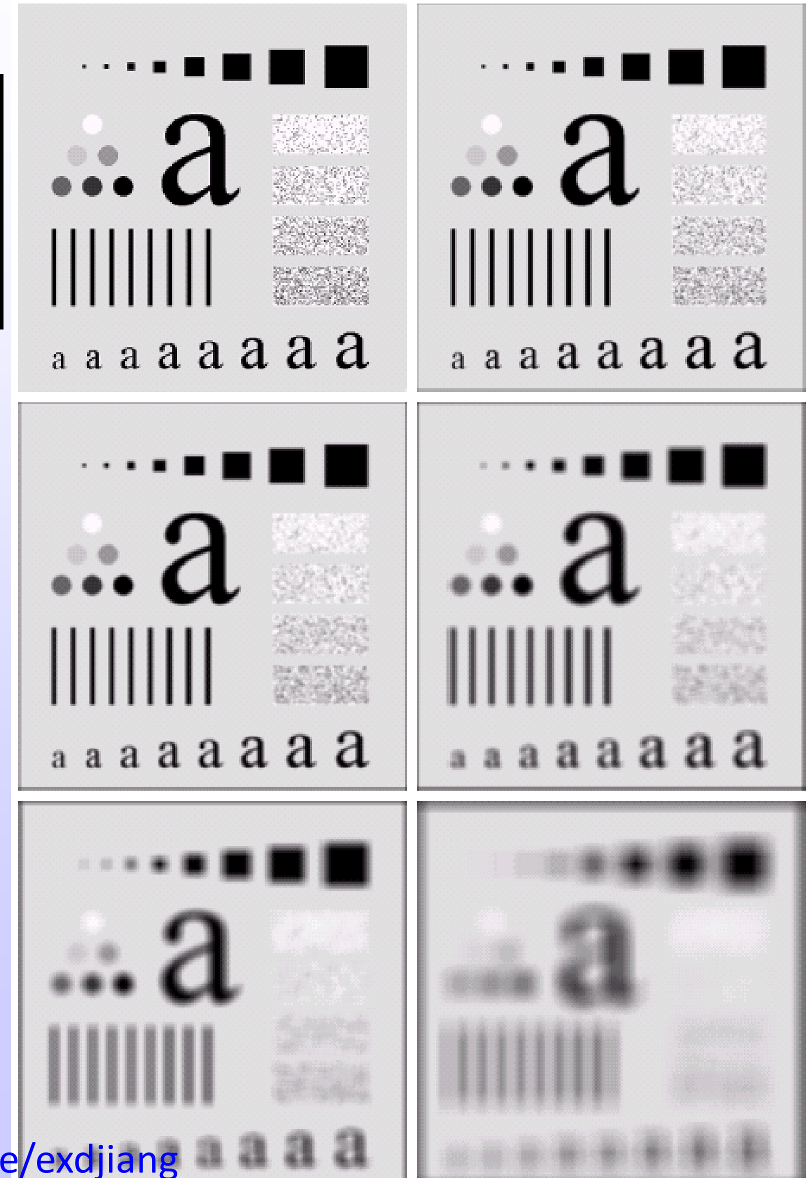
image blurring, sharpness details are lost,
difficult to smooth strong noise
Why?

3. Image Enhancement—Problems of Linear Filter

a) original image 500x500 pixel

b). - f). results of smoothing with square averaging filter masks of size nxn, n = 3, 5, 9, 15 and 35, respectively.

a	b
c	d
e	f



Note:

– The size of the mask establishes the relative size of the objects that will be blended with the background.

3. Image Enhancement—Order-Statistic Filters

- The response is based on ordering (ranking) the pixels contained in the image area encompassed by the filter.

- The best-known example is median filter, which replaces the value of a pixel by the median of the gray levels in the neighborhood of that pixel.

$$\hat{f}(x,y) = \underset{(s,t)\in S_{xy}}{median}\{f(s,t)\}$$

10	20	20
20	15	20
25	20	100

→ (10,15,20,20,20,20,20,25,100)

Median=20

So replace (15) with (20)

3. Image Enhancement—Median Filter

$$\hat{f}(x,y) = \underset{(s,t)\in S_{xy}}{median}\{f(s,t)\}$$

- Median filter forces the points with distinct gray levels to be more like their neighbors.
- Isolated clusters of pixels that are lighter or darker with respect to their neighbors, and whose area is less than $n^2/2$ (one-half the filter area), are eliminated by an n x n median filter.
- eliminated = forced to have the value equal the median intensity of the neighbors.
- Larger clusters are affected considerably less.

3. Image Enhancement—Median Filter

- Edge is a basic and significant structure of an image.

What is the outputs of a mean filter?

$$mean\{0,0,0,1,1,1,1\} = 0.57$$

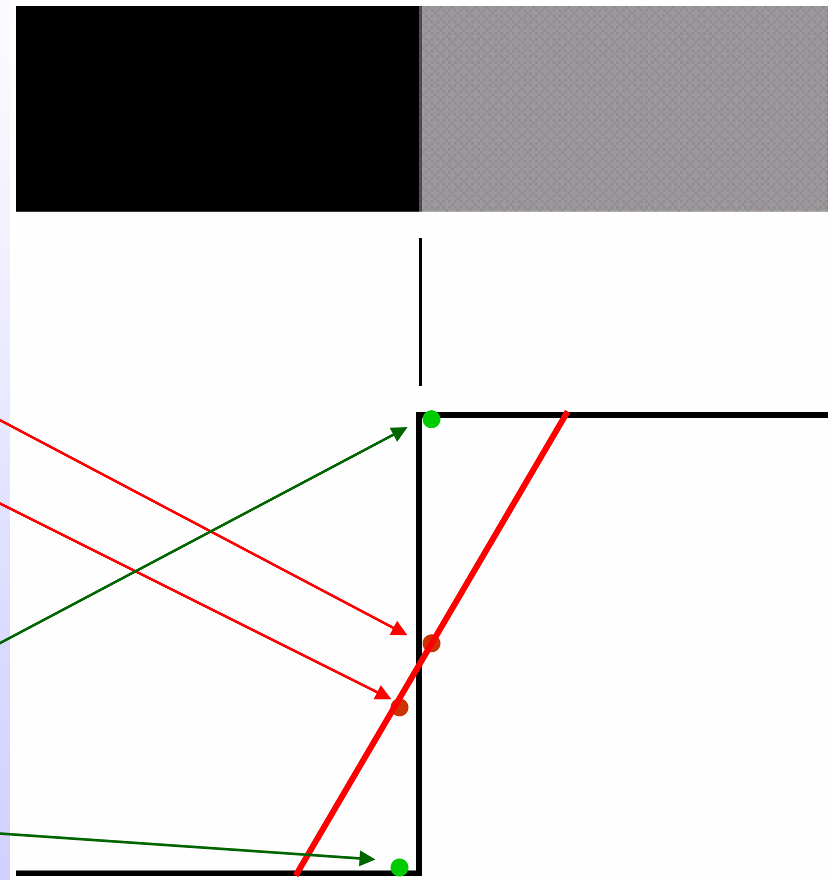
$$mean\{0,0,0,0,1,1,1,\} = 0.43$$

What is the outputs of a median filter?

$$median\{0,0,0,1,1,1,1\} = 1$$

$$median\{0,0,0,0,1,1,1,\} = 0$$

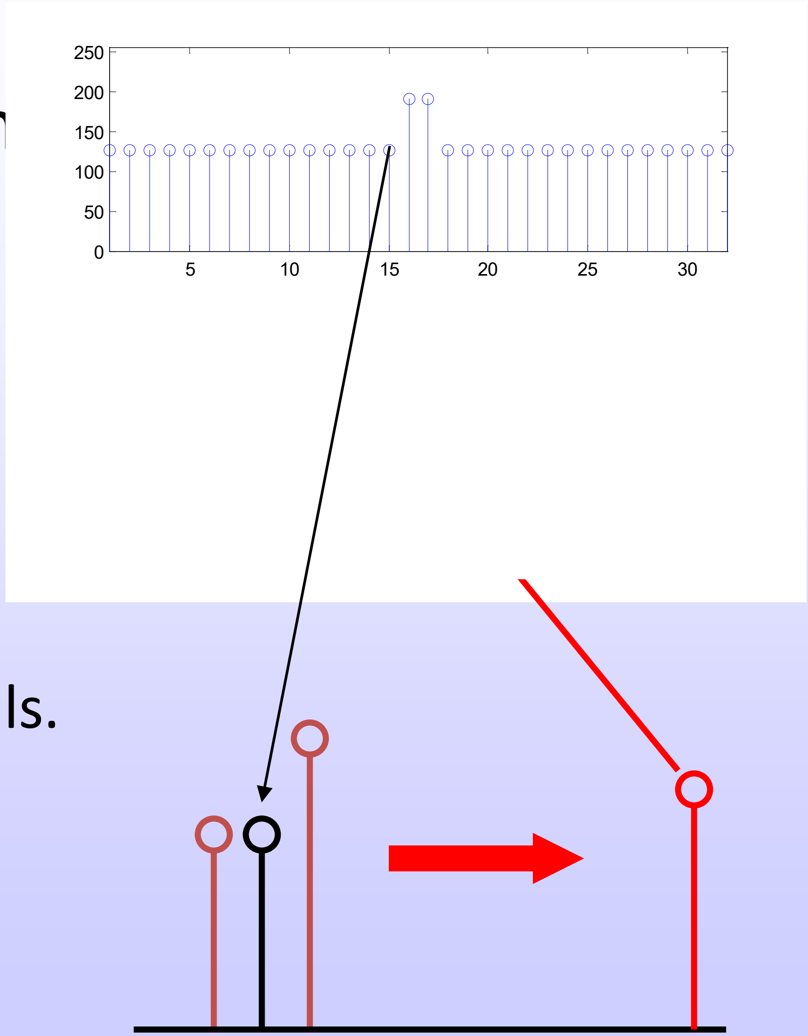
Model of an ideal digital edge



Gray-level profile of a horizontal line through the image

3. Image Enhancement—Mean vs. Median Filter

- Consider a uniform 1-D image with a pulse function
- Pulse function corresponds to fine image detail such as lines and curves.



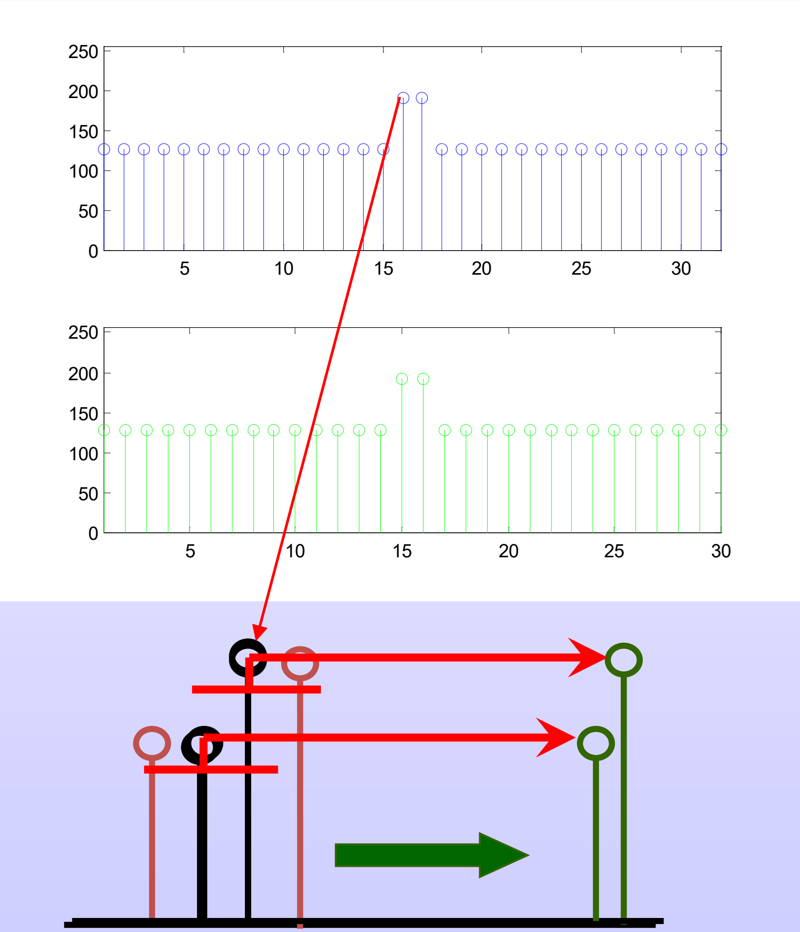
- ➢ Mean filter ‘blurs’ the image details.
- ➢ If the pulse is noise, mean filter suppress it only for some extent but spread the noise.

3. Image Enhancement—Mean vs. Median Filter

- Consider a uniform 1-D image with a pulse function.

- Pulse function corresponds to fine image detail such as lines and curves.

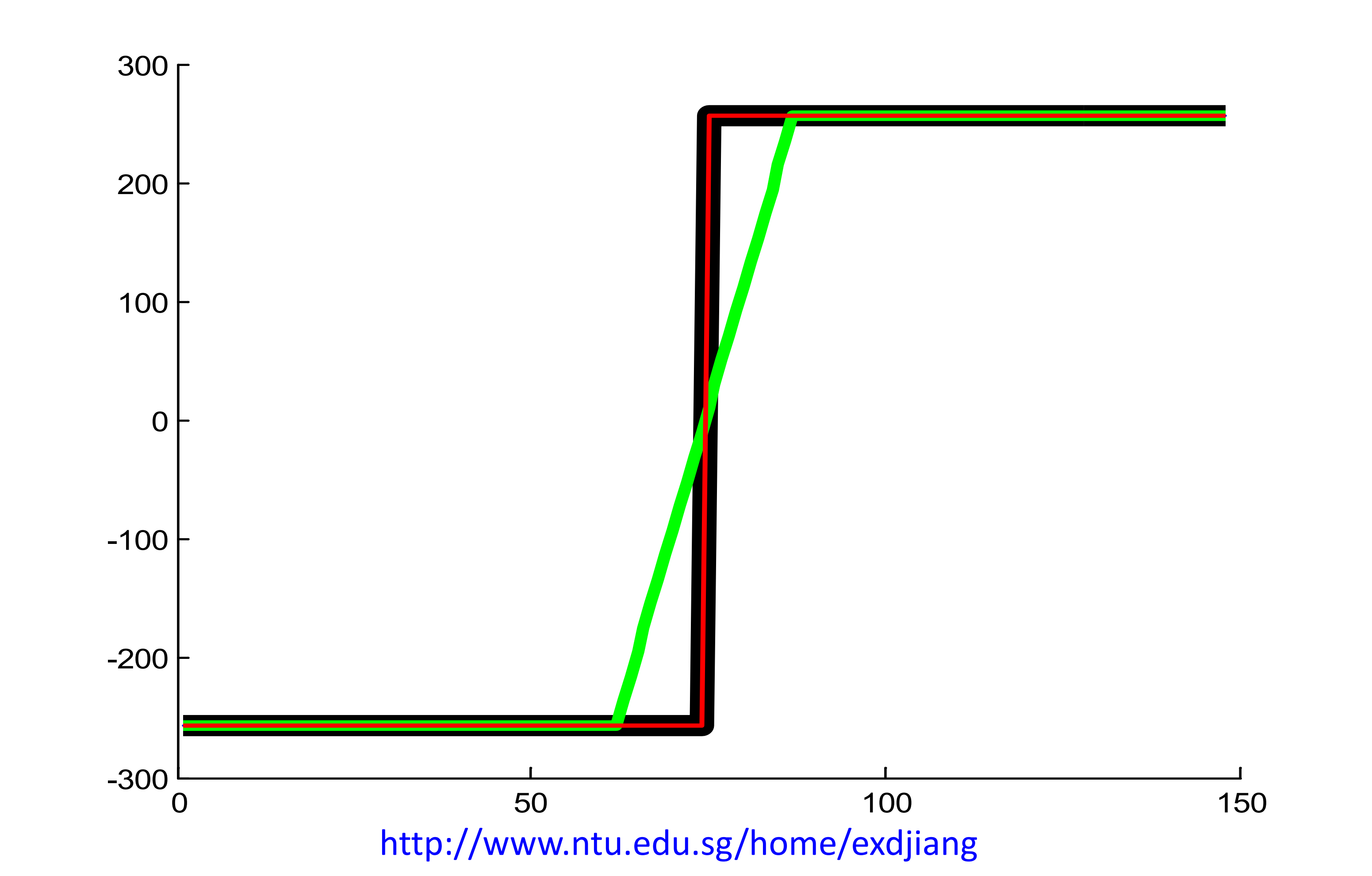
- Median filter does not 'blur' the edge.



➢ If the pulse is noise, 5X5 median filter totally remove such noise.

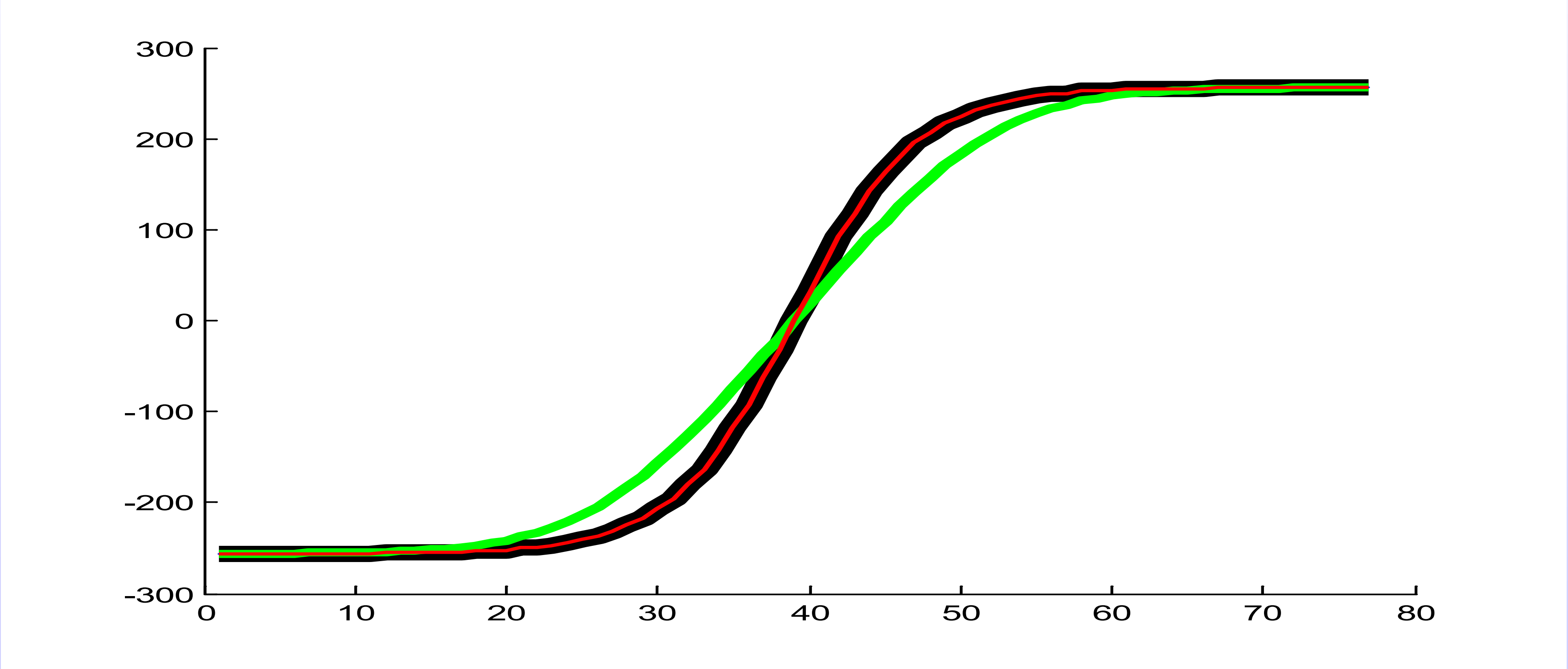
3. Image Enhancement—Mean vs. Median Filter

A simple MATLAB program can show: Mean filter blurs the step edge. Median filter preserves the step edge.



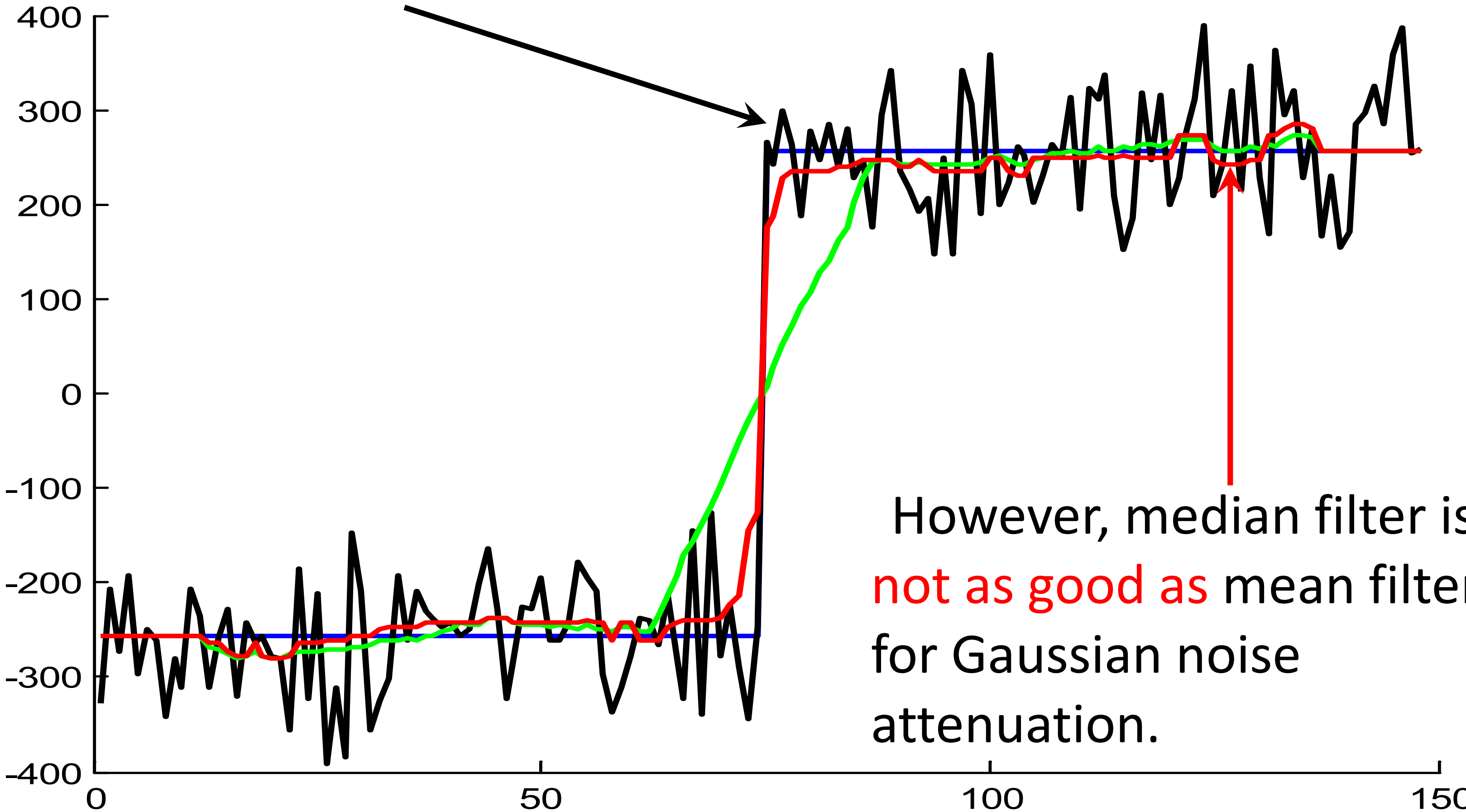
3. Image Enhancement—Mean vs. Median Filter

A simple MATLAB program can show: Mean filter further blurs the smooth edge. Median filter preserves the smooth edge.



3. Image Enhancement—Mean vs. Median Filter

A simple MATLAB program can show: Mean filter attenuates additive Gaussian noise but blurs the edge. Median filter attenuates Gaussian noise and preserves the edge.

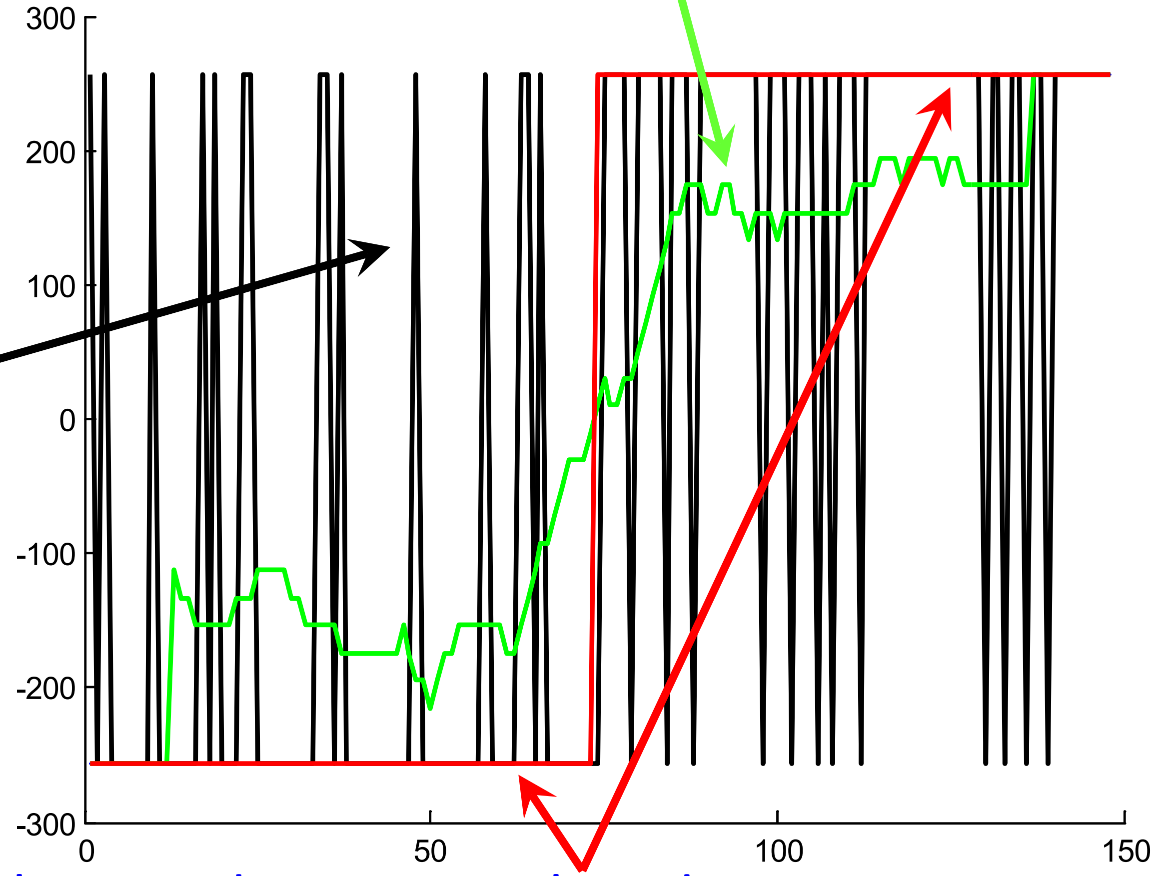


However, median filter is not as good as mean filter for Gaussian noise attenuation.

3. Image Enhancement—Mean vs. Median Filter

A simple MATLAB program can show: Mean filter is ineffective to attenuate impulsive noise and blurs the edge.

Impulsive noise is strong in amplitude and spatial sparse



Median filter provides excellent noise-reduction capabilities and preserves the edge.

3. Image Enhancement—Mean vs. Median Filter

Original and noise corrupted images

impulse noise ⇨ salt and pepper noise.

3. Image Enhancement—Mean vs. Median Filter

Example outputs of

mean filter and median filter.

3 Nonlinear Image Smoothing–Med. Filter Properties

- Linear filter has established theory to analyze its properties, especially in the frequency domain.
- However, It is difficult to analyze Median filter and other order-statistic filters due to their nonlinearity.

➢ Repeated applications of median filter to a signal result in an invariant signal called the "root signal". A root signal is invariant to further application of the median filter.

➢ Example: 1-D signal: Median filter length = 3

0 0 0 1 2 1 2 1 2 1 0 0 0

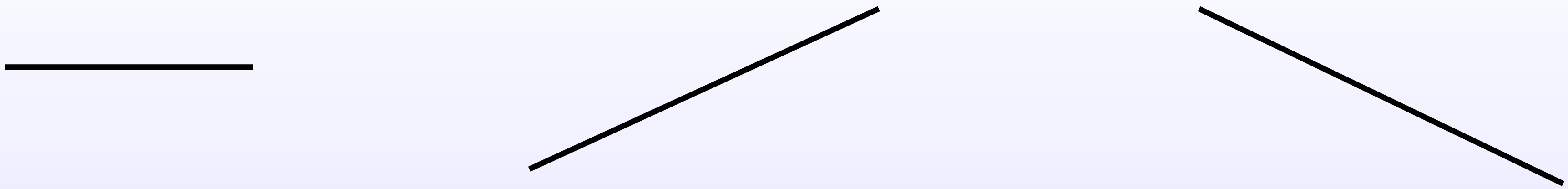
0 0 0 1 1 2 1 2 1 1 0 0 0

0 0 0 1 1 1 2 1 1 1 0 0 0

0 0 0 1 1 1 1 1 1 1 0 0 0 ~ root signal

3. Image Enhancement—Med. Filter Properties

- Invariant signals to a median filter:



Constant

Monotonically

increasing decreasing

length?

3. Image Enhancement—Other Order-stati. Filter

- Simple extension of the median filter

  - Max filter $$\hat{f}(x,y) = \max_{(s,t)\in S_{xy}} \{f(s,t)\}$$

  - Min filter $$\hat{f}(x,y) = \min_{(s,t)\in S_{xy}} \{f(s,t)\}$$

  - Midpoint filter

$$\hat{f}(x,y) = \frac{1}{2}\left[\max_{(s,t)\in S_{xy}} \{f(s,t)\} + \min_{(s,t)\in S_{xy}} \{f(s,t)\}\right]$$

3. Image Enhancement—Limitation and Solution

- Although Median filter preserves image edges, it removes image details such as corner, thin lines / curves and other fine details.

- How to design a rank order filter that can effectively removes impulsive noise and preserves these image details at the same time?

- The research work on this topic can be found in the research publication:

  X.D. Jiang, "Image Detail-Preserving Filter for Impulsive Noise Attenuation," *IEE Proceedings: Vision, Image and Signal Processing*, Vol. 150, No. 3, pp. 179-185, June 2003.

3. Image Enhancement—Other Order-stati. Filter

- As median filter underperforms mean filter in attenuating short-tailed noise, e.g. Gaussian noise, filters that own merits of the both mean and median filters have been developed:
- Alpha-trimmed mean filter $$\hat{f}(x,y)=\frac{1}{mn-d}\sum_{(s,t)\in S_{xy}} f_r(s,t)$$

where $f_r(s,t)$ are the remaining $mn-d$ pixels around median.

- Iterative Truncated Arithmetic Mean Filter

X.D. Jiang, "Iterative Truncated Arithmetic Mean Filter And Its Properties," ***IEEE Transactions on Image Processing***, vol. 21, no. 4, PP. 1537-1547, April 2012.

Z. Miao and X.D. Jiang, "Further Properties and a Fast Realization of the Iterative Truncated Arithmetic Mean Filter" ***IEEE Transactions on Circuits and Systems-II***, vol. 59, no. 11, pp. 810-814, November 2012.

Z. Miao and X.D. Jiang, "Weighted Iterative Truncated Mean Filter," ***IEEE Transactions on Signal Processing***, Vol. 61, no. 16, pp. 4149-4160, August, 2013.

Z. Miao and X.D. Jiang, "Additive and exclusive noise suppression by iterative trimmed and truncated mean algorithm," ***Signal Processing***, vol. 99, pp. 147-158, June, 2014.

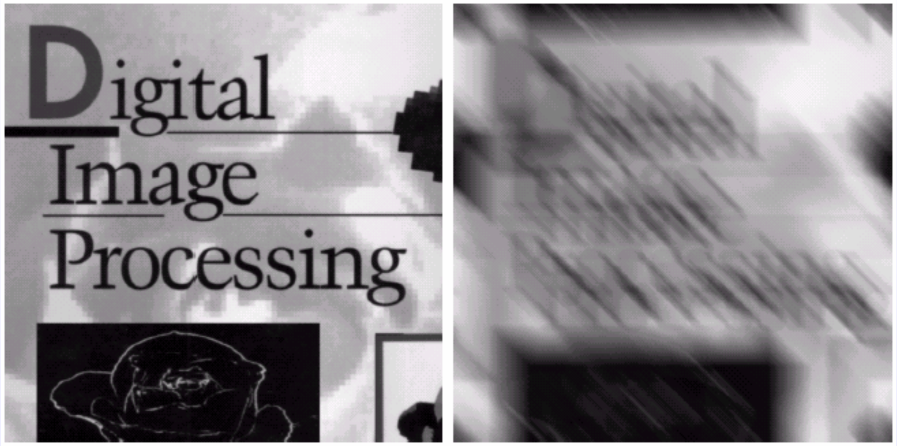
4 Image Restoration –Outline

- Introduction
- Noise and Degradation Models
- Inverse Filtering
- Wiener Filtering
- Frequency Domain Filtering

4 Image Restoration –Introduction

- Look at an image and its degraded version. What technique you learnt can recover the original image from the degraded one?

  low pass filter?
  high pass filter?
  band pass filter?
  or histogram equalization?



➢ None of these techniques can well recover the image. Why?

➢ We can only recover the original image well with priori knowledge of the degradation.

➢ This is the task of image restoration.

4 Image Restoration –Introduction

- What is the Image Restoration?
  - Reconstruct or recover a degraded image using priori knowledge of the degradation
  - It is an objective process and modeling oriented

- Therefore, most restoration techniques assume some knowledge of the degradation process.

- Different from image enhancement that is a subjective process that takes heuristic procedures for human visual system or for easy computer manipulation

4 Image Restoration –Introduction

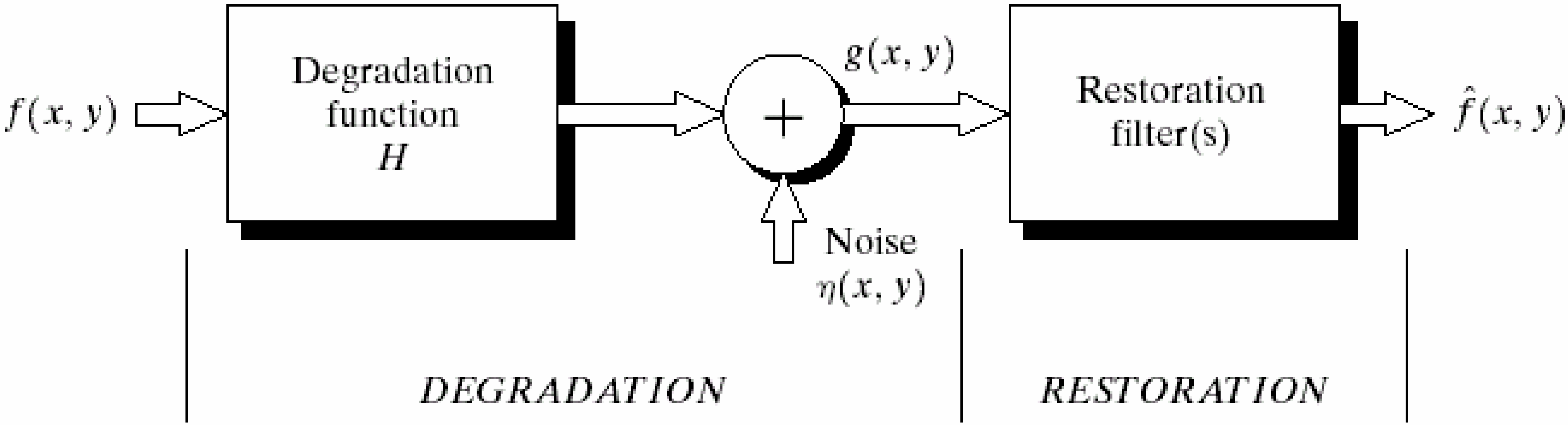
- An image may be degraded by
  - camera noise
  - motion blur
  - defocus blur
  - Transmission disturbance
  - and so forth.

- Processing/recognition on a degraded image will perform poorly.

- Therefore, it is necessary to restore the image before further processing.

4 Image Restoration –Introduction

- A Model of the Image Degradation / Restoration Process



$$f(x, y) \rightarrow \text{original image}$$

$$H \rightarrow \text{degredation function}$$

$$\eta(x, y) \rightarrow \text{additive noise}$$

$$g(x, y) \rightarrow \text{degraded (observed) image}$$

$$\hat{f}(x, y) \rightarrow \text{estimated (restored) image}$$

4 Image Restoration –Degradation Models

- A simple degradation model assumes the degradation process to be an LTI / LSI system.
- A simple noise model assumes additive uncorrelated noise.

$$g(x,y)=h(x,y)*f(x,y)+\eta(x,y)=\sum_{i=-\infty}^{\infty}\sum_{j=-\infty}^{\infty}f(i,j)h(x-i,y-j)+\eta(x,y)$$

$$\Updownarrow \ \text{(Fourier Transform)}$$

$$G(u,v)=H(u,v)F(u,v)+N(u,v)$$

- Assume the impulse response (point spread function) of the degradation process is finite in space,

$$g(x,y)=\sum_{i=-I}^{I}\sum_{j=-J}^{J}h(i,j)f(x-i,y-j)+\eta(x,y)$$

4 Image Restoration –Degradation Models

- **1) Motion blur** due to relative motion between camera and object (moving camera or moving object)

For example, if an image undergoes planar motion in x- and y-directions with $x_o(t)$ and $y_o(t)$ and T is the duration of the exposure, we have

$$g(x,y)=\int_0^T f[x-x_o(t),y-y_o(t)]dt$$

Suppose: $x_o(t)=at/T, y_o(t)=bt/T,$

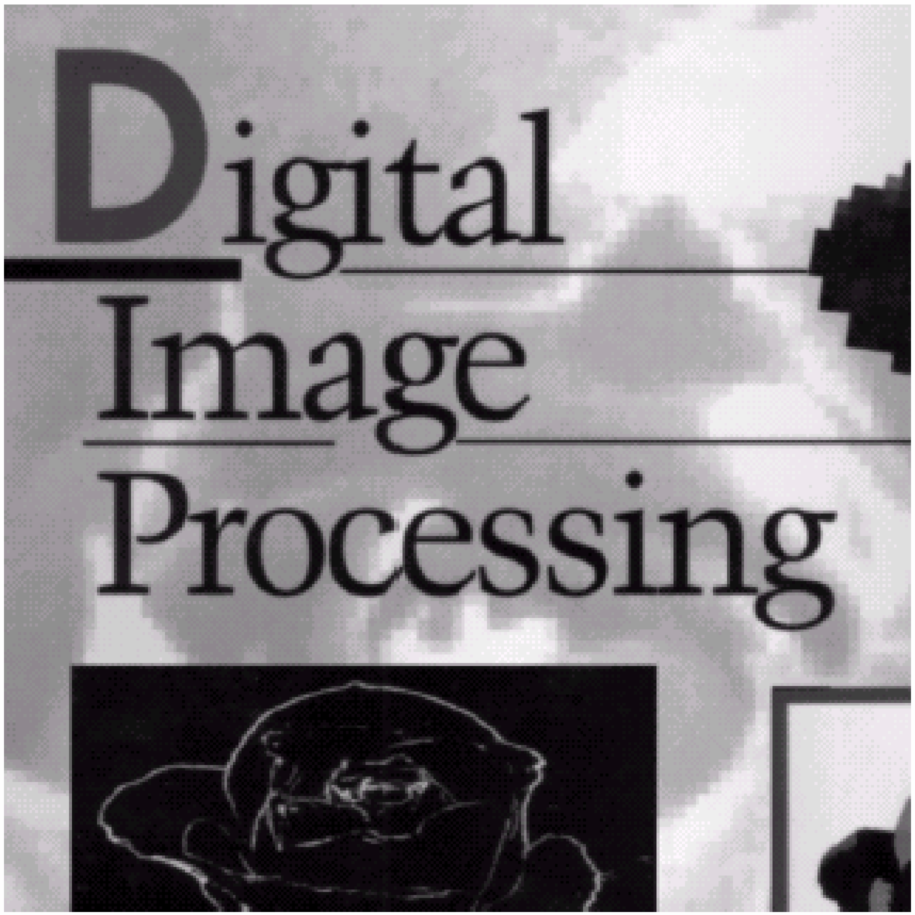
We get the degradation function in Fourier domain

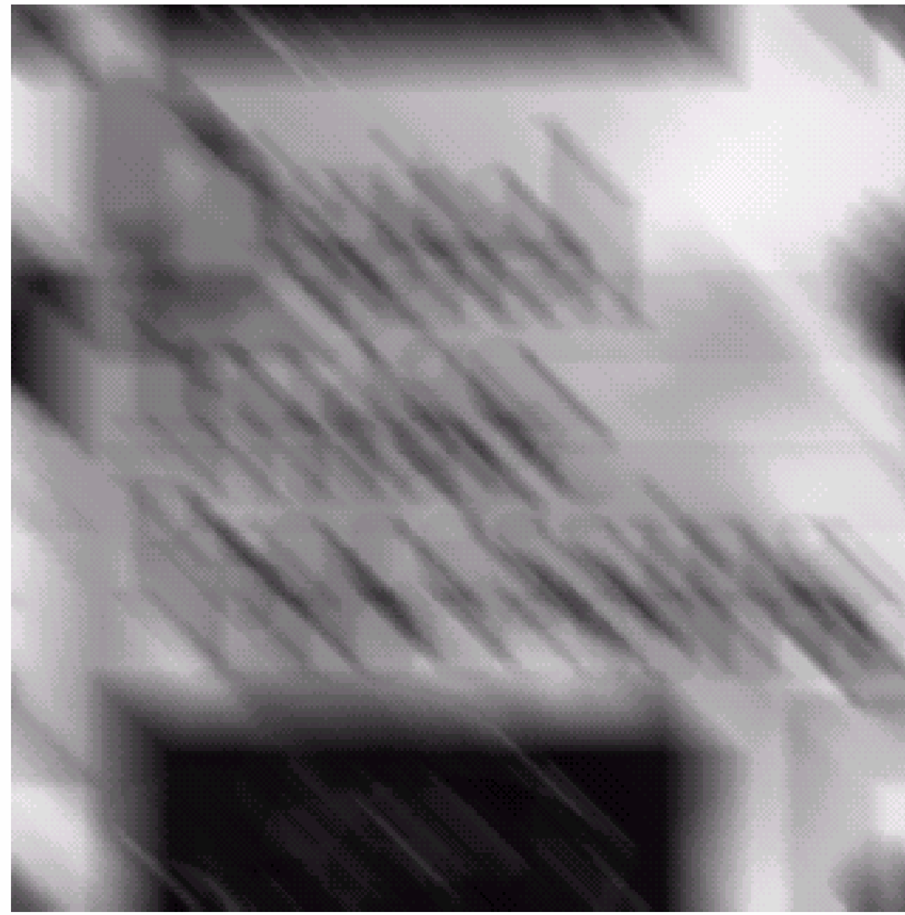
$$H(u,v)=\frac{G(u,v)}{F(u,v)}=\frac{T\sin[\pi(ua+vb)]}{\pi(ua+vb)}e^{-j\pi(ua+vb)}$$

Frequency response is a directional sinc function.

4 Image Restoration –Degradation Models

- Original image and motion blurred image with a=b=0.1, T=1





4 Image Restoration –Degradation Models

- **2) Rectangular aperture of the camera**

$$h(x,y)=\begin{cases}1, & -a\le x\le a, -b\le y\le b\\ 0, & \text{else}\end{cases}$$

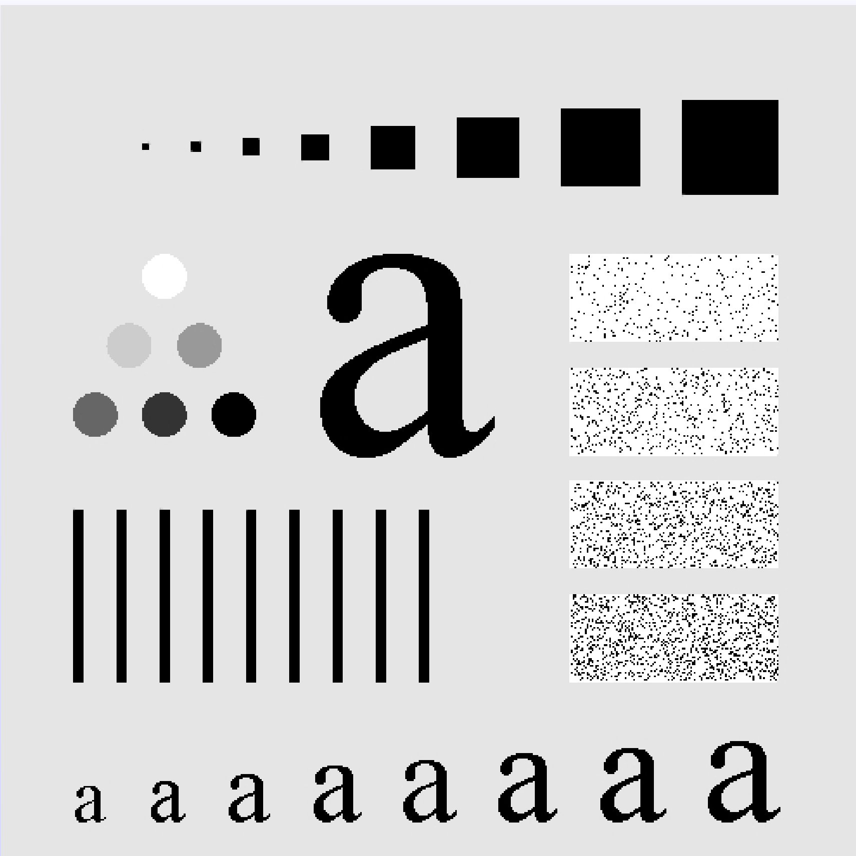
a and b depend on the aperture dimensions

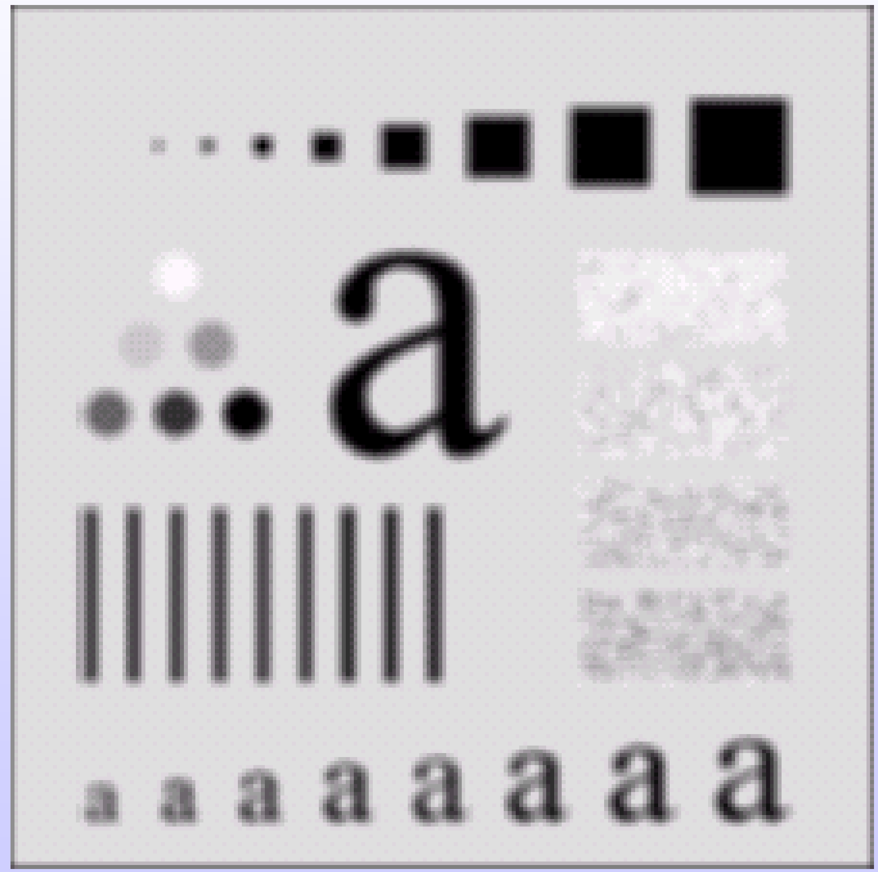
For example, **out-of-focus** means large aperture. So the above gives out-of-focus blur, and the more out-of-focus the image is, the larger the values of a and b.

- The frequency response will be the product of a horizontal and a vertical sinc function.

4 Image Restoration –Degradation Models

- Original image and out-of-focus blurred image with $a=b=4$





4 Image Restoration –Degradation Models

- **3) Atmospheric turbulence** due to random variation in the atmosphere / air between the camera and the object (for satellite / aerial / astronomical images)

➢ The frequency response is a 2-D Gaussian function (with circular contours).

$$H(u,v) = e^{-k(u^2+v^2)^{5/6}}$$

k is the turbulence parameter

4 Image Restoration –Degradation Models

- Images with negligible Atmospheric turbulence and severe Atmospheric turbulence k=0.0025


4 Image Restoration –Noise Models

Typically, noise $\eta(x, y)$ is modeled as

- zero-mean: $E\{\eta(x, y)\} = 0$
- Independent to original image: $E\{\eta(x, y) f(x+i, y+j)\} = 0$
- Gaussian: probability density function (PDF) of $\eta(x, y)$ is Gaussian
- White noise: power spectral density (PSD) of $\eta(x, y)$ is flat and constant. This means the autocorrelation is an impulse

$$E\{\eta(x, y)\eta(x+i, y+j)\} = \delta(i, j)$$

Noise of different pixels is thus uncorrelated.

- Some images, however, have a signal-dependent or periodic noise component.

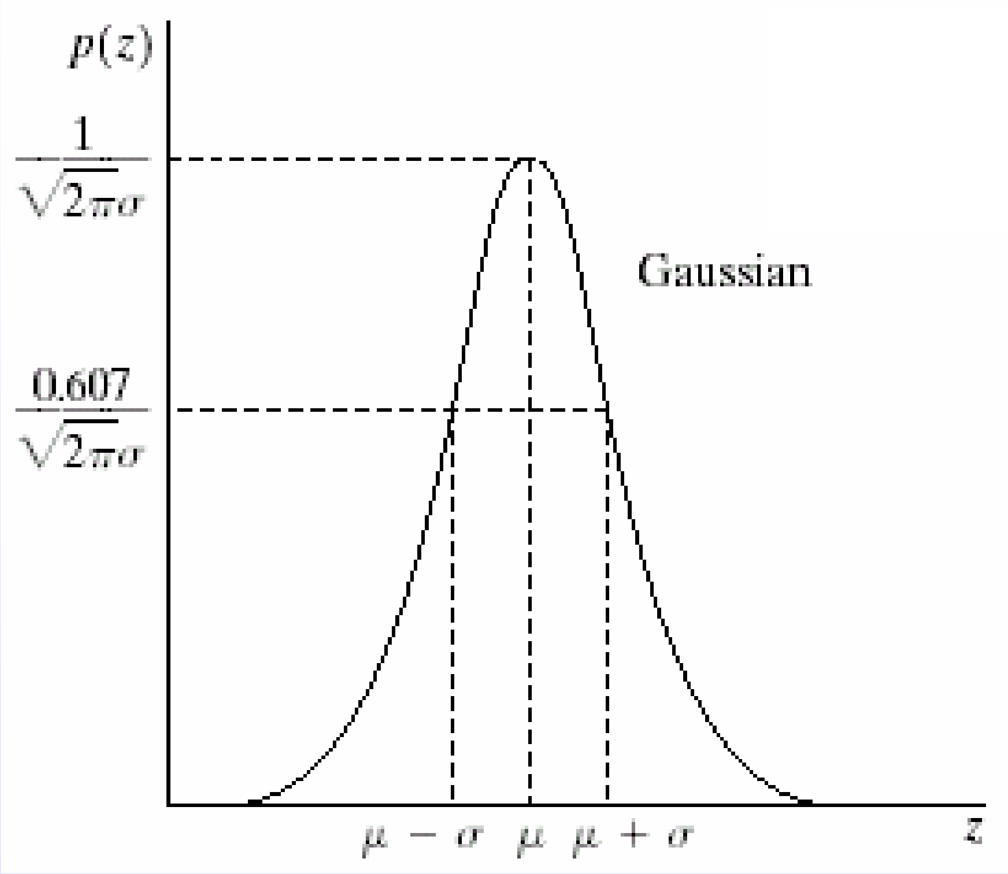
4 Image Restoration –Noise Models

PDF of Gaussian noise:

$$p(z) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(z-\mu)^2}{2\sigma^2}}$$

mean: $E\{z\} = \mu$

variance: $E\{(z-\mu)^2\} = \sigma^2$



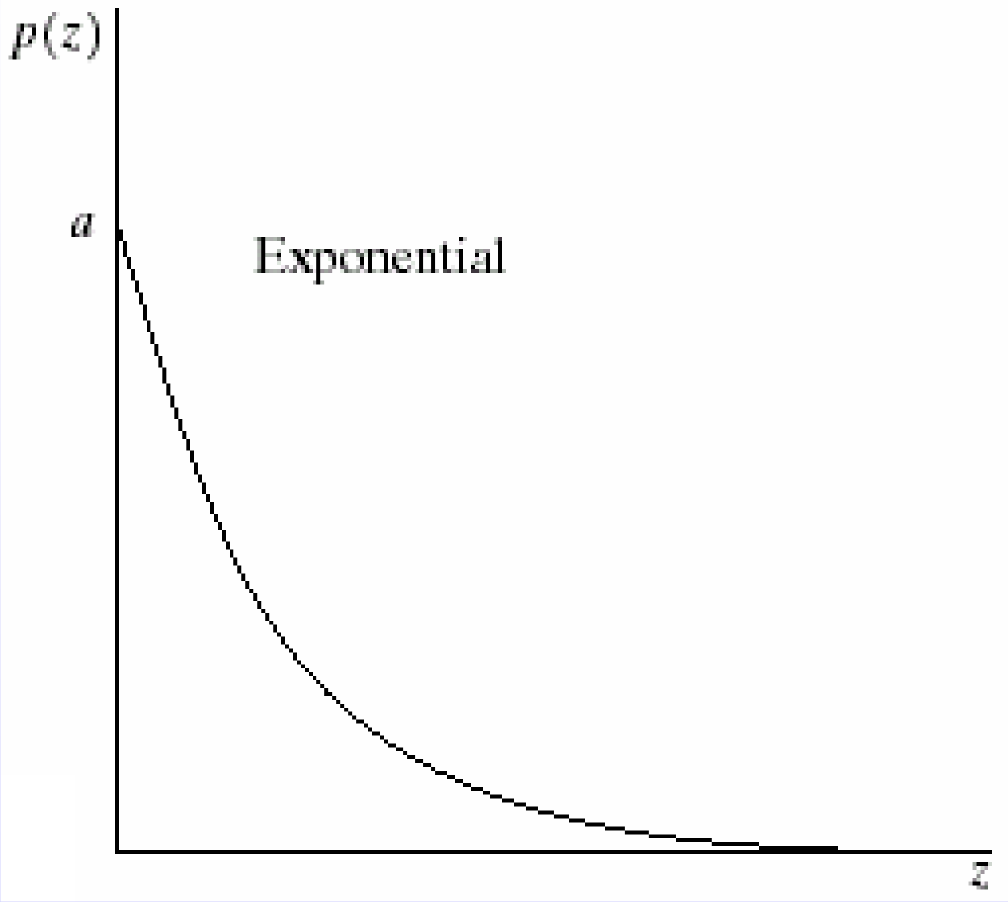
4 Image Restoration –Noise Models

PDF of exponential noise:

$$p(z) = \begin{cases} ae^{-az}, & \text{for } z \geq 0 \\ 0, & \text{for } z < 0 \end{cases}$$

mean: $\mu = 1/a$

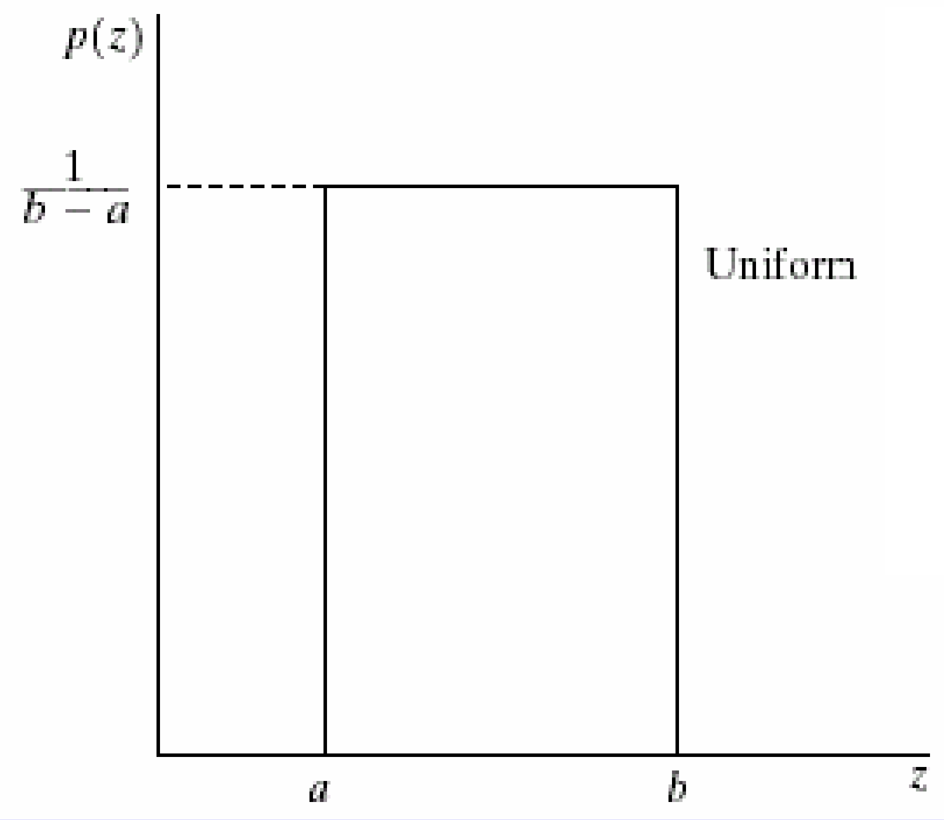
variance: $\sigma^2 = 1/a^2$



4 Image Restoration –Noise Models

PDF of uniform noise:

$$p(z) = \begin{cases} \dfrac{1}{b-a}, & \text{for } a \le z \le b \\ 0, & \text{otherwise} \end{cases}$$



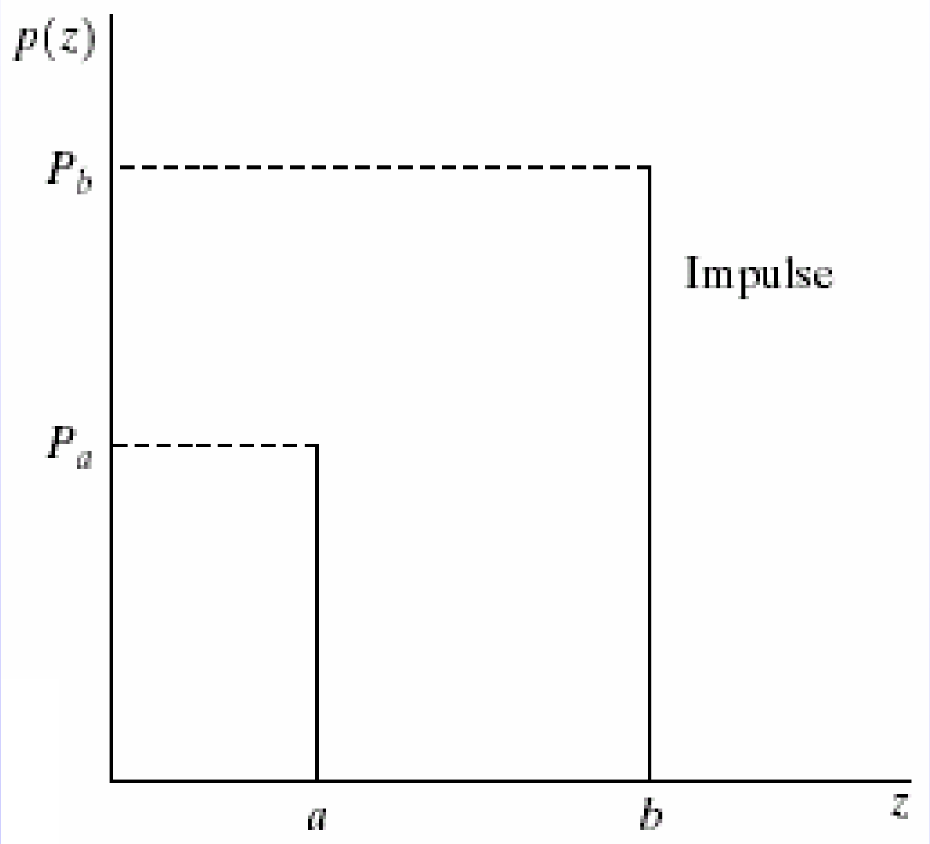
$$\text{mean: } \mu = \frac{b-a}{2}$$

$$\text{variance: } \sigma^2 = \frac{(b-a)^2}{12}$$

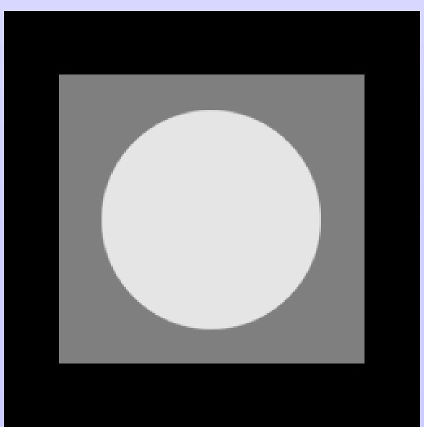
4 Image Restoration –Noise Models

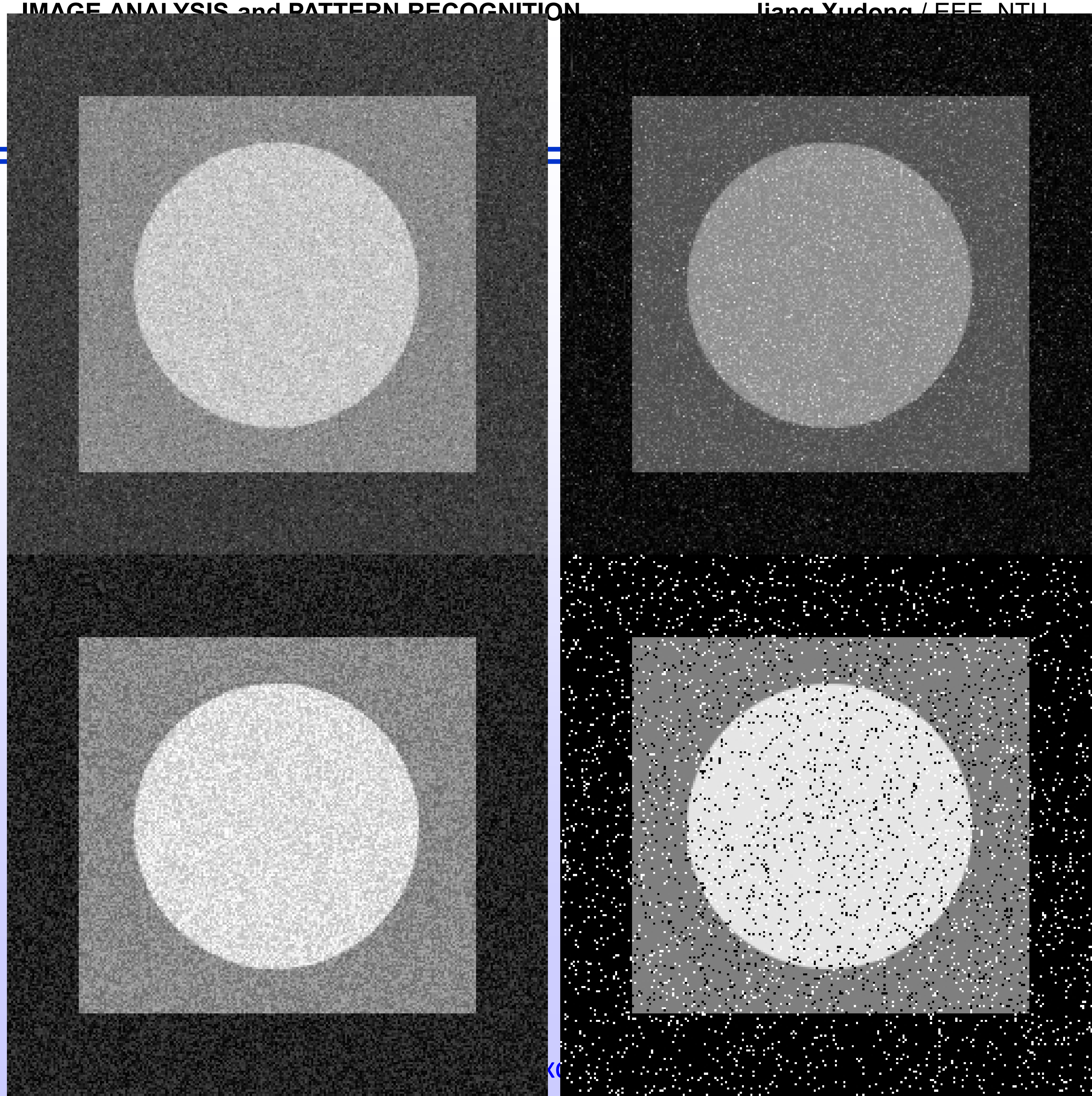
PDF of impulsive (salt & pepper) noise:

$$p(z) = \begin{cases} P_a, & \text{for } z = a \\ P_b, & \text{for } z = b \\ 0, & \text{otherwise} \end{cases}$$



Comparison of images corrupted by different types of noise: Gaussian exponential uniform and impulsive (salt & pepper) noise





4 Image Restoration –Inverse Filter

- **Inverse filter** recovers the original image $f(x,y)$ from the observed image $g(x,y)$.

$$G(u,v) \longrightarrow \boxed{H^I(u,v)} \longrightarrow \hat{F}(u,v)$$

Inverse filter: $H^I(u,v) = H^{-1}(u,v)$

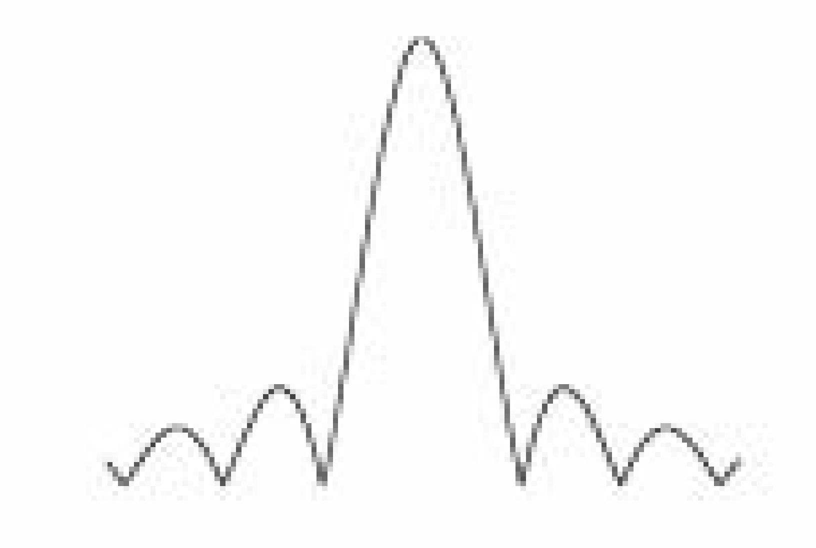
$$\hat{F}(u,v) = G(u,v)H^I(u,v) = [F(u,v)H(u,v) + N(u,v)]H^{-1}(u,v)$$

$$= F(u,v) + \frac{N(u,v)}{H(u,v)}$$

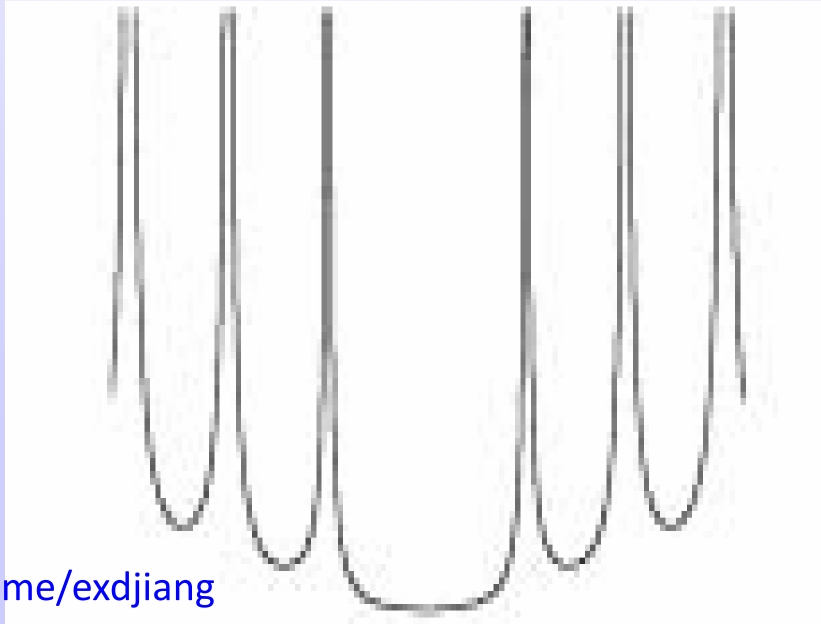
- Problem 1) $H^I(u,v)$ will not exist if $H(u,v)$ has any zero.
- Problem 2) Even otherwise, inverse filters result in noise amplification if $H(u,v)$ is small at certain frequency.

4 Image Restoration –Inverse Filter

- Consider the example of motion blur. The degradation frequency response $H(u,v)$ is a sinc function:



- Thus, the inverse filter $H^I(u,v)$ becomes infinite at some frequency points:

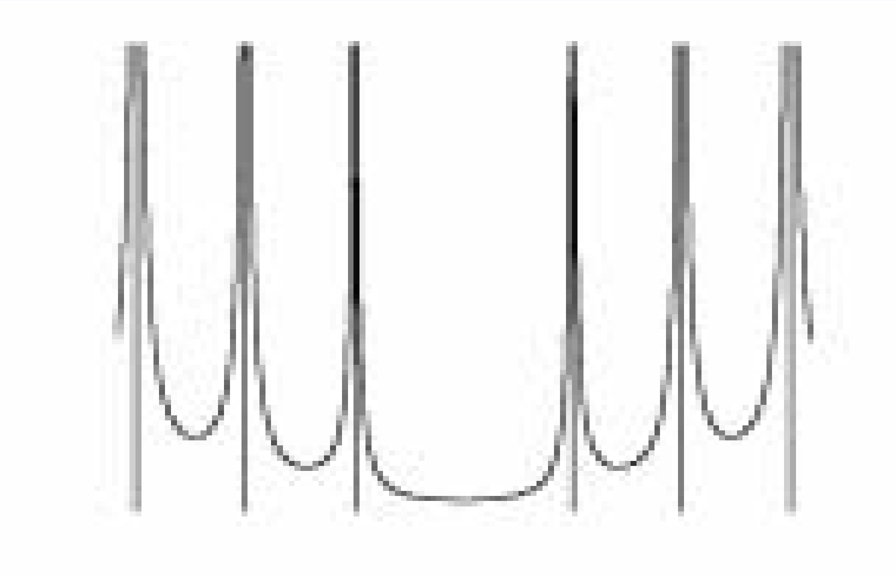


4 Image Restoration –Inverse Filter

- The **generalized inverse** (or, pseudo-inverse) filter solves this problem, and is defined as:

$$H^{-}(u,v)=\begin{cases}\dfrac{1}{H(u,v)}, & |H|\neq 0\\ 0, & |H|=0\end{cases}$$

In the example, the pseudo-inverse filter looks like



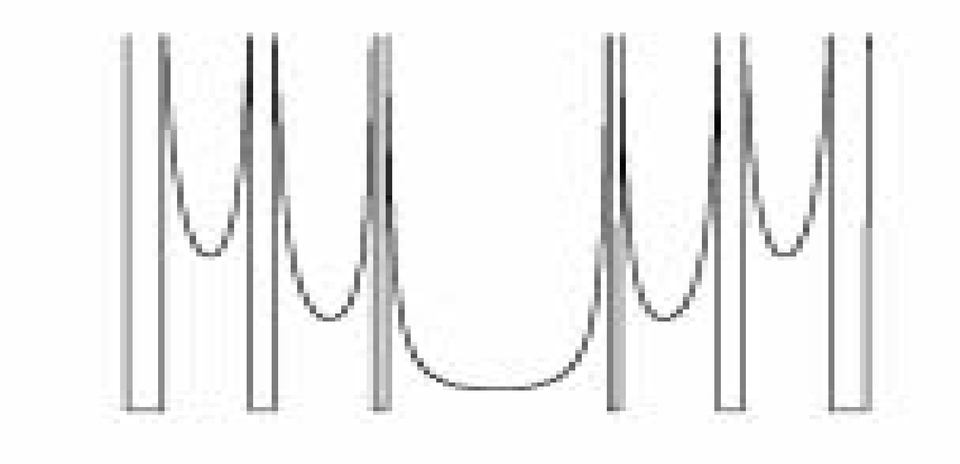
➢ Such a filter may be practically impossible to design due to the sharp transition on either side of zero.

4 Image Restoration –Inverse Filter

- Therefore, in practice, the pseudo-inverse filter is constructed as:

$$H^{-}(u,v) = \begin{cases} \dfrac{1}{H(u,v)}, & |H| \geq \varepsilon \\ 0, & |H| < \varepsilon \end{cases}$$

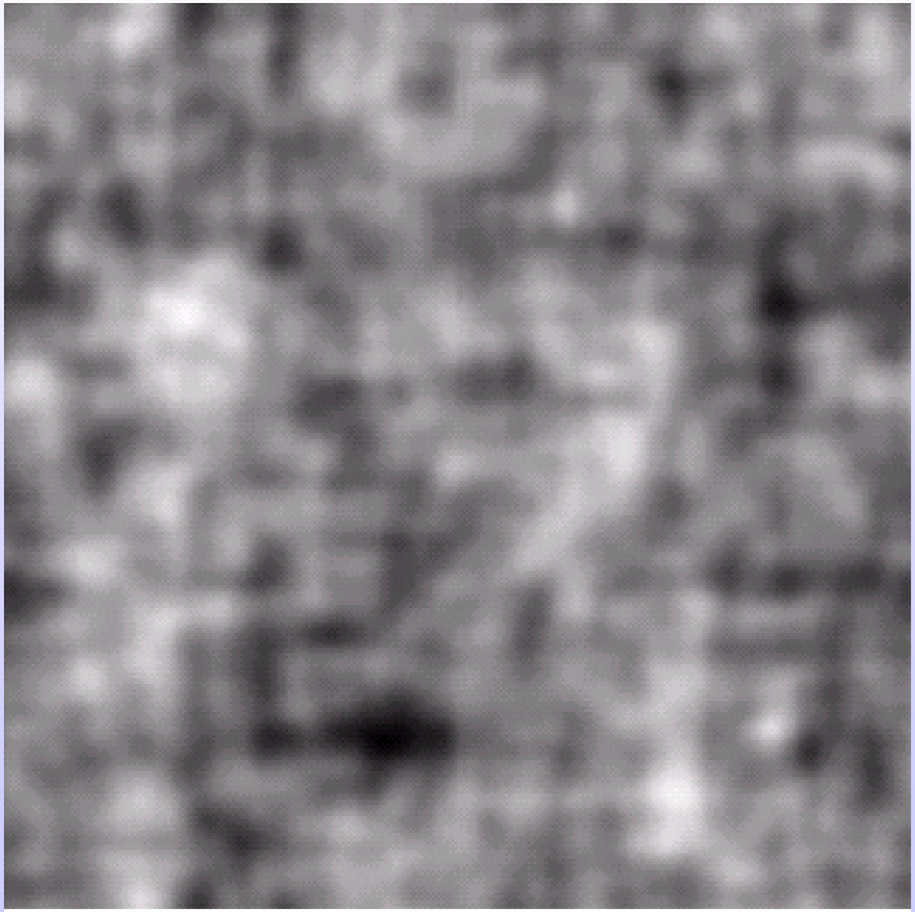
- The pseudo-inverse filter of the example becomes for ε= 0.05



4 Image Restoration –Inverse Filter

Image degraded by atmospheric turbulence

Restored image with full inverse filter



4 Image Restoration –Inverse Filter

Image degraded by atmospheric turbulence

Restored image with generalized inverse filter

4 Image Restoration –Wiener Filter

- Inverse and pseudo-inverse filtering doesn't perform well in the presence of noise

$$\hat{F}(u,v) = G(u,v)H^{I}(u,v)$$
$$= [F(u,v)H(u,v) + N(u,v)]H^{-1}(u,v)$$
$$= F(u,v) + \frac{N(u,v)}{H(u,v)}$$

- From the above formula we see that if $H(u,v)$ is zero or small at certain frequency, the term N/H at the output will be large, resulting in noise amplification.

4 Image Restoration –Wiener Filter

- To solve noise amplification Problem, note that if H is zero or small at some frequency or frequency range, then FH is also zero or small.

  Since $G(u,v) = F(u,v)H(u,v) + N(u,v)$

  at those frequencies the signal component (FH) is smaller than the noise component (N).

- If we amplify at those frequencies (because $1/H$ is large), the noise will become larger.

- Instead, we should attenuate at those frequencies.

4 Image Restoration –Wiener Filter

- This noise amplification problem of the inverse and pseudo-inverse filtering is solve by Wiener filter which is a minimum mean square error (MMSE) linear filter.

➢ The filter mean square error is given by:

$$
\begin{aligned}
e^2 &= E\{[f(x,y) - \hat{f}(x,y)]^2\} \\
&= E\{[f(x,y) - h^w(x,y) * g(x,y)]^2\}
\end{aligned}
$$

➢ To minimize e^2, let its differentiation with respect to h^w zero

$$\frac{\partial e^2}{\partial h^w(x,y)} = \frac{\partial E\{[f(x,y) - h^w(x,y) * g(x,y)]^2\}}{\partial h^w(x,y)} = 0$$

4 Image Restoration –Wiener Filter

As

$$g(x,y) = h(x,y) * f(x,y) + \eta(x,y) \quad \text{or}$$

$$G(u,v) = H(u,v)F(u,v) + N(u,v)$$

We have

$$\frac{\partial E\{[f(x,y) - h^w(x,y) * h(x,y) * f(x,y) - h^w(x,y) * \eta(x,y)]^2\}}{\partial h^w(x,y)} = 0$$

- Solving this equation with assumption that the noise has zero mean and is uncorrelated with the image and using the Fourier transform properties, it is not difficult to obtain the Wiener filter in the transform domain as:

4 Image Restoration –Wiener Filter

$$H^{w}(u,v)=\frac{H^{*}(u,v)S_{f}(u,v)}{\left|H(u,v)\right|^{2}S_{f}(u,v)+S_{\eta}(u,v)}$$

$H(u,v)$ is degradation function

$H^{*}(u,v)$ is complex conjugate of $H(u,v)$

$\left|H(u,v)\right|^{2}=H(u,v)H^{*}(u,v)$

$S_{f}(u,v)=\left|F(u,v)\right|^{2}$ is power spectrum of the original image

$S_{\eta}(u,v)=\left|N(u,v)\right|^{2}$ is power spectrum of the noise

The output of the Wiener filter is $\hat{F}(u,v)=H^{w}(u,v)G(u,v)$

4 Image Restoration –Wiener Filter Interpretation

- It is very easy to express the Wiener filter (least mean square error (LMSE) linear filter) as

$$H^{w}(u,v) = \frac{1}{H(u,v)}\left[\frac{|H(u,v)|^{2} S_{f}(u,v)}{|H(u,v)|^{2} S_{f}(u,v) + S_{\eta}(u,v)}\right]$$

$$= \frac{1}{H(u,v)} W(u,v)$$

➢ So, the Wiener filter is an inverse filter with a scale factor:

$$W(u,v) = \frac{|H(u,v)|^{2} S_{f}(u,v)}{|H(u,v)|^{2} S_{f}(u,v) + S_{\eta}(u,v)}$$

4 Image Restoration –Wiener Filter Interpretation

- Wiener filter is an inverse filter with a scale factor:

$$W(u,v) = \frac{|H(u,v)|^2 S_f(u,v)}{|H(u,v)|^2 S_f(u,v) + S_\eta(u,v)}$$

- If there is no noise, scaling is 1, the Wiener filter reduces to inverse filter.

➢ If for some frequency there is no signal, scaling is 0, the Wiener filter becomes 0 there, noise is suppressed.

➢ If there is no degradation, H = 1, the Wiener filter reduces to smoothing filter.

4 Image Restoration –Wiener Filter Interpretation

- Wiener filter is an inverse filter with a scale factor:

$$W(u,v)=\frac{|H(u,v)|^2 S_f(u,v)}{|H(u,v)|^2 S_f(u,v)+S_\eta(u,v)}$$

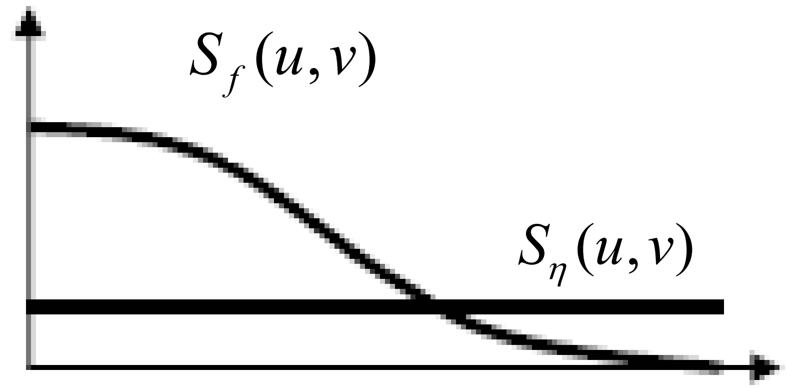
- This scale factor decreases with increasing noise $S_\eta(u,v)$ and increases with decreasing noise.
- This scale factor approaches to zero when the degradation function approaches to zero.

- Therefore, the Wiener filter solves noise amplification problem and the infinity problem of the inverse filter.

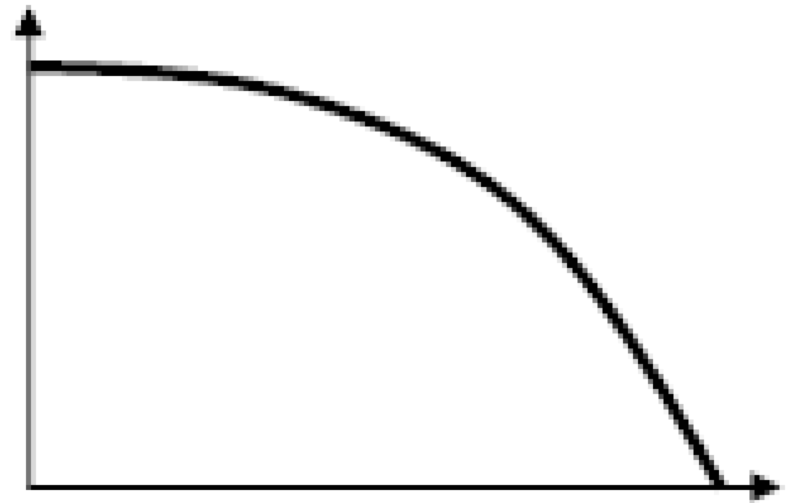
4 Image Restoration –Wiener Filter Interpretation

For Example:

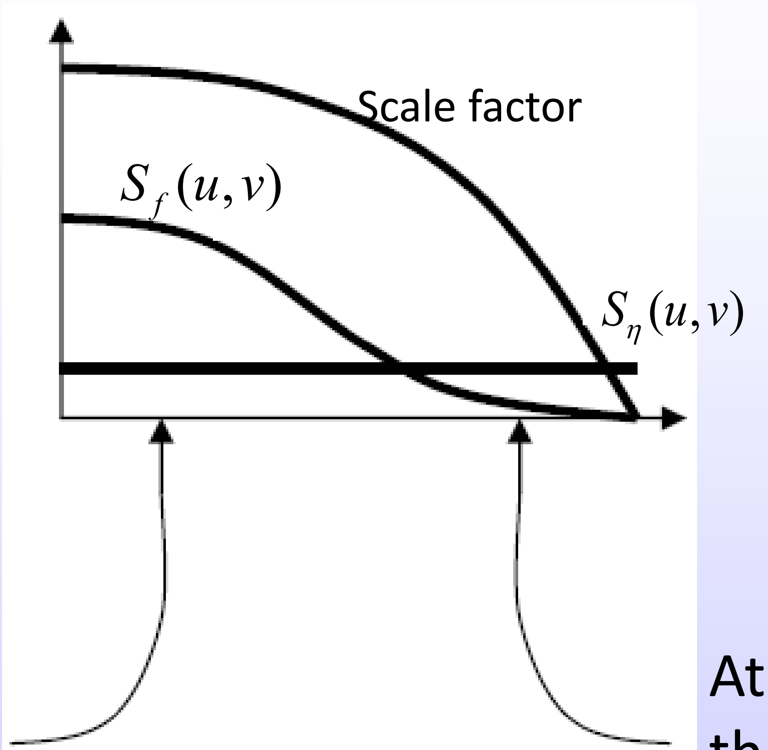
Consider the signal and noise components shown:



The scaling function is:



4 Image Restoration –Wiener Filter Interpretation



At these frequencies, the signal component is more than the noise component. So scaling is nearly 1.

At these frequencies, the noise component is more than the signal component. So scaling is nearly 0.

4 Image Restoration –Wiener Filter Interpretation

- In general, Image restoration is to inverse the degradation process $H(u,v)$.
- Problem 1) $H^{I}(u,v)$ will not exist if $H(u,v)$ has any zero.
- Problem 2) Even otherwise, inverse filters result in noise amplification if $H(u,v)$ is small at certain frequency.
- Therefore, we modify the inverse filter. $\frac{1}{H(u,v)} \not\Rightarrow \frac{1}{H(u,v)+c}?$

$$\frac{1}{H(u,v)} = \frac{H^*(u,v)}{H(u,v)H^*(u,v)} = \frac{H^*(u,v)}{|H(u,v)|^2} \Rightarrow \frac{H^*(u,v)}{|H(u,v)|^2 + c} \quad \text{Solve problem1}$$

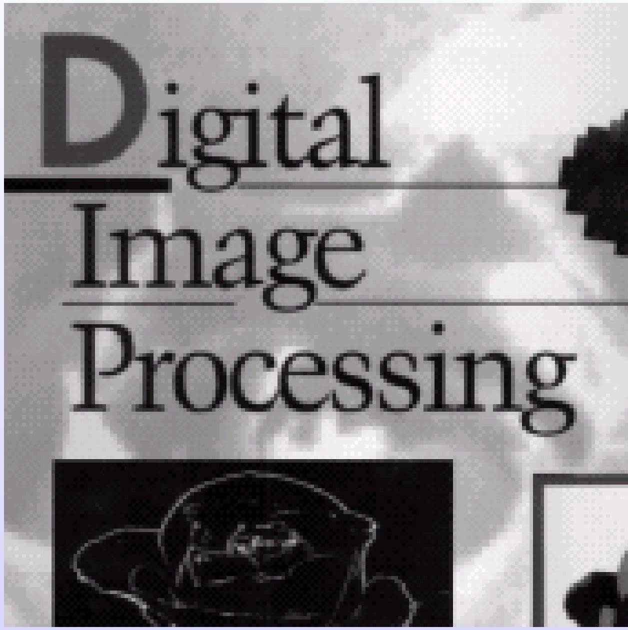
$$\Rightarrow \frac{H^*(u,v)}{|H(u,v)|^2 + [c = S_\eta(u,v)/S_f(u,v)]} \quad \text{Solve the problem 2!}$$

$$= \frac{H^*(u,v)S_f(u,v)}{|H(u,v)|^2 S_f(u,v) + S_\eta(u,v)} \quad \text{Same as Wiener filter!}$$

4 Image Restoration –Wiener Filter Results

Image corrupted by motion blur and light additive noise

Results of inverse filtering



Results of Wiener filtering

4 Image Restoration –Wiener Filter Results

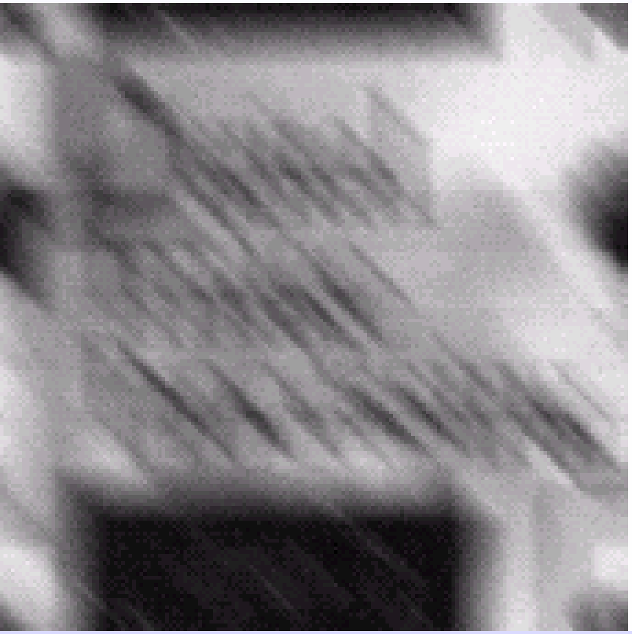


Image corrupted by motion blur and heavy additive noise

Results of inverse filtering

Results of Wiener filtering

4 Image Restoration –Frequency Domain Filtering

- In some cases, the noise model can be well represented in the frequency domain, such as various periodic noise and directional periodic noise.

$$g(x,y) = h(x,y) * f(x,y) + \eta(x,y) = f(x,y) + \eta(x,y)$$

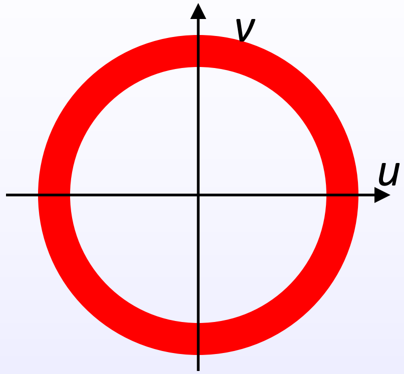
$$G(u,v) = H(u,v)F(u,v) + N(u,v) = F(u,v) + N(u,v)$$

- To remove such kinds of noise effectively, we design the restoration filter in the frequency domain because the 2-D Fourier transform provide the direction and frequency information of the image and hence noise.

- We will discuss band reject / pass filters, notch reject / pass filters and optimum notch filtering.

4 Image Restoration –Frequency Domain Filtering

- An ideal **band reject filter** is given by:



$$H(u,v)=\begin{cases}0, & \text{if } D_0-W/2\le D(u,v)\le D_0+W/2\\ 1, & \text{otherwise}\end{cases}$$

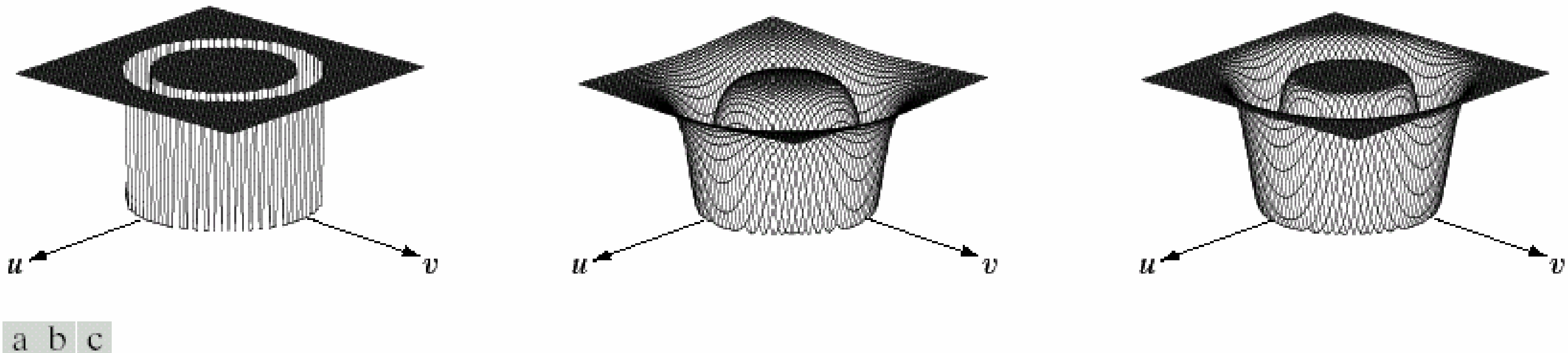
$D(u,v)=\sqrt{u^2+v^2}$ is the distance of point (u,v) from the origin.

Butterworth band reject filter: $$H(u,v)=\frac{1}{1+\left[\dfrac{D(u,v)W}{D^2(u,v)-D_0^2}\right]^{2n}}$$

Gaussian band reject filter:

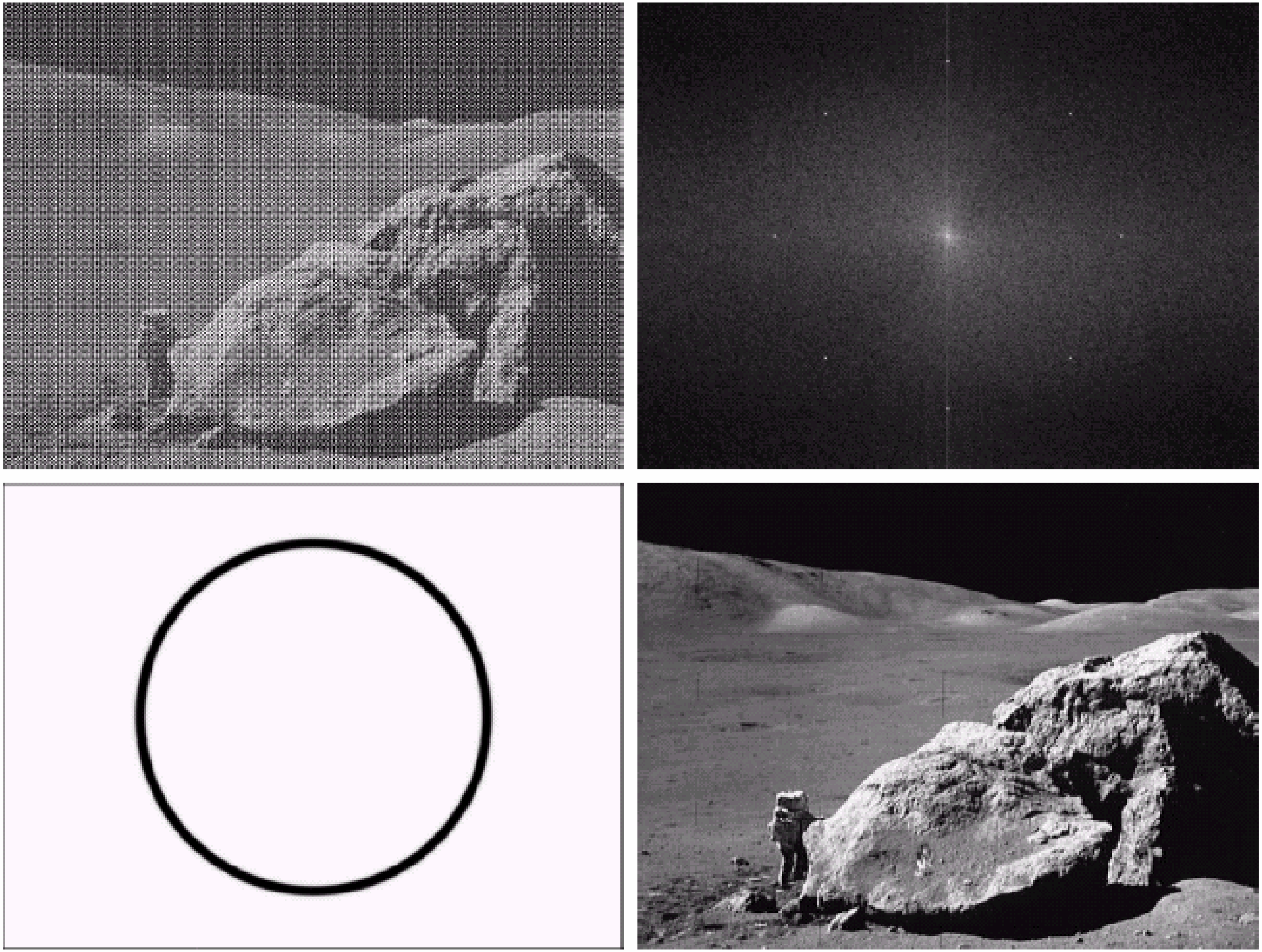
$$H(u,v)=1-e^{-\frac{1}{2}\left[\frac{D^2(u,v)-D_0^2}{D(u,v)W}\right]^2}$$

4 Image Restoration –Frequency Domain Filtering



From left to right, perspective plots of ideal, Butterworth (of order 1), and Gaussian bandreject filters.

4 Image Restoration –Frequency Domain Filtering



a	b
c	d

(a) Image corrupted by sinusoidal noise. (b) Spectrum of (a). (c) Butterworth bandreject filter (white represents 1). (d) Result of filtering. (Original image courtesy of NASA.)

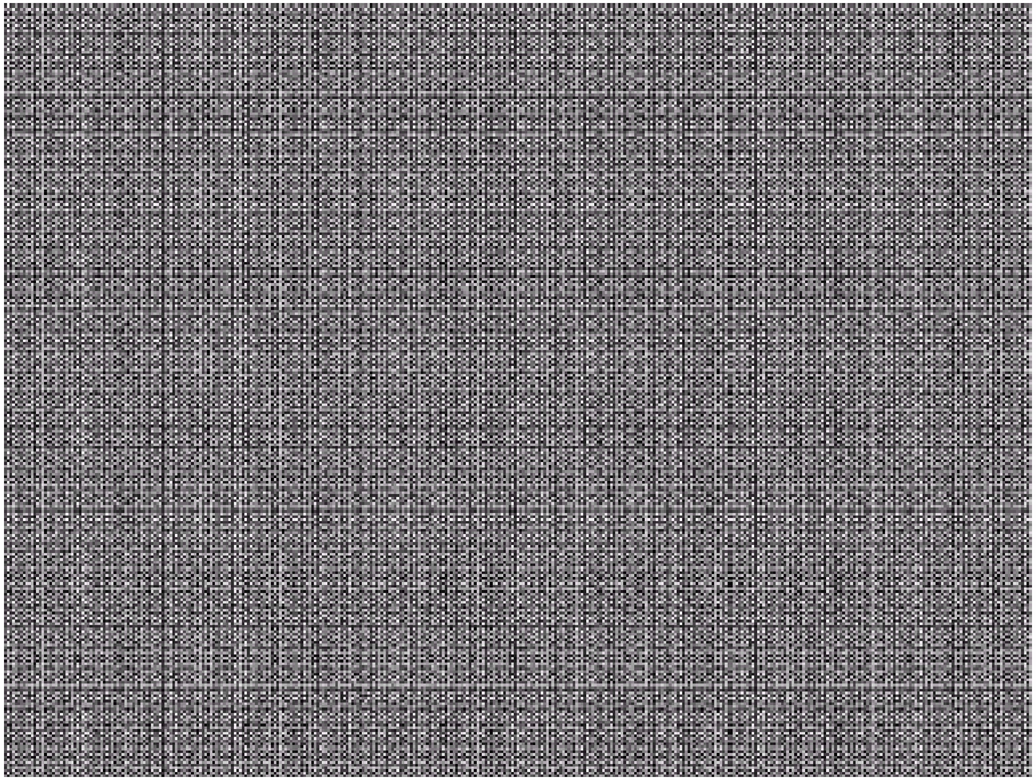
4 Image Restoration –Frequency Domain Filtering

- A band pass filter is given by:

$$H_{bp}(u,v) = 1 - H_{br}(u,v)$$

- It is used to separate noise from the image.

Noise pattern of the image in Fig. 5.16(a) obtained by bandpass filtering.



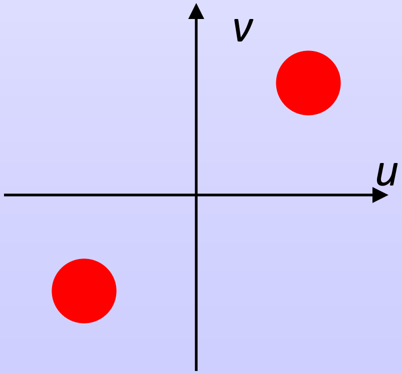
4 Image Restoration –Frequency Domain Filtering

- An ideal **notch reject filter** is given by:

$$H(u,v)=\begin{cases}0, & \text{if } D_1(u,v)\le D_0 \text{ OR } D_2(u,v)\le D_0\\ 1, & \text{otherwise}\end{cases}$$

Where $D_1(u,v)=\left[\left(u-u_0\right)^2+\left(v-v_0\right)^2\right]^{1/2}$

and $D_2(u,v)=\left[\left(u+u_0\right)^2+\left(v+v_0\right)^2\right]^{1/2}$



4 Image Restoration –Frequency Domain Filtering

- A Butterworth notch reject filter:

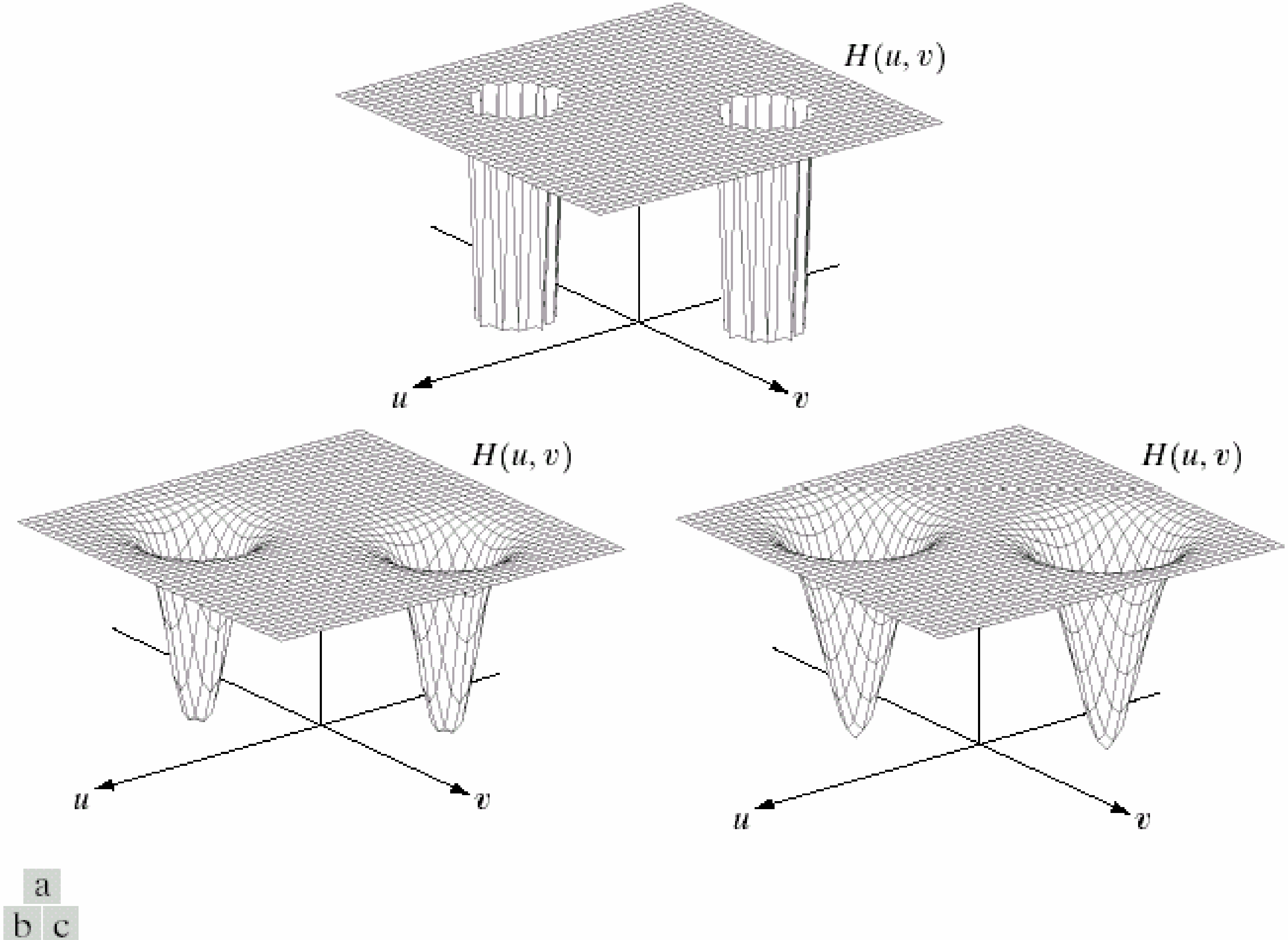
$$H(u,v) = \frac{1}{1 + \left[\frac{D_0^2}{D_1(u,v)D_2(u,v)}\right]^n}$$

- A Gaussian notch reject filter:

$$H(u,v) = 1 - e^{-\frac{1}{2}\left[\frac{D_1(u,v)D_2(u,v)}{D_0^2}\right]}$$

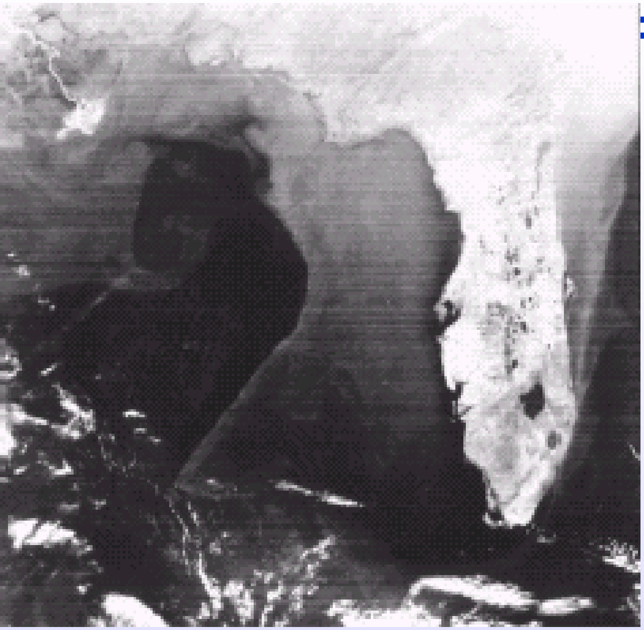
- The notch pass filter is given by: $H_{np}(u,v) = 1 - H_{nr}(u,v)$

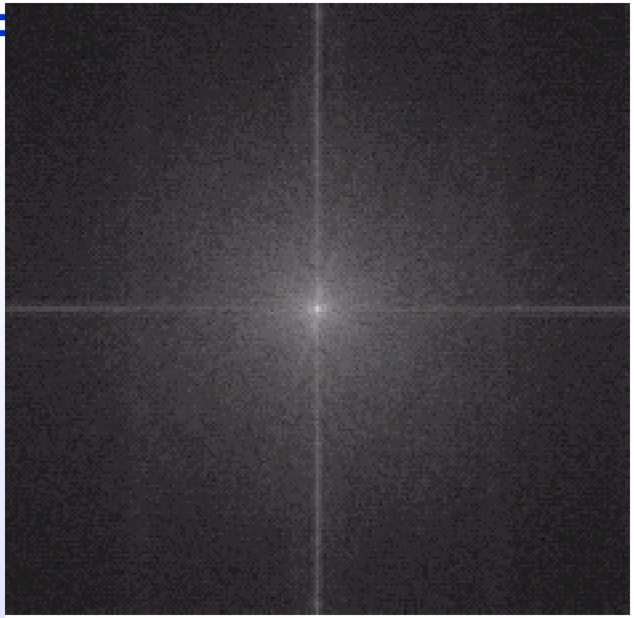
4 Image Restoration –Frequency Domain Filtering

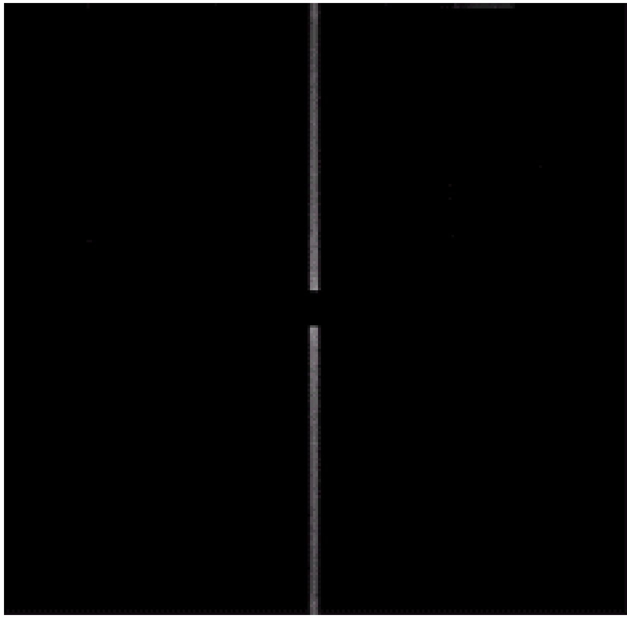


Perspective plots of (a) ideal, (b) Butterworth (of order 2), and (c) Gaussian notch (reject) filters.

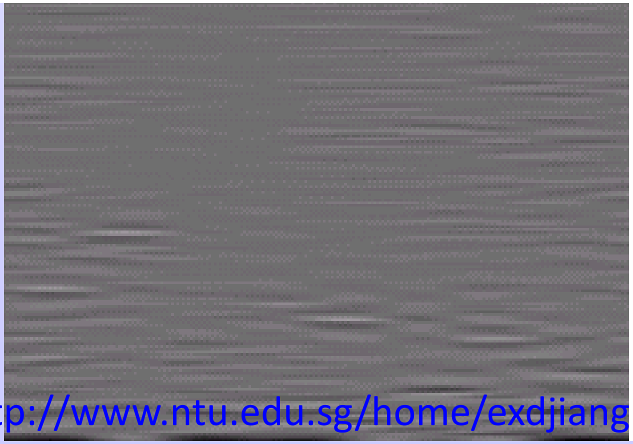
4 Image Restoration –Frequency Domain Filtering

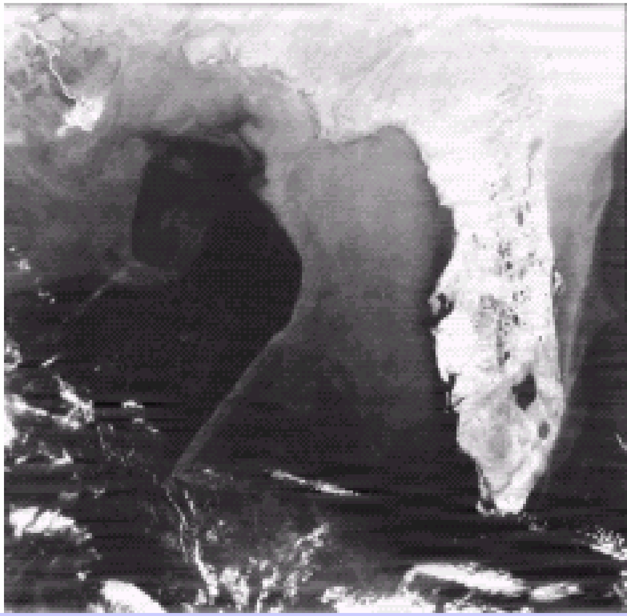






(a) Satellite image of Florida and the Gulf of Mexico (note horizontal sensor scan lines). (b) Spectrum of (a). (c) Notch pass filter shown superimposed on (b). (d) Inverse Fourier transform of filtered image, showing noise pattern in the spatial domain. (e) Result of notch reject filtering. (Original image courtesy of NOAA.)





4 Image Restoration –Frequency Domain Filtering

- **Optimum Notch Filter**

1. Observe the DFT of the image to find the noise spikes and hence design notch pass filter $H(u,v)$
2. $\eta(x,y) = \mathfrak{F}^{-1}\{H(u,v)G(u,v)\}$
3. $w(x,y) = \dfrac{\overline{g(x,y)\eta(x,y)} - \bar{g}(x,y)\bar{\eta}(x,y)}{\overline{\eta^2}(x,y) - \bar{\eta}^2(x,y)}$
4. $\hat{f}(x,y) = g(x,y) - w(x,y)\eta(x,y)$

Where the bar operation is mean over the neighborhood pixels of (x, y), S(x,y).

4 Image Restoration –Frequency Domain Filtering

- $w(x,y)$ is determined by making the mean square error minimum.

$$\overline{\hat{f}}(x,y) = \underset{(s,t)\in S(x,y)}{mean} \left\{ \hat{f}(s,t) \right\}$$

$$\sigma^2(x,y) = \sum_{(s,t)\in S(x,y)} \left[\overline{\hat{f}}(x,y) - \hat{f}(s,t) \right]^2$$

$S(x,y)$ is a set that contains the neighborhood pixels of (x, y).

Minimize $\sigma^2(x,y)$ by Slove: $\dfrac{\partial \sigma^2(x,y)}{\partial w(x,y)} = 0$

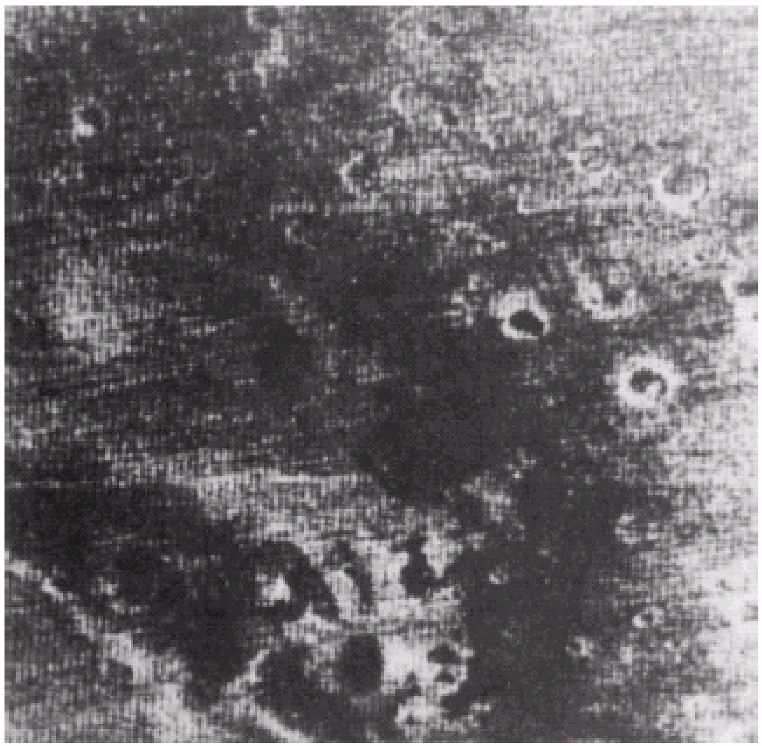
Obtain: $w(x,y) = \dfrac{\overline{g(x,y)\eta(x,y)} - \overline{g}(x,y)\overline{\eta}(x,y)}{\overline{\eta^2}(x,y) - \overline{\eta}^2(x,y)}$

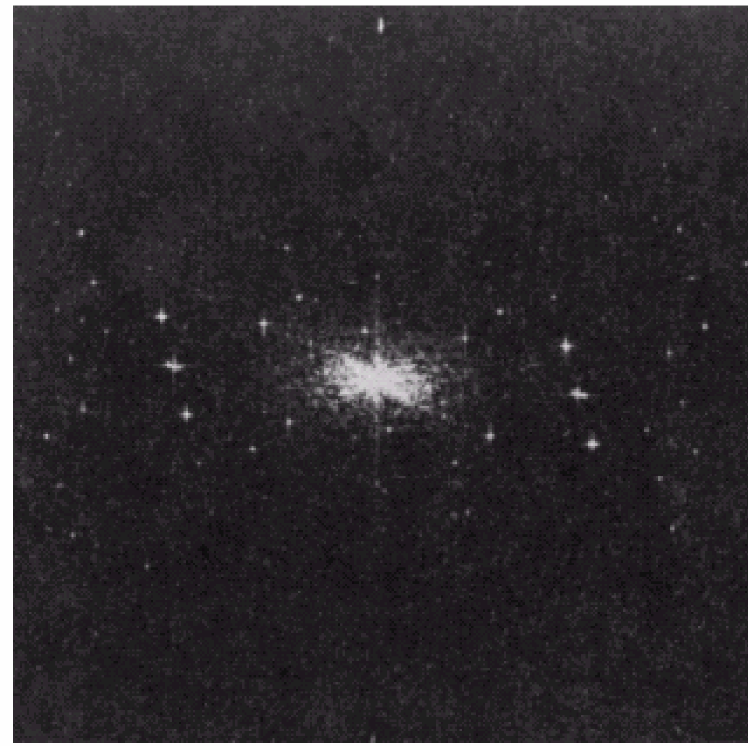
4 Image Restoration –Frequency Domain Filtering

- Optimum Notch Filter

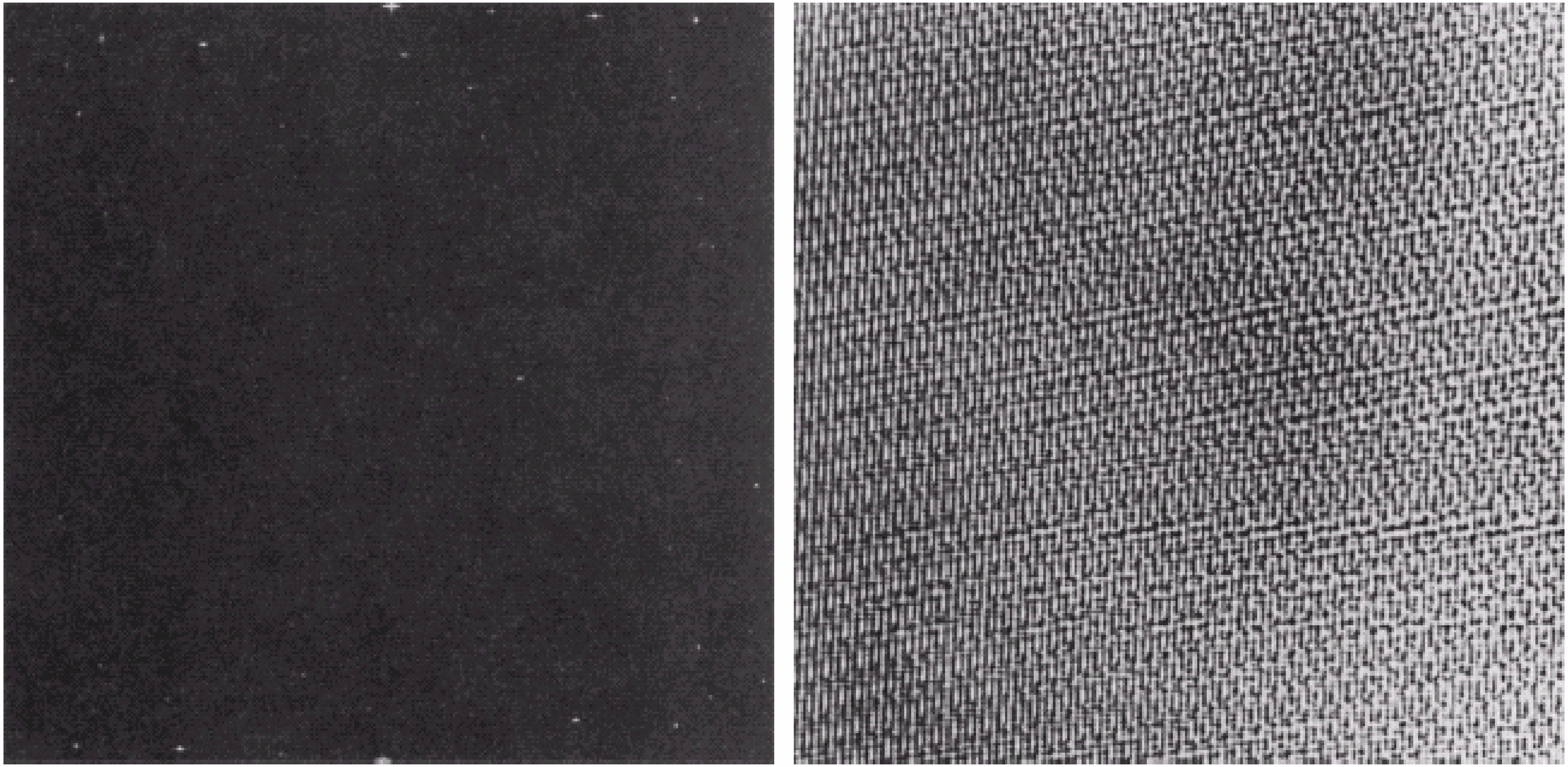
a b

(a) Image of the Martian terrain taken by *Mariner 6*. (b) Fourier spectrum showing periodic interference. (Courtesy of NASA.)





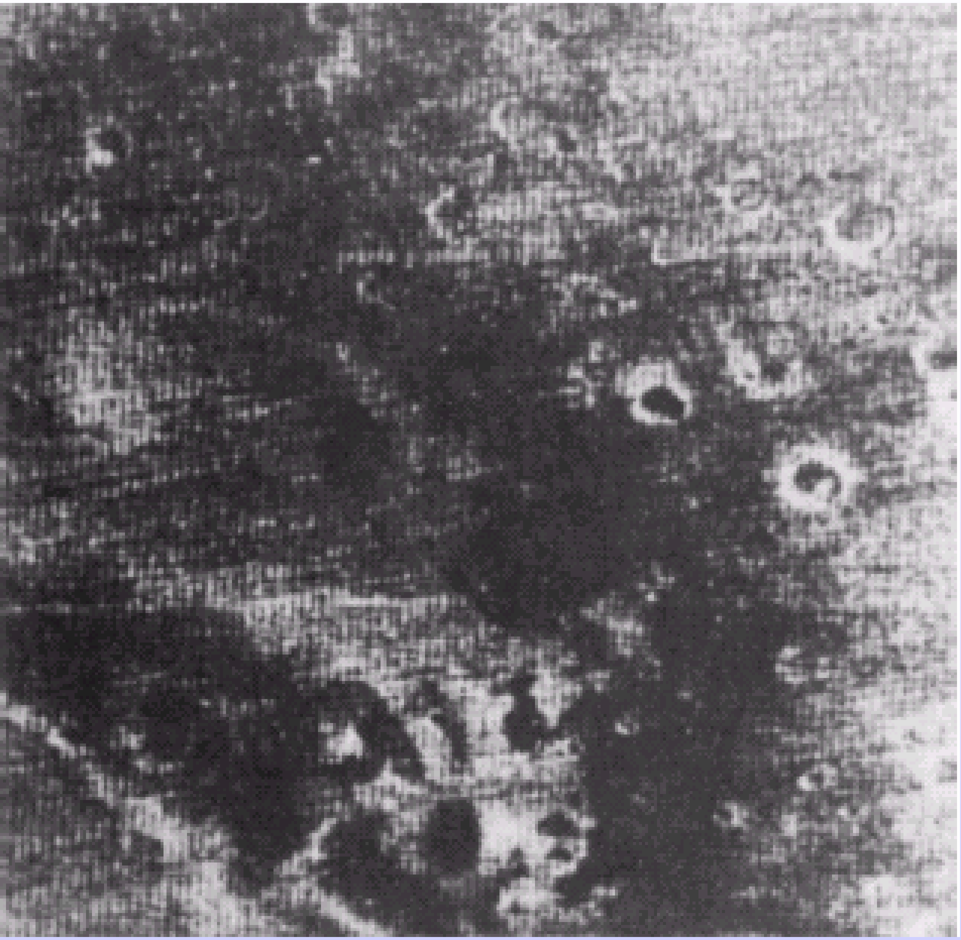
4 Image Restoration –Frequency Domain Filtering



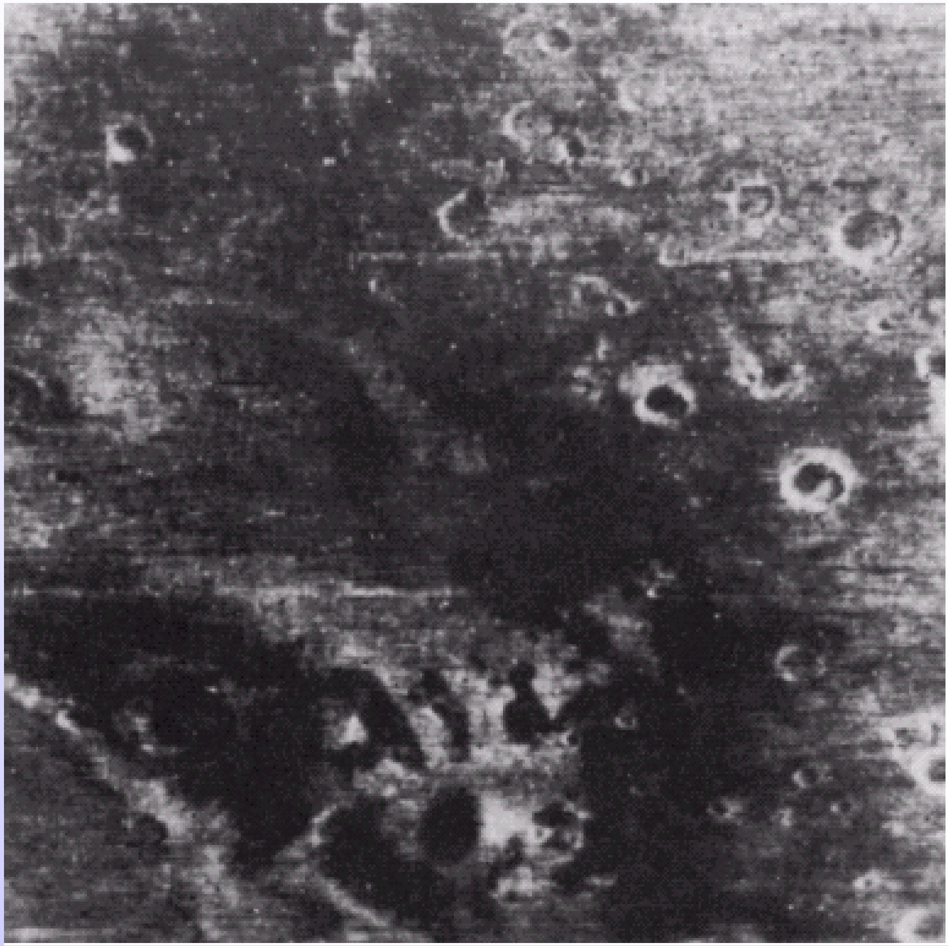
a b

(a) Fourier spectrum of $N(u, v)$, and (b) corresponding noise interference pattern $\eta(x, y)$. (Courtesy of NASA.)

4 Image Restoration –Frequency Domain Filtering



Original image



Processed image

4 Image Restoration –Problem Example

Problem:

During acquisition, an image undergoes uniform planar motion in *x*- and *y*-directions with $x_o(t)$ and $y_o(t)$ and T is the duration of the exposure. Assuming that shutter opening and closing time are negligible, give an expression for the blurring function $H(u,v)$.

Solution:

Suppose the real image without motion blurring is $f(x,y)$ and the motion blurred image is $g(x,y)$. We have

$$g(x,y) = \int_0^T f(x - x_o(t), y - y_o(t))dt$$

4 Image Restoration –Problem Example

Solution:

Appling Fourier transform, we get

$$G(u,v)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}g(x,y)e^{-j2\pi(ux+vy)}dxdy$$

$$=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left[\int_{0}^{T}f[x-x_o(t),y-y_o(t)]dt\right]e^{-j2\pi(ux+vy)}dxdy$$

$$=\int_{0}^{T}\left[\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}f[x-x_o(t),y-y_o(t)]e^{-j2\pi(ux+vy)}dxdy\right]dt$$

4 Image Restoration –Problem Example

Solution:

Based on the Fourier transform properties, we get

$$G(u,v)=\int_0^T F(u,v)e^{-j2\pi[ux_0(t)+vy_0(t)]}dt$$

$$=F(u,v)\int_0^T e^{-j2\pi[ux_0(t)+vy_0(t)]}dt$$

$$\therefore H(u,v)=\frac{G(u,v)}{F(u,v)}=\int_0^T e^{-j2\pi[ux_0(t)+vy_0(t)]}dt$$

4 Image Restoration –Problem Example

Problem:

If the image undergoes uniform linear planar motion, i.e. $x_o(t)=at/T$ and $y_o(t)=bt/T$. What is the blurring function $H(u,v)$?

Solution:

$$H(u,v)=\int_0^T e^{-j2\pi[ux_0(t)+vy_0(t)]}dt=\int_0^T e^{-j2\pi(ua+vb)t/T}dt$$

$$=\frac{-T}{j2\pi(ua+vb)}\left(e^{-j2\pi(ua+vb)}-e^{0}\right)$$

$$=\frac{T\sin[\pi(ua+vb)]}{\pi(ua+vb)}e^{-j\pi(ua+vb)}$$

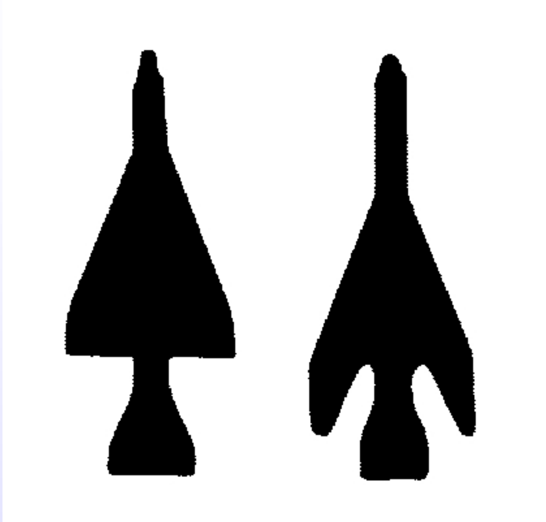
5 Morphological Image Processing –Outline

- Introduction
- Set Theory and Logic Operation
- Dilation and Erosion
- Opening and Closing
- Morphological Algorithm and applications
- ~~Morphological Processing for Gray-scale Images~~

5 Morphological Image Processing –Introduction

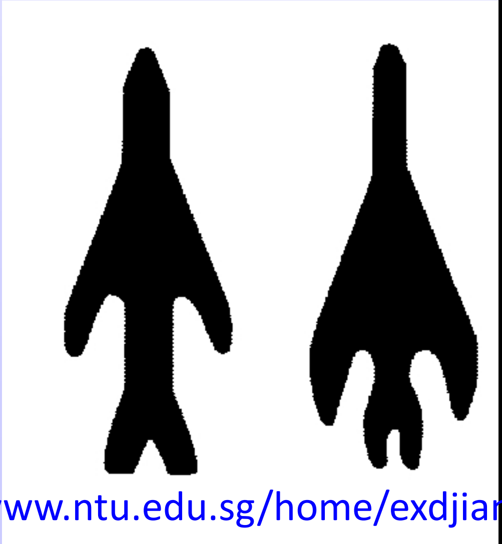
Looking at these images……

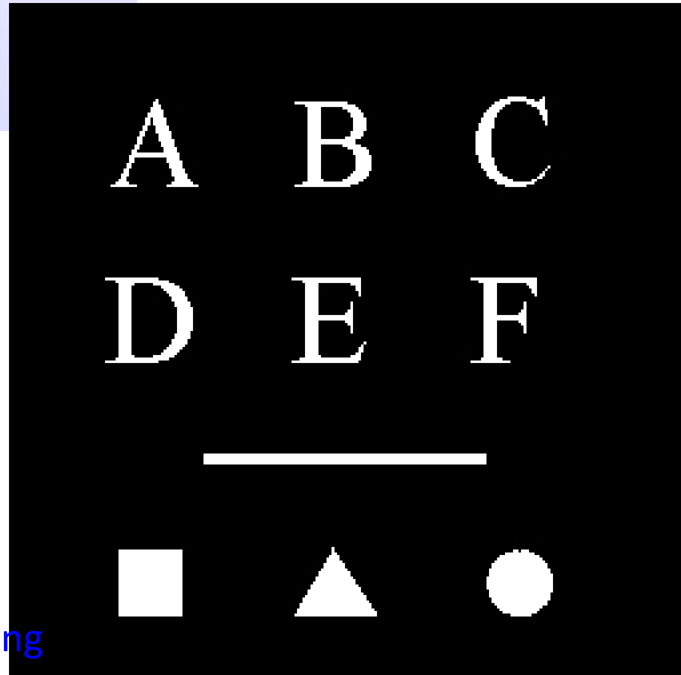
What is interesting, important or useful information we care about?



The grey value of the image is not important as there are only two different grey values.

- Region shape and boundaries of object are important.

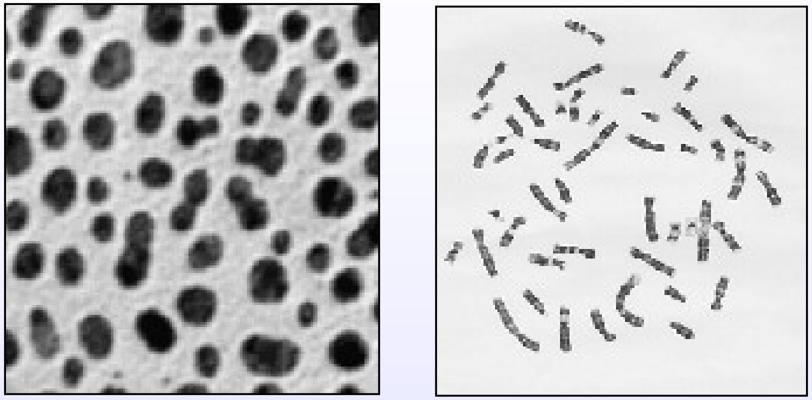




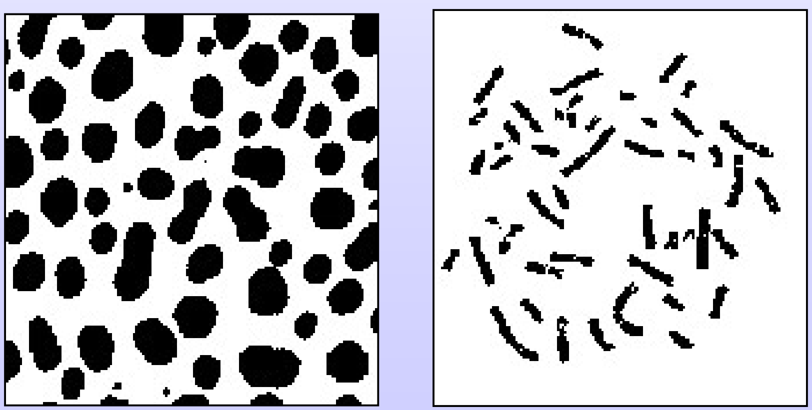
- Form and structure can be represented by object pixel set.

exdjiang@ntu.edu.sg http://www.ntu.edu.sg/home/exdjiang

5 Morphological Image Processing –Introduction



Grayscale Images



Binary Images

Image analysis needs to measure the characteristics of objects in the images.

Geometric measurements (e.g., object location, orientation, area, length of perimeter) are important characteristics of objects

These geometric characteristics is often easier to be extracted/measured from binary images.

5 Morphological Image Processing –Introduction

- Visual perception requires transformation of images so as to make explicit particular **shape information**.

- **Goal:** Distinguish meaningful shape information from irrelevant one.

- The vast majority of shape processing and analysis techniques are based on designing a **shape operator** which satisfies desirable properties.

5 Morphological Image Processing –Introduction

- Morphology deals with form and structure
- Mathematical morphology is a tool for extracting image components useful in:
  - representation and description of region shape (e.g. boundaries)
  - pre- or post-processing (filtering, thinning, etc.)
- Morphological operations are powerful tools in image analysis. They usually operate on binary images and thus often follow a segmentation task or an edge detection task.
- Based on set theory and logic operations

5 Morphology –Set Theory

- A two dimensional integer space is denoted by $\mathbf{Z}^2$.
- An element in this space has two components $a=(a_1, a_2)$.
- For image representation, $a=(a_1, a_2)$ are the x- and y-coordinates of a pixel.
- Let A be a set in $\mathbf{Z}^2$. If $a=(a_1, a_2)$ is an element of A, we denote

$$a \in A$$

- If not, then

$$a \notin A$$

5 Morphology –Set Theory

- $\varnothing$ denotes null (empty) set

- An example that specifies a set C:

$$C = \{\, w \mid w = a+d,\ a \in A \,\}$$
$$d = (8, 5).$$

- If a set A is a subset of B, we denote:

$$A \subseteq B$$

5 Morphology –Set Theory

- Union of A and B: $C = A \cup B$

- Intersection of A and B: $D = A \cap B$

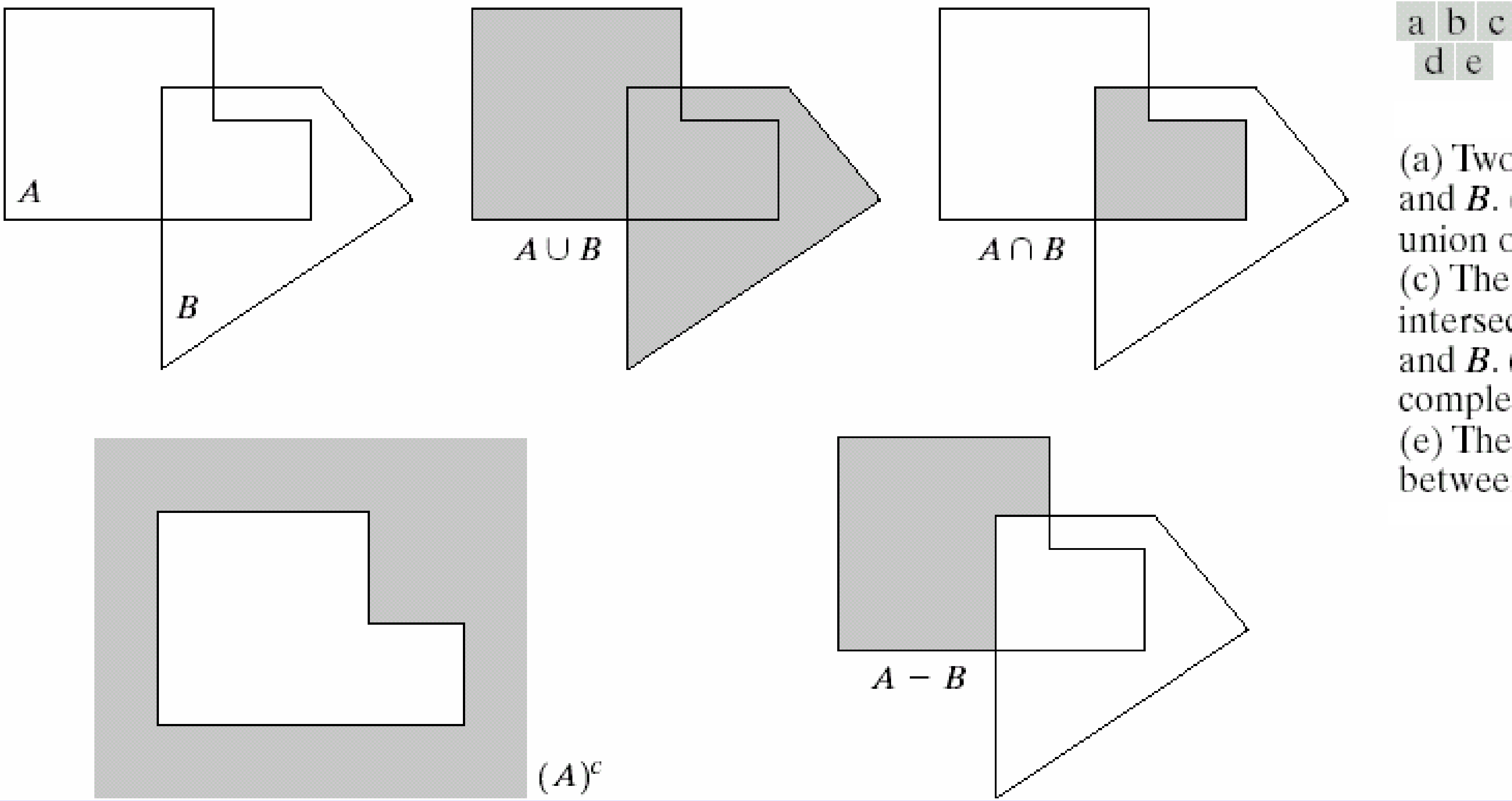
- Disjoint sets: $A \cap B = \emptyset$

- Complement of A: $A^c = \{w \mid w \notin A\}$

- Difference of A and B:

$$A - B = \{w \mid w \in A, w \notin B\} = A \cap B^c$$

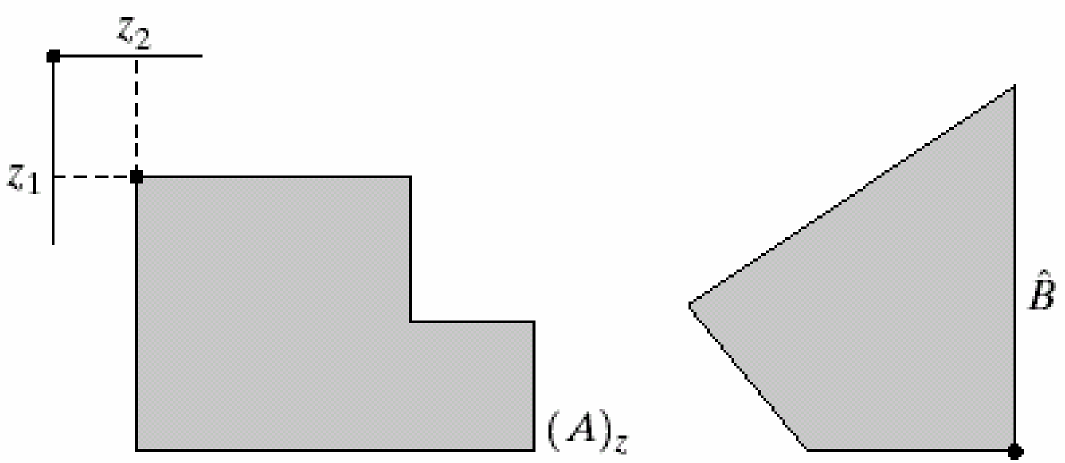
5 Morphology –Set Theory



(a) Two sets A and B. (b) The union of A and B. (c) The intersection of A and B. (d) The complement of A. (e) The difference between A and B.

5 Morphology –Set Theory

- Translation of A by $z=(z_1, z_2)$: $(A)_z = \{c \mid c = a+z, a \in A\}$



a b

(a) Translation of A by z.
(b) Reflection of B.

$$\hat{B} = \{w \mid w = -b,\ b \in B\}$$

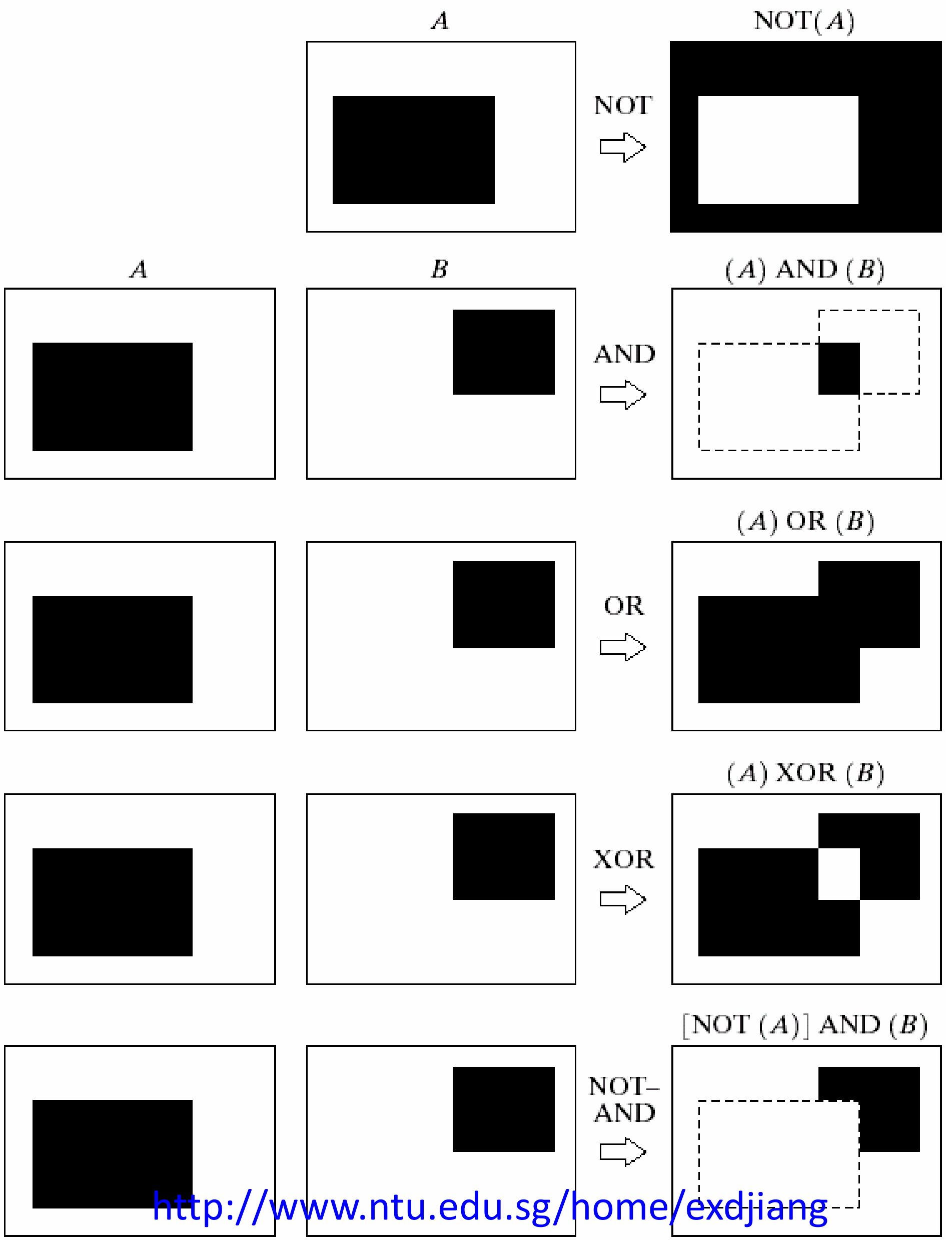
- Reflection of B:

5 Morphology –Set Theory

- Three basic logical operations

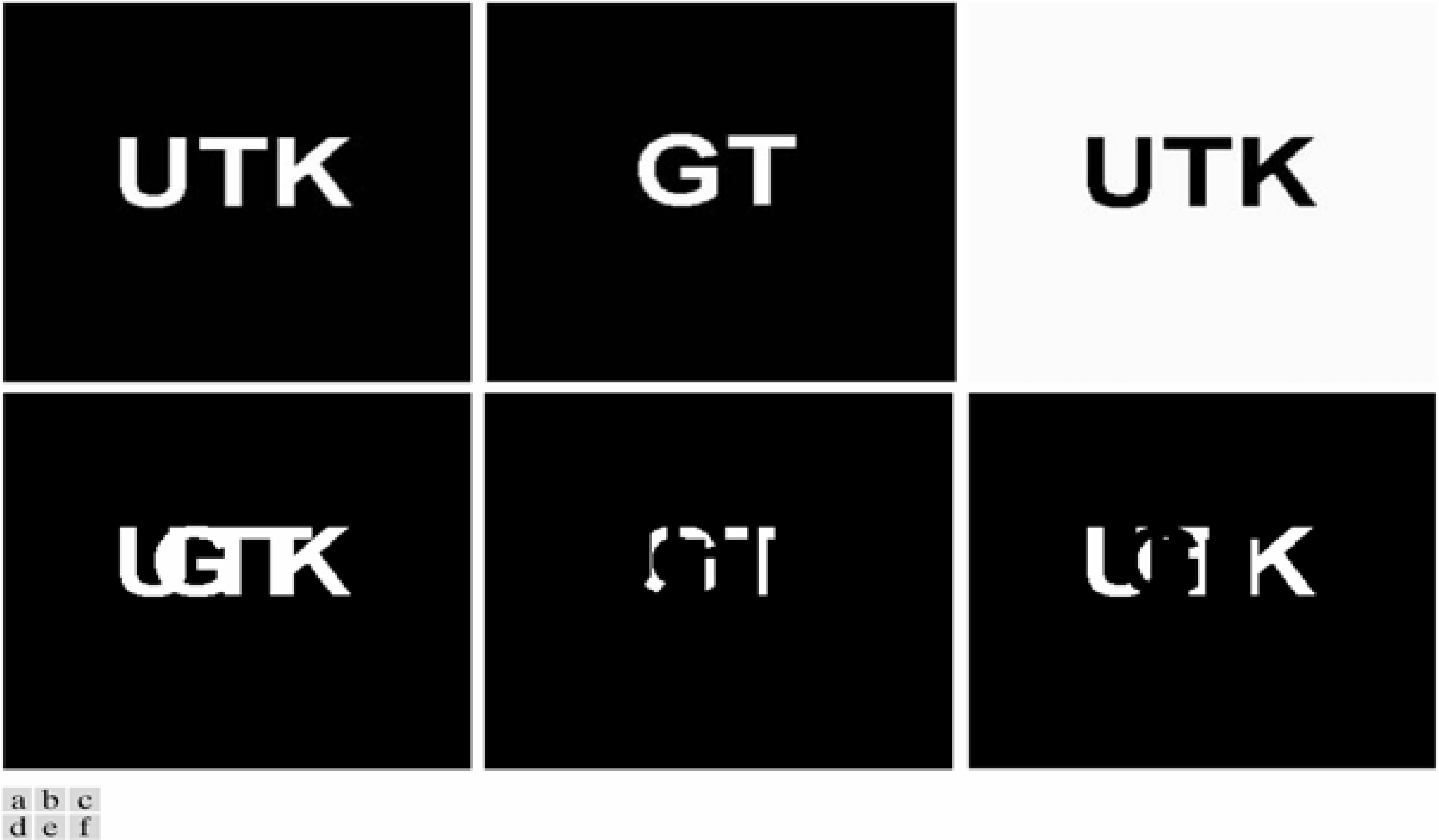
p	q	p AND q (also $p \cdot q$)	p OR q (also $p + q$)	NOT (p) (also $\bar{p}$)
0	0	0	0	1
0	1	0	1	1
1	0	0	1	0
1	1	1	1	0

5 Morphology –Set Theory



Some logic operations between binary images. Black represents binary 1s and white binary 0s in this example.

5 Morphology –Set Theory



(a) Binary image A. (b) Binary image B. (c) Complement ~A. (d) Union A | B. (e) Intersection A & B. (f) Set difference A & ~B.

5 Morphology –Morphological Operators

- Primary morphological operations are Dilation and Erosion
- More complicated morphological operators such as Opening and Closing can be designed by means of combining erosions and dilations
- Opening generally smoothes the contour of an image and eliminates protrusions
- Closing smoothes sections of contours, but it generally fuses breaks, holes and gaps

5 Morphology –Dilation

- Dilation of A by B, denoted by $A \oplus B$, is defined as:

$$A \oplus B = \left\{ z \mid \left[\left(\hat{B} \right)_z \cap A \right] \neq \varnothing \right\}$$

- Interpretation:
  Obtaining the reflection of B about its origin and then shifting this reflection by z. Dilation of A by B then is the set of all z displacements such that the shifted $\hat{B}$ and A overlap by at least one nonzero element.

- B is called the structuring element in Dilation.

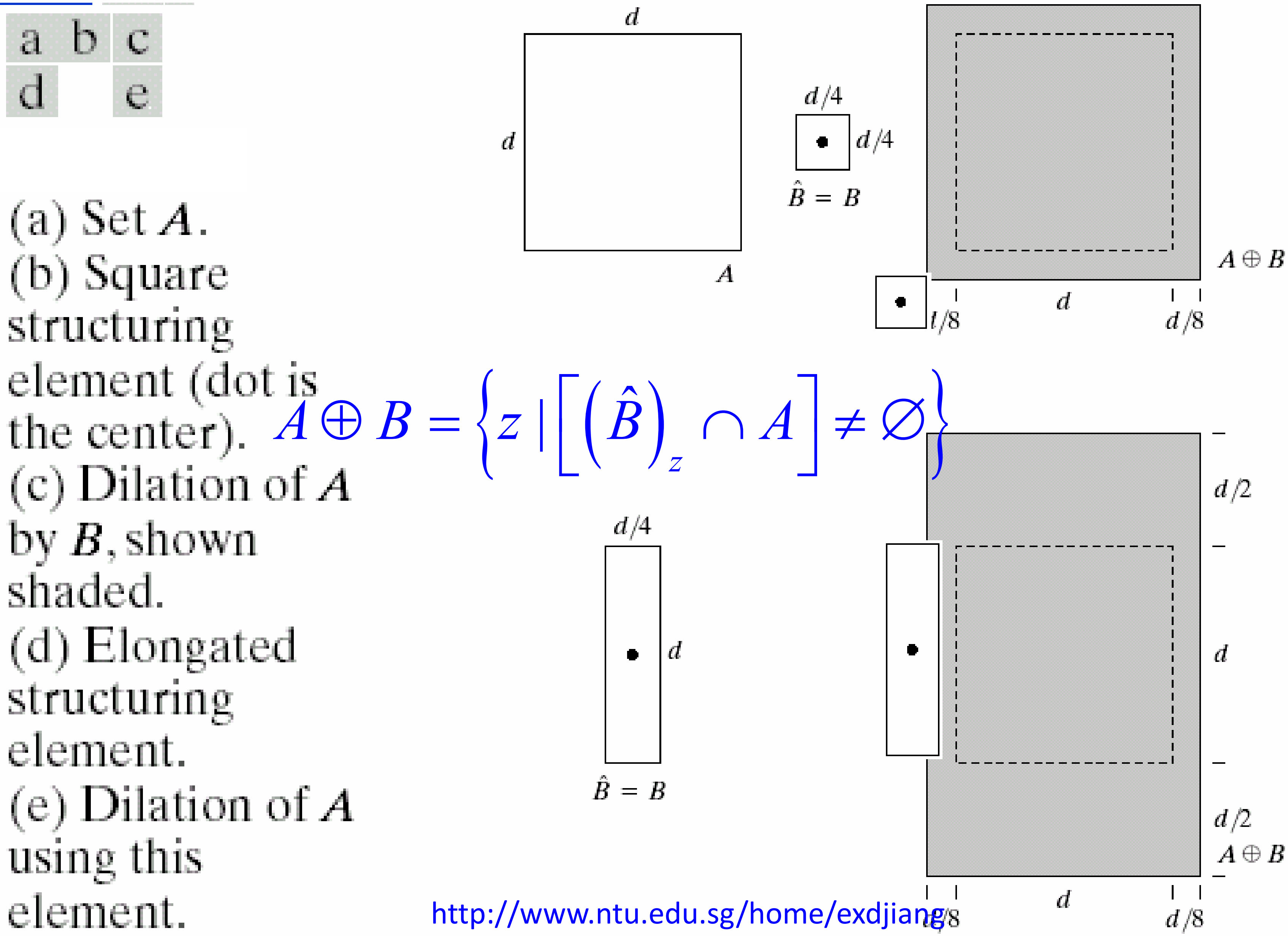
5 Morphology –Dilation

- Dilation of A by B can also be expressed as:

$$A \oplus B = \left\{ z \mid \left[\left(\hat{B} \right)_z \cap A \right] \subseteq A \right\}$$

- Further Interpretation:
  Set B can be viewed as a convolution mask. The basic process of "flipping" B and then successively displace it so that it slides over set (image) A is analogous to the convolution.

5 Morphology –Dilation



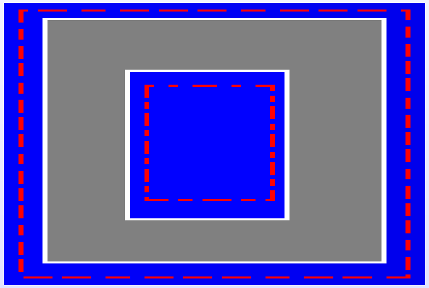
(a) Set A.
(b) Square structuring element (dot is the center).
(c) Dilation of A by B, shown shaded.
(d) Elongated structuring element.
(e) Dilation of A using this element.

$$A \oplus B = \left\{ z \mid \left[\left(\hat{B} \right)_z \cap A \right] \neq \varnothing \right\}$$

5 Morphology –Dilation

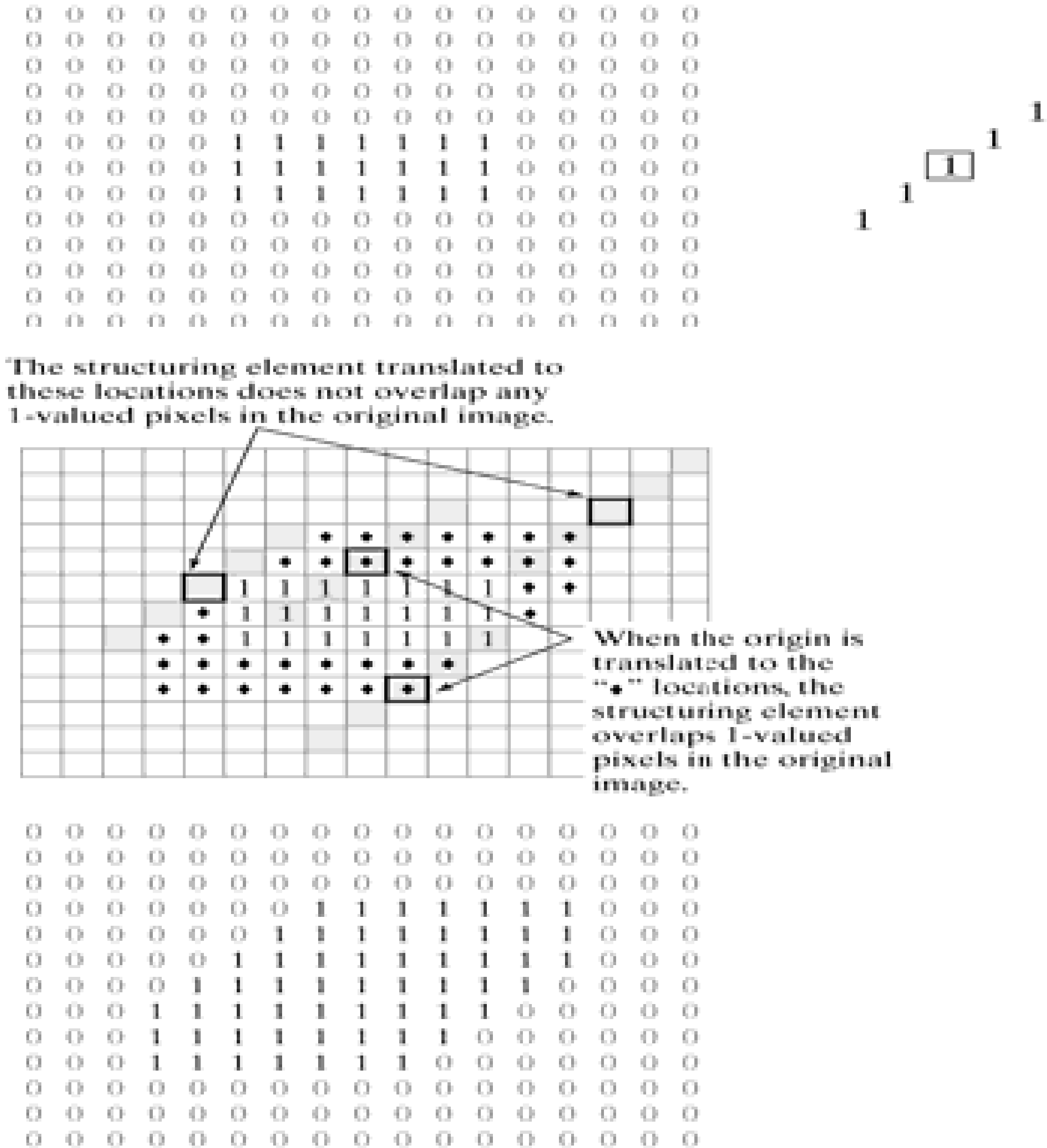
- The dilation morphological operation generates an output image 'g' from an input image 'f' using a structuring element 'h' where:
  Place(shift) the center(origin) of 'h' at(to) (x, y)

$$g(x,y)=\begin{cases}1, & \text{if } h \text{ hints } f\\ 0, & \text{else}\end{cases}$$



- The effect of dilation with 3 x 3 mask is to add a single layer of pixels to the outer edge of an object and to decrease by a single layer of pixels to the holes in the object.
- A 5 x 5 mask will add two layers of pixels which is equivalent to applying a 3 x 3 mask twice.
- The main application of dilation is to remove small holes from the interior of an object.

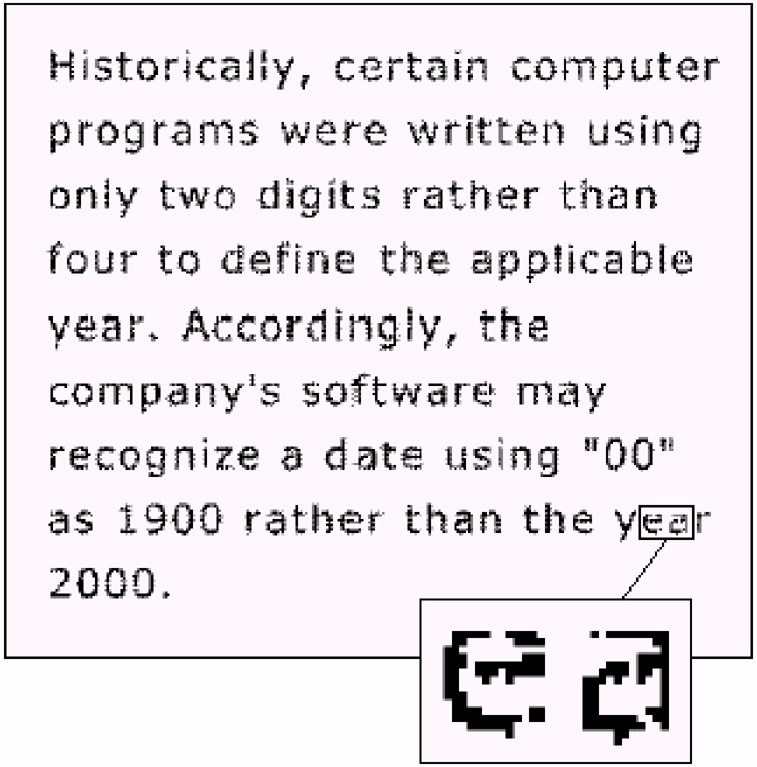
Dilation

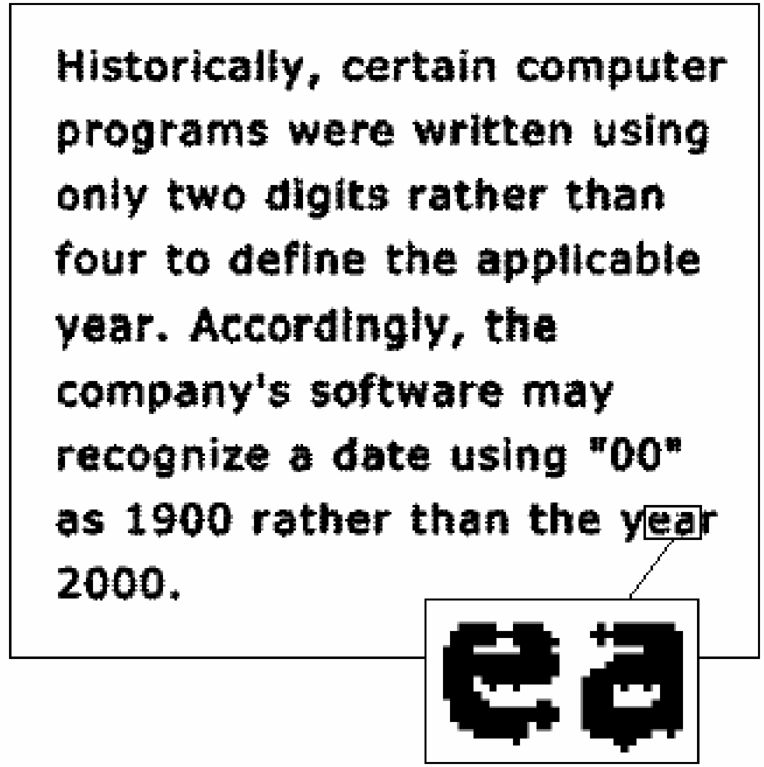


a b
c
d

Illustration of dilation.
(a) Original image with rectangular object.
(b) Structuring element with five pixels arranged in a diagonal line. The origin of the structuring element is shown with a dark border.
(c) Structuring element translated to several locations on the image.
(d) Output image.

5 Morphology –Dilation


Historically, certain computer programs were written using only two digits rather than four to define the applicable year. Accordingly, the company's software may recognize a date using "00" as 1900 rather than the year 2000.


Historically, certain computer programs were written using only two digits rather than four to define the applicable year. Accordingly, the company's software may recognize a date using "00" as 1900 rather than the year 2000.

0	1	0
1	1	1
0	1	0

a c
b

(a) Sample text of poor resolution with broken characters (magnified view).
(b) Structuring element.
(c) Dilation of (a) by (b). Broken segments were joined.

5 Morphology –Erosion

- Erosion of A by B, denoted $A \ominus B$, is defined as:

$$A \ominus B = \left\{ z \mid (B)_z \subseteq A \right\}$$

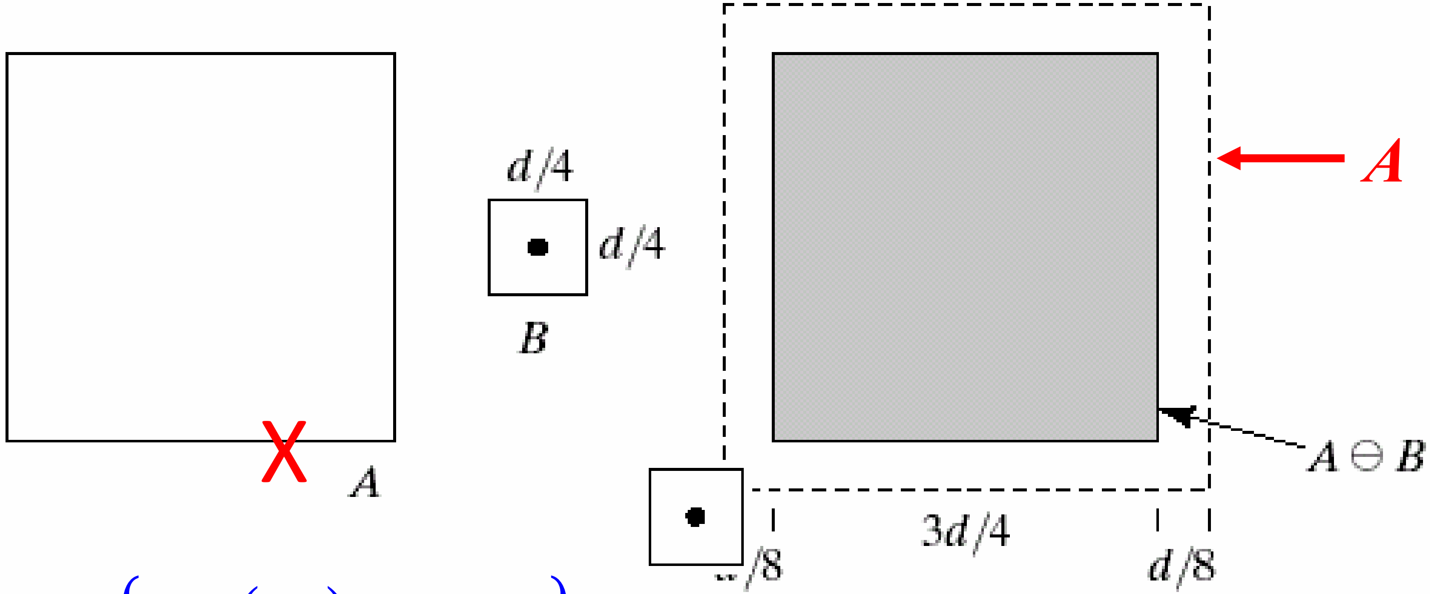
- Erosion of A by B is the set of all points z such that B, translated by z, is contained in A.

- Comparing with the Dilation:

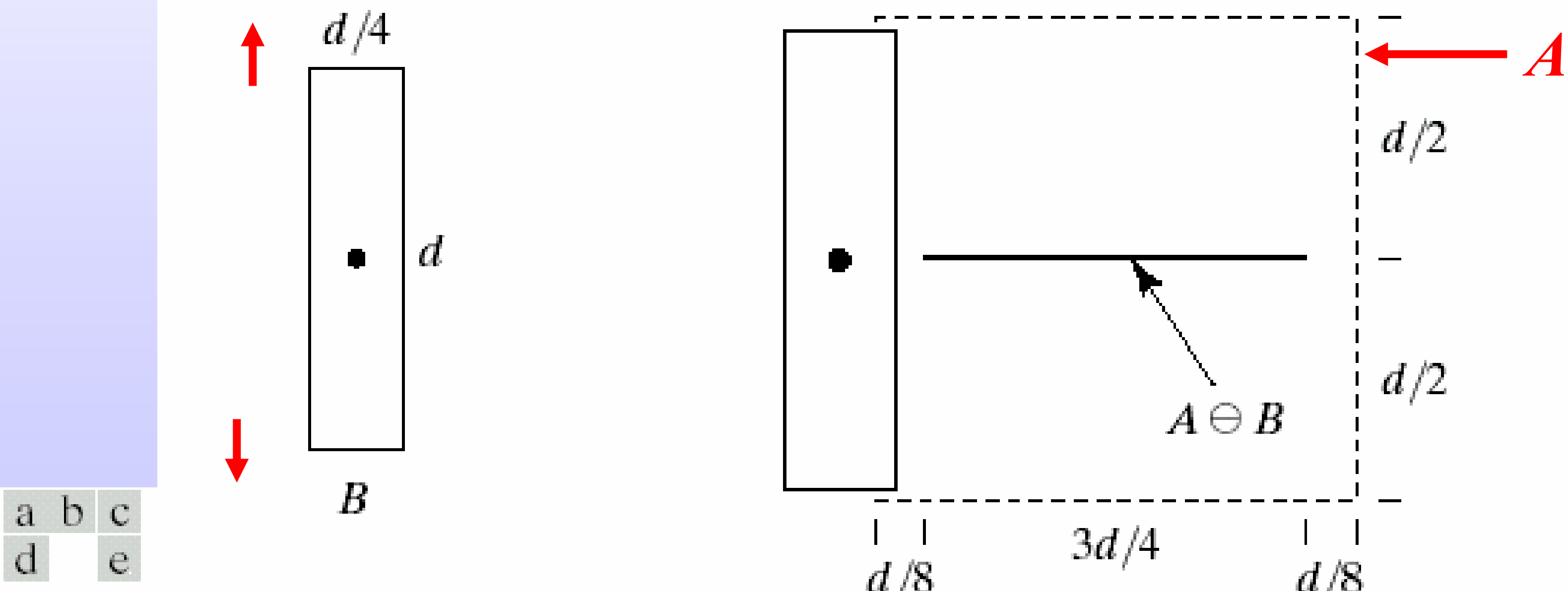
$$A \oplus B = \left\{ z \mid \left[(\hat{B})_z \cap A \right] \subseteq A \right\}$$

- Dilation and erosion are duals of each other with respect to set complementation and reflection. That is,

$$\left(A \ominus B \right)^c = A^c \oplus \hat{B}$$



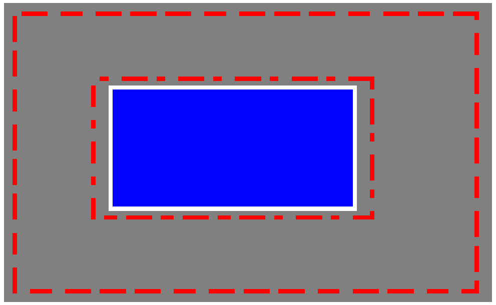
$$A \ominus B = \left\{ z \mid (B)_z \subseteq A \right\}$$



(a) Set A. (b) Square structuring element. (c) Erosion of A by B, shown shaded. (d) Elongated structuring element. (e) Erosion of A using this element.

5 Morphology –Erosion

- The erosion morphological operation generates an output image 'g' from an input image 'f' using a structuring element 'h' where: Place(shift) the center(origin) of 'h' at(to) (x, y)



$$g(x,y)=\begin{cases}1, & \text{if } h \text{ completly falls in } f\\ 0, & \text{else}\end{cases}$$

- The effect of an erosion with 3 x 3 mask is to strip a single layer of pixels from the outer edge of an object and to increase by a single layer of pixels to holes in the object.
- A 5 x 5 mask will strip off two layers of pixels which is equivalent to applying a 3 x 3 mask twice.

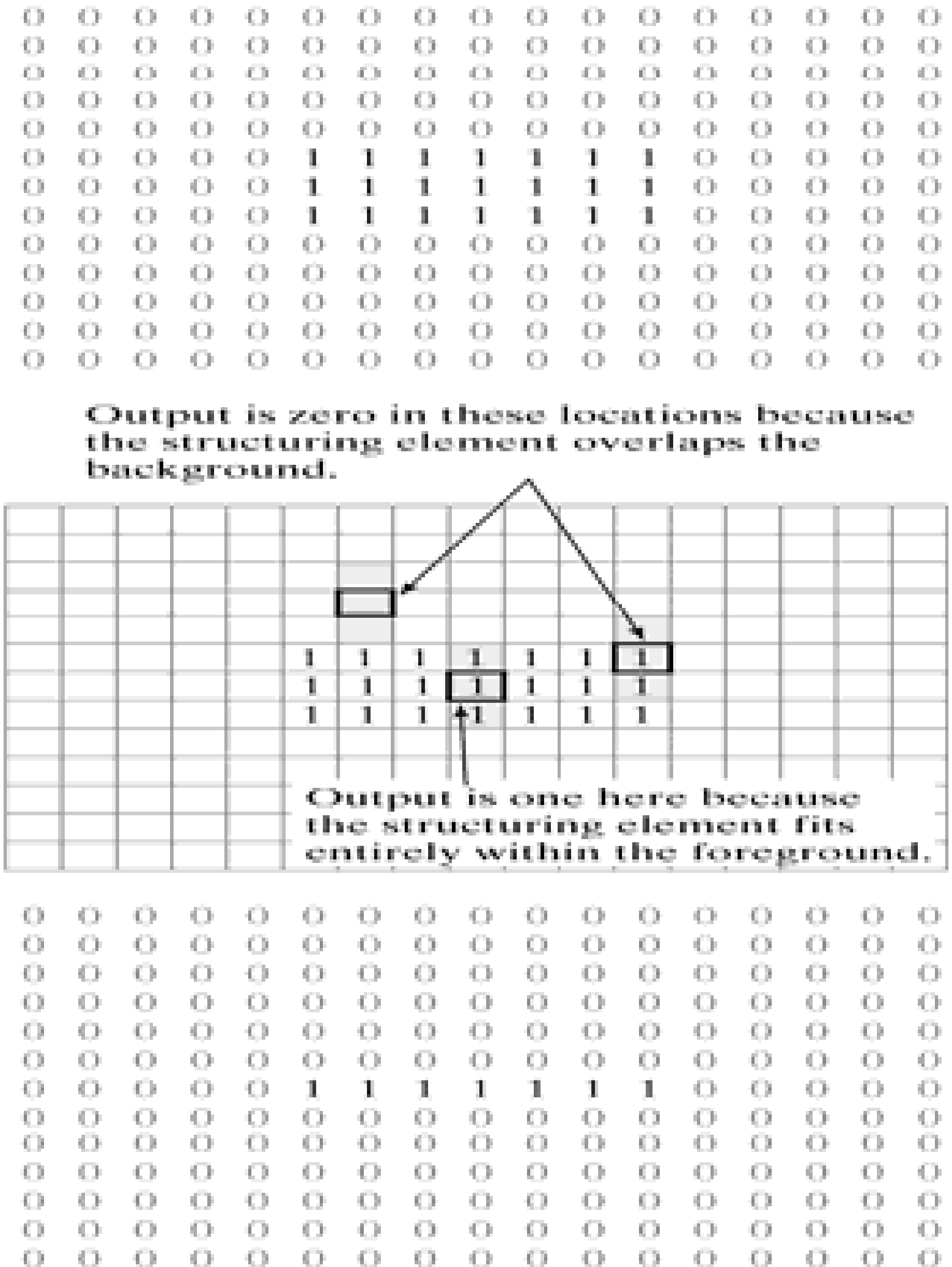
- The main application of erosion is to remove small noise artifacts from an image.

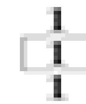
5 Morphology –Erosion

a b c

(a) Image of squares of size 1, 3, 5, 7, 9, and 15 pixels on the side. (b) Erosion of (a) with a square structuring element of 1's, 13 pixels on the side. (c) Dilation of (b) with the same structuring element.

Erosion

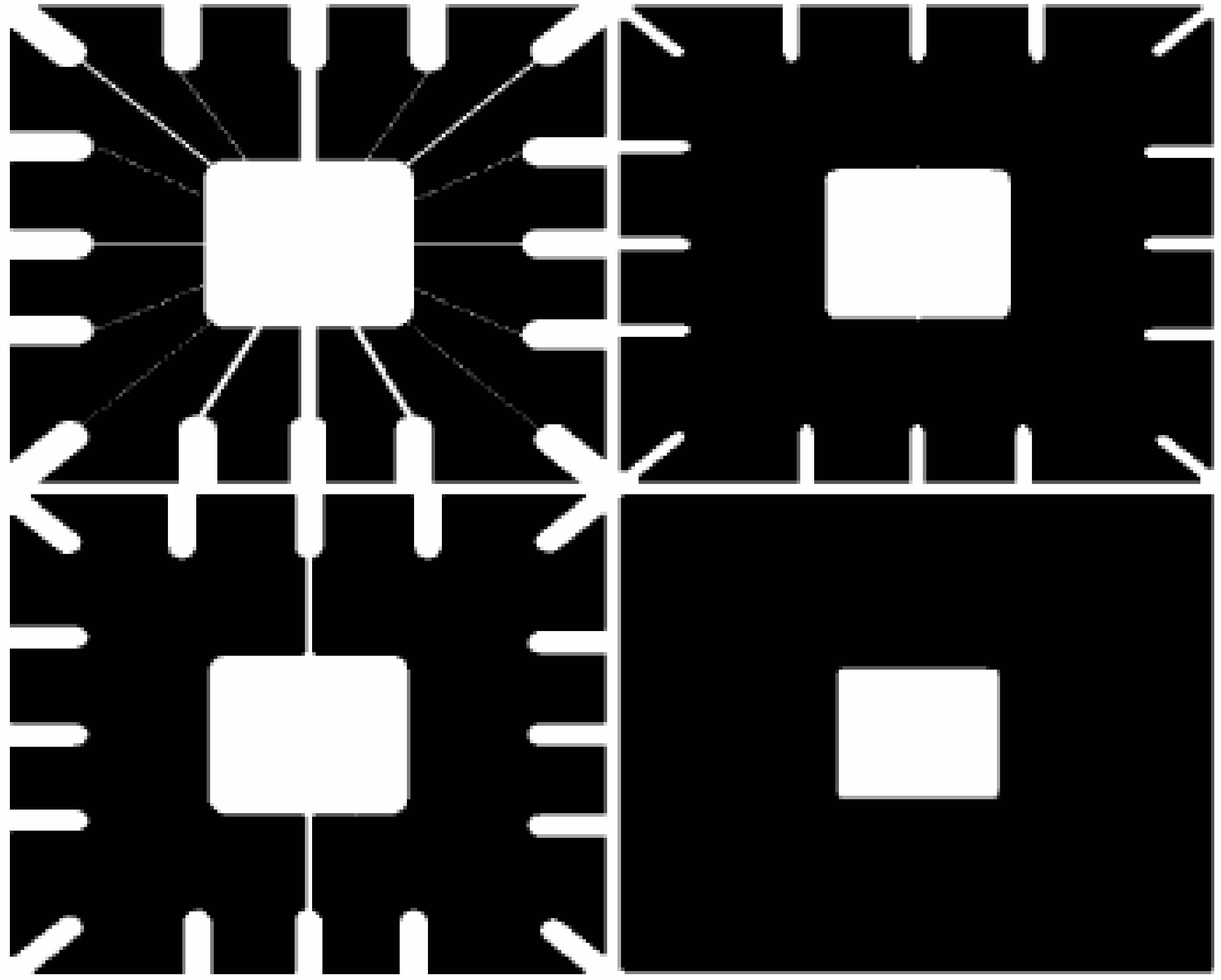




a b
c
d

FIGURE 9.7 Illustration of erosion. (a) Original image with rectangular object. (b) Structuring element with three pixels arranged in a vertical line. The origin of the structuring element is shown with a dark border. (c) Structuring element translated to several locations on the image. (d) Output image.

5 Morphology –Erosion



a	b
c	d

An illustration of erosion. (a) Original image. (b) Erosion with a disk of radius 10. (c) Erosion with a disk of radius 5. (d) Erosion with a disk of radius 20.

5 Morphology –Opening

- Compound operations – Opening

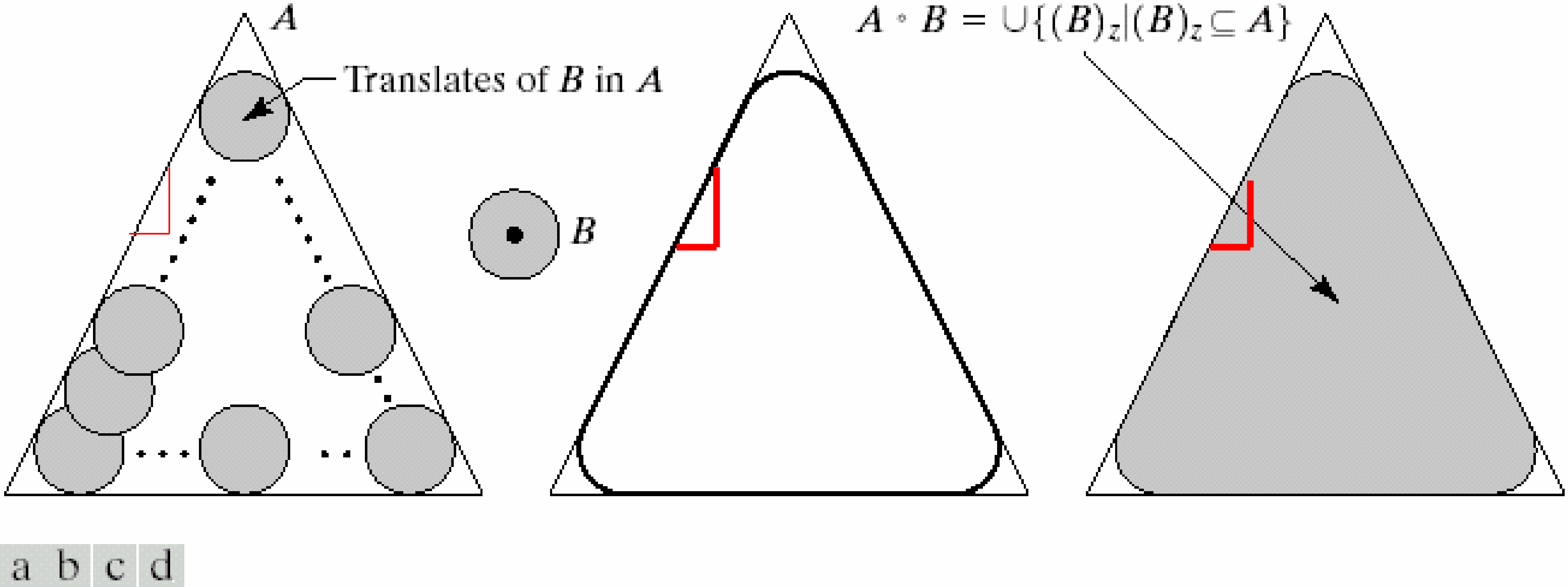
- A compound operation is when two or more morphological operations are performed in succession. A common example is opening which is an erosion followed by a dilation:

$$A \circ B = \left(A \ominus B \right) \oplus B$$

➢ The opening A by B is obtained by taking the union of all translates of B that fit into A. This can be expressed as a fitting processing such that:

$$A \circ B = \bigcup \{ (B)_z \mid (B)_z \subseteq A \} \qquad A \ominus B = \left\{ z \mid \left(B \right)_z \subseteq A \right\}$$

5 Morphology –Opening



(a) Structuring element B "rolling" along the inner boundary of A (the dot indicates the origin of B). (c) The heavy line is the outer boundary of the opening. (d) Complete opening (shaded).

- Note that the outward pointing corners are rounded, where the inward pointing corners remain unchanged.

5 Morphology –Opening

$$A \circ B = \left(A \ominus B\right) \oplus B \qquad A \circ B = \bigcup \left\{ (B)_z \mid (B)_z \subseteq A \right\}$$

- Opening is often performed to clear an image of noise whilst retaining the original object size. Care must be taken that the operation does not distort the shape size of the object if this is significant.
- The opening operation tends to flatten the sharp peninsular projections on the object.
- A useful way to see the effects of an opening operation is to look for differences between the original image and the image after opening by projecting these differences onto the original image.

5 Morphology –Closing

- Compound operations – Closing

- Closing is the complementary operation of opening, defined as dilation followed by erosion.

$$A \bullet B = \left(A \oplus B \right) \ominus B$$

➢ Opening and closing are duals of each other as:

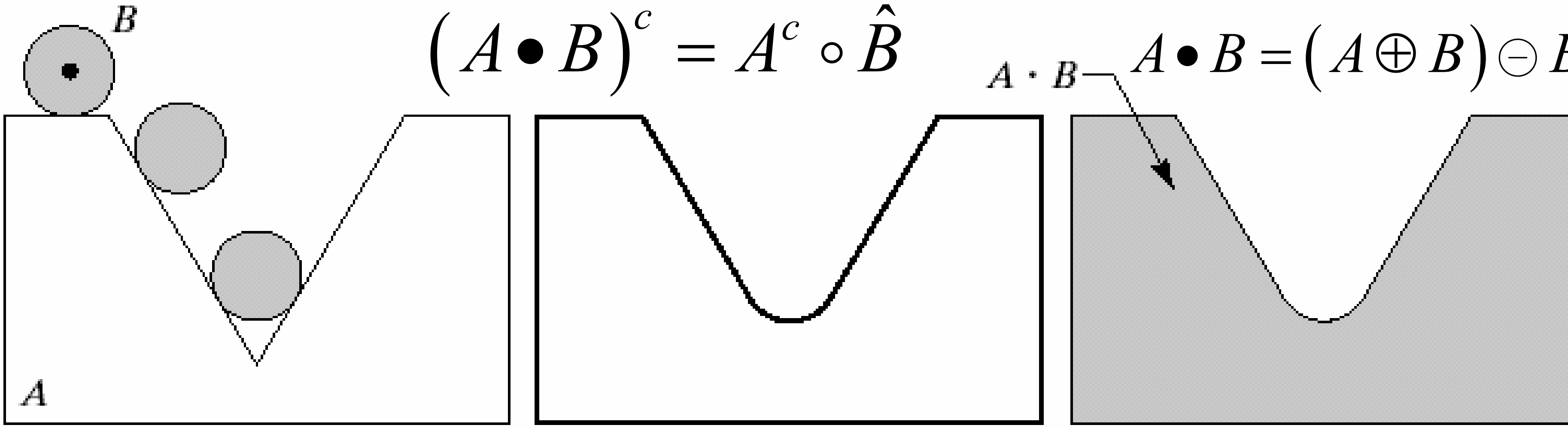
$$\left(A \bullet B \right)^c = A^c \circ \hat{B}$$

$$\text{Or:} \quad A \bullet B = (A^c \circ \hat{B})^c$$

5 Morphology –Closing

$$(A \bullet B)^c = A^c \circ \hat{B}$$

$$A \bullet B = (A \oplus B) \ominus B$$



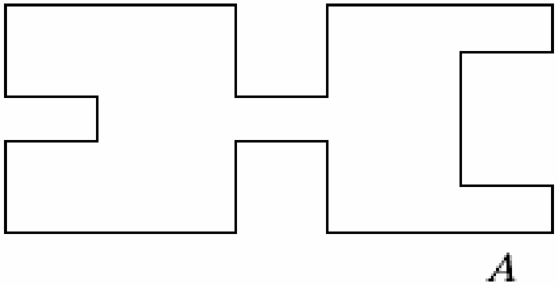
a b c

(a) Structuring element B "rolling" on the outer boundary of set A. (b) Heavy line is the outer boundary of the closing. (c) Complete closing (shaded).

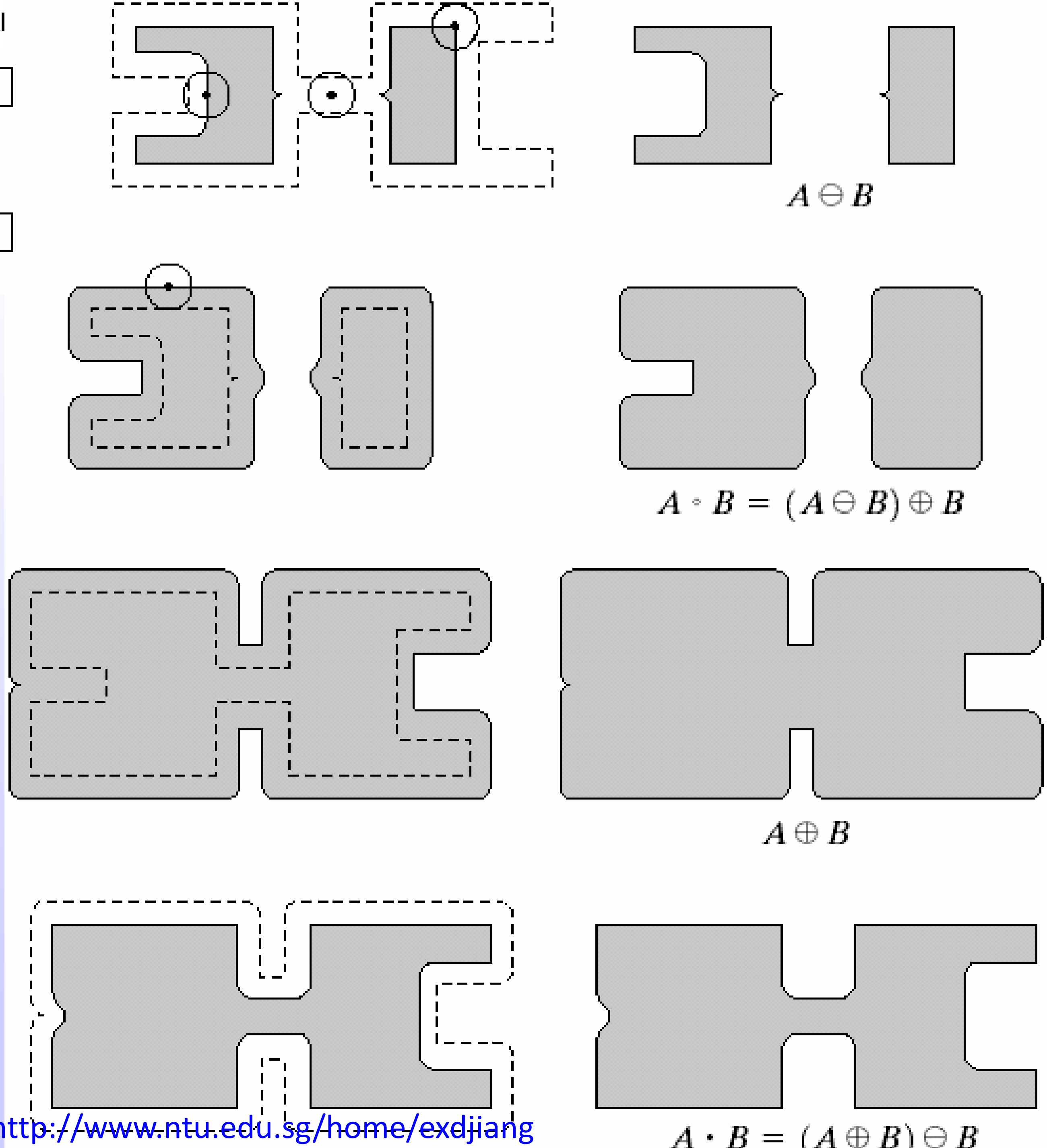
- Note that the inward pointing corners are rounded, where the outward pointing corners remain unchanged.

5 Morphology –Closing

- The classic application of closing is to fill holes in a region whilst retaining the original object size.
- Dilation fills the holes and erosion restores the original region size.
- In addition to filling holes the closing operation tends to fill the 'bays' on the edge of a region.



Examples and Interpretation of erosion, dilation, opening and closing



5 Morphology –Opening and Closing

- The opening operation satisfies the following properties:
  - $A \circ B$ is a subset (subimage) of A
  - If C is a subset of D, then $C \circ B$ is a subset of $D \circ B$
  - $(A \circ B) \circ B = A \circ B$

➢ Similarly, the closing operation satisfies the following properties:

A is a subset (subimage) of $A \bullet B$

If C is a subset of D, then $C \bullet B$ is a subset of $D \bullet B$

$$(A \bullet B) \bullet B = A \bullet B$$

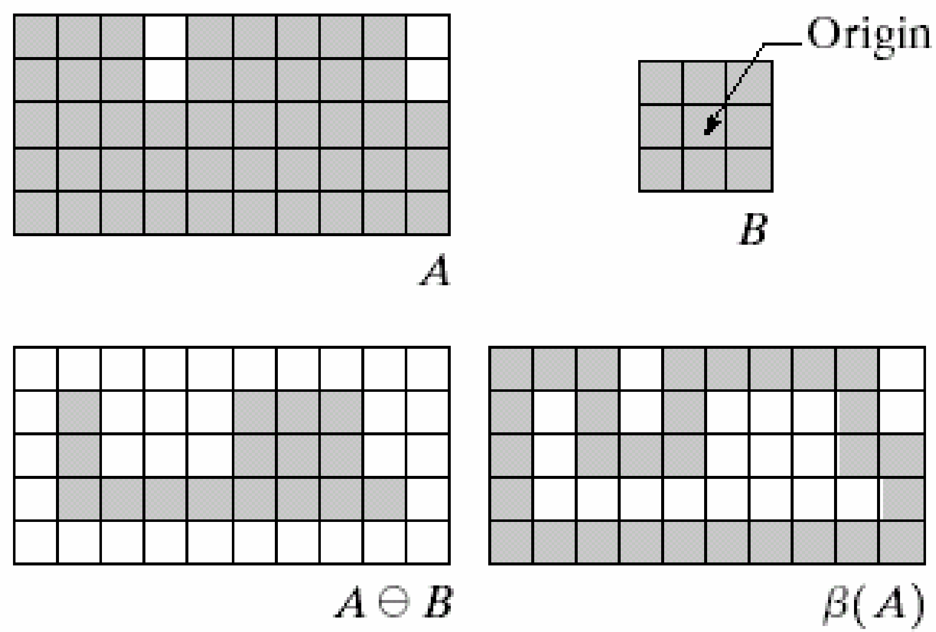
5 Morphology –Algorithms and Applications

- Boundary Extraction:

The boundary of a set A, denoted by $\beta(A)$, can be obtained by: $\beta(A) = A - (A \ominus B)$

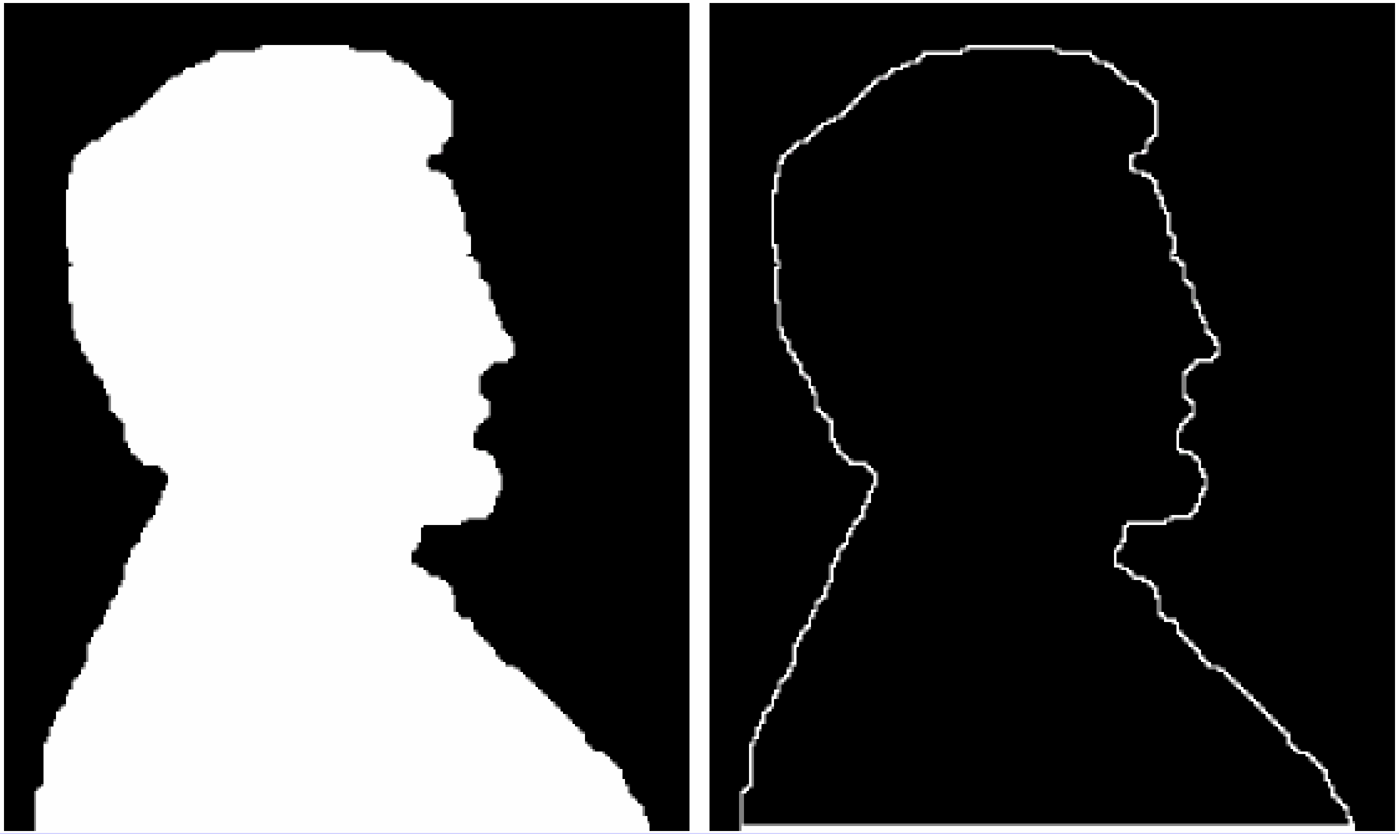
a	b
c	d

(a) Set A. (b) Structuring element B. (c) A eroded by B. (d) Boundary, given by the set difference between A and its erosion.



5 Morphology –Algorithms and Applications

- An example of Boundary Extraction:



a b

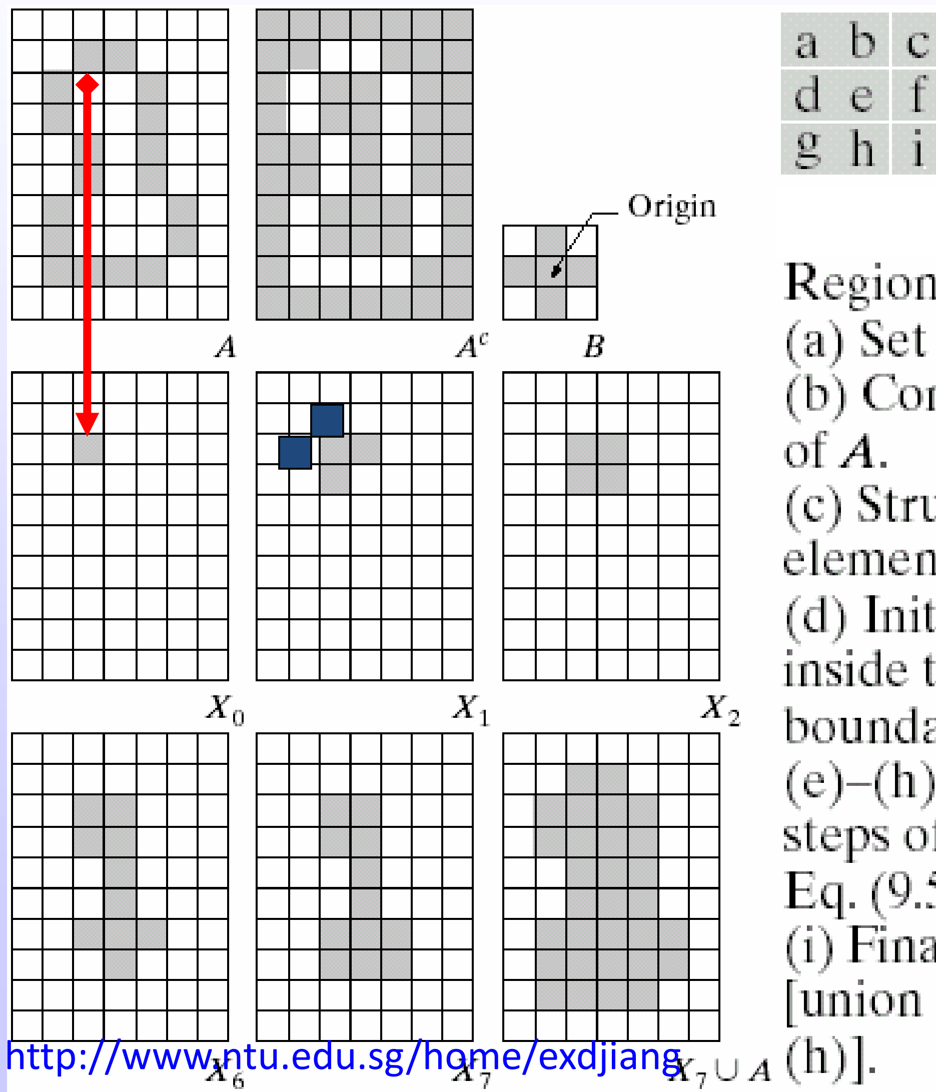
(a) A simple binary image, with 1's represented in white. (b) Result of using Eq. (9.5-1) with the structuring element in Fig. 9.13(b).

5 Morphology –Algorithms and Applications

- Region Filling: $X_k = \left(X_{k-1} \oplus B\right) \cap A^c, \quad k = 1, 2, 3, \ldots\ldots$

$$A^F = X_k \cup A$$

Beginning with a point X_0 inside the boundary, the entire region inside the boundary is filled by the above procedure.

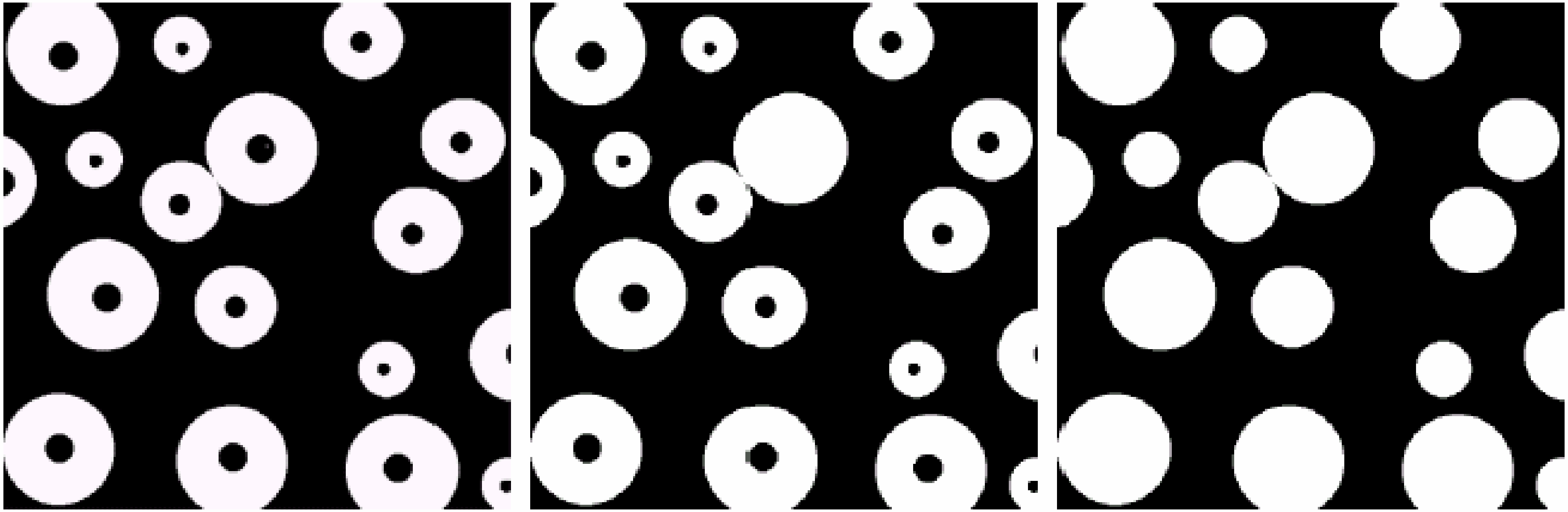


Region filling. (a) Set $\boldsymbol{A}$. (b) Complement of $\boldsymbol{A}$. (c) Structuring element $\boldsymbol{B}$. (d) Initial point inside the boundary. (e)–(h) Various steps of Eq. (9.5-2). (i) Final result [union of (a) and (h)].

5 Morphology –Algorithms and Applications

- An example of Region Filling: $A^F = X_k \bigcup A$

$$X_k = \left(X_{k-1} \oplus B\right) \bigcap A^c, \ k = 1, 2, 3, \ldots$$

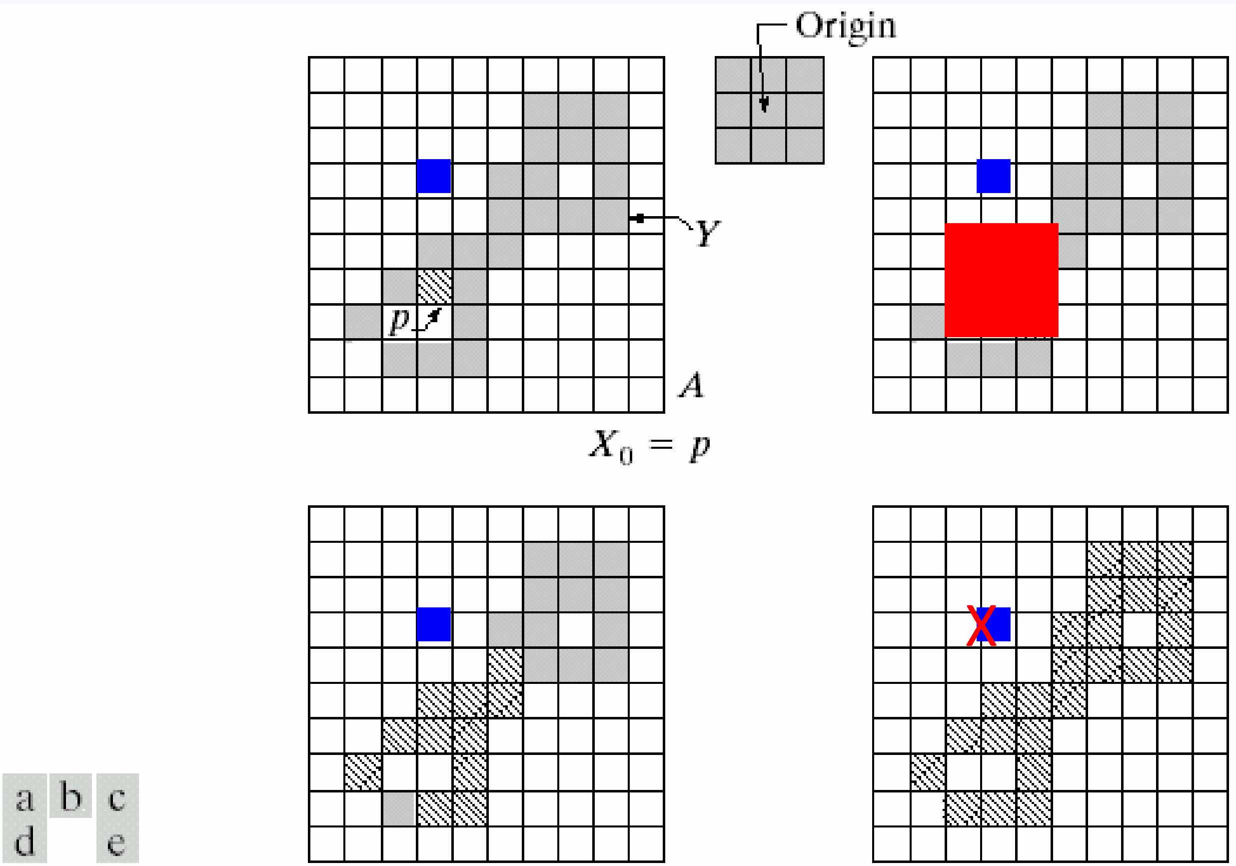


a b c

(a) Binary image (the white dot inside one of the regions is the starting point for the region-filling algorithm). (b) Result of filling that region (c) Result of filling all regions.

5 Morphology –Algorithms and Applications

- Extract connected components: $X_k = (X_{k-1} \oplus B) \cap A,\ k = 1, 2, 3, \ldots$



(a) Set A showing initial point p (all shaded points are valued 1, but are shown different from p to indicate that they have not yet been found by the algorithm). (b) Structuring element. (c) Result of first iterative step. (d) Result of second step. (e) Final result.

5 Morphology –Algorithms and Applications

a
b
c d

(a) X-ray image of chicken filet with bone fragments. (b) Thresholded image. (c) Image eroded with a 5×5 structuring element of 1's. (d) Number of pixels in the connected components of (c). (Image courtesy of NTB Elektronische Geraete GmbH, Diepholz, Germany, www.ntbxray.com.)

Connected component	No. of pixels in connected comp
01	11
02	9
03	9
04	39
05	133
06	1
07	1
08	743
09	7
10	11
11	11
12	9
13	9
14	674
15	85

5 Morphology –Algorithms and Applications

- Denoising:

$$\left(A \circ B\right) \bullet B$$

- or

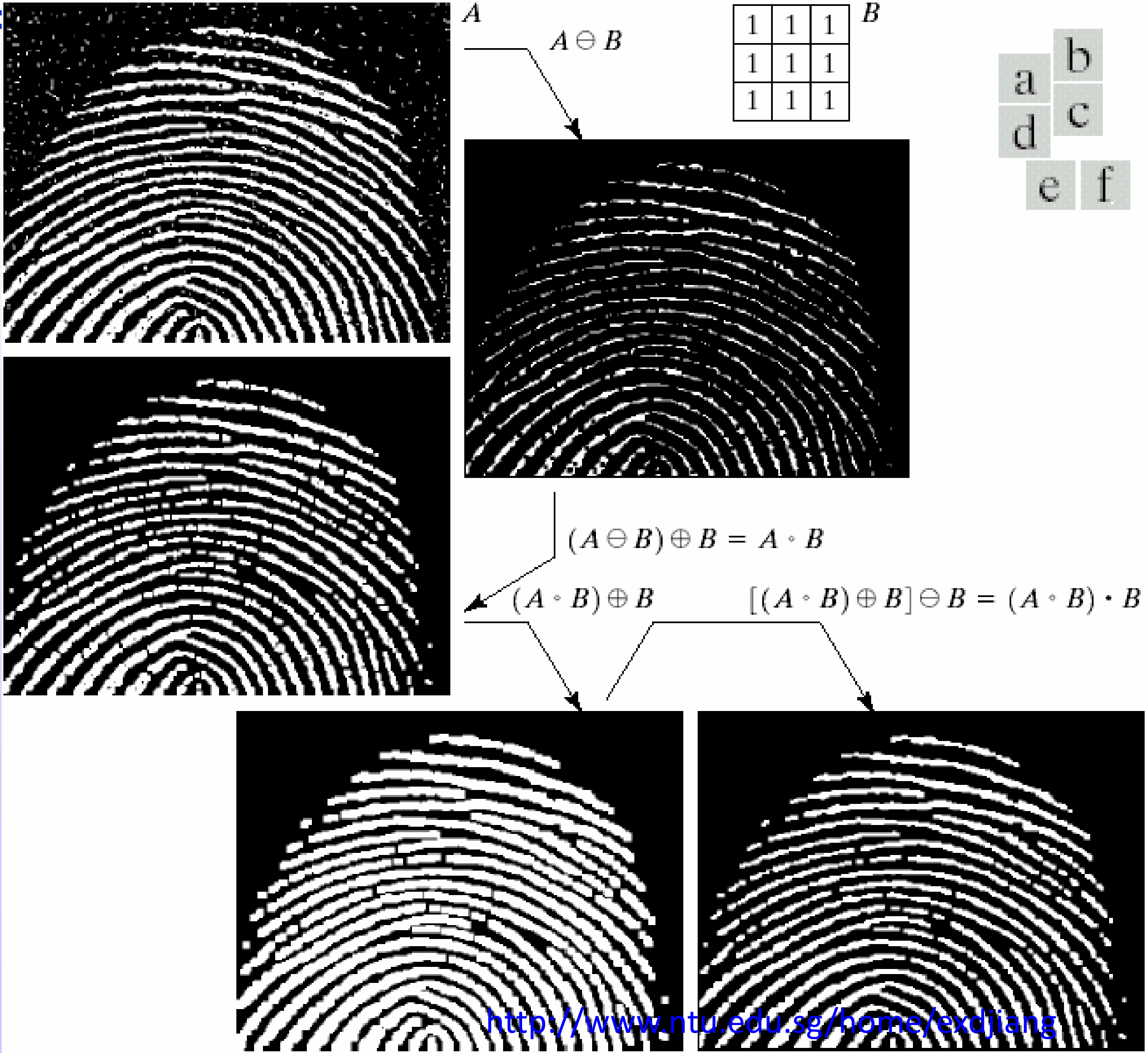
$$\left(A \bullet B\right) \circ B$$

Can be used to eliminate noise and its effect on the object.

➢ Noise pixels outside the object area are removed by opening with B while noise pixels inside the object area are removed by closing with B.

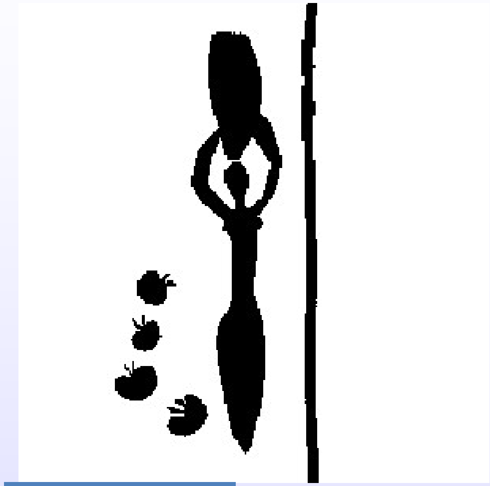
See example in the next slide

5 Morphology –Algorithms and Applications

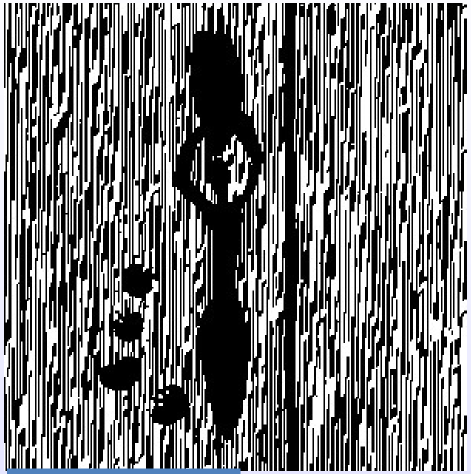


(a) Noisy image. (c) Eroded image. (d) Opening of A. (d) Dilation of the opening. (e) Closing of the opening. (Original image for this example courtesy of the National Institute of Standards and Technology.)

5 Morphology –Algorithms and Applications



ORIGINAL



DEGRADED

FILTERED

Henri Matisse, *Woman with Amphora and Pomegranates*, 1952

5 Morphology –Summary

Operation	Equation	Comments (The Roman numerals refer to the structuring elements shown in Fig. 9.26).
Translation	$(A)_z = \{w \mid w = a + z, \quad \text{for } a \in A\}$	Translates the origin of A to point z.
Reflection	$\hat{B} = \{w \mid w = -b, \quad \text{for } b \in B\}$	Reflects all elements of B about the origin of this set.
Complement	$A^c = \{w \mid w \notin A\}$	Set of points not in A.
Difference	$A - B = \{w \mid w \in A, w \notin B\} = A \cap B^c$	Set of points that belong to A but not to B.
Dilation	$A \oplus B = \{z \mid (\hat{B})_z \cap A \neq \varnothing\}$	"Expands" the boundary of A. (I)
Erosion	$A \ominus B = \{z \mid (B)_z \subseteq A\}$	"Contracts" the boundary of A. (I)

5 Morphology –Summary

Opening	$A \circ B = (A \ominus B) \oplus B$	Smoothes contours, breaks narrow isthmuses, and eliminates small islands and sharp peaks. (I)
Closing	$A \bullet B = (A \oplus B) \ominus B$	Smoothes contours, fuses narrow breaks and long thin gulfs, and eliminates small holes. (I)
Boundary extraction	$\beta(A) = A - (A \ominus B)$	Set of points on the boundary of set A. (I)
Region filling	$X_k = (X_{k-1} \oplus B) \cap A^c$; $X_0 = p$ and $k = 1, 2, 3, \ldots$	Fills a region in A, given a point p in the region. (II)
Connected components	$X_k = (X_{k-1} \oplus B) \cap A$; $X_0 = p$ and $k = 1, 2, 3, \ldots$	Finds a connected component Y in A, given a point p in Y. (I)

6-7. Image Analysis –Outline

- Introduction
- Image Segmentation
- Edge Detection
  - First-order derivative detector
  - Second-order derivative detector
- Hough Transform
  - Edge Linking
- Image Local Dominant Orientation Analysis

6-7. Image Analysis –Introduction

Image Analysis

Extract information from the image

Understand the content of the image
Recognize the meaning of image

Feature Extraction

- Edges and boundaries
- Spatial features
- Transform features
- Shape
- Texture
- Color

Segmentation

- Thresholding
- Region-based techniques
- Quad-trees
- Edge-based techniques

Classification

- Statistical classifiers
- Bayes Decision Theory
- Supervised learning
- Unsupervised learning

6 Image Analysis –Segmentation

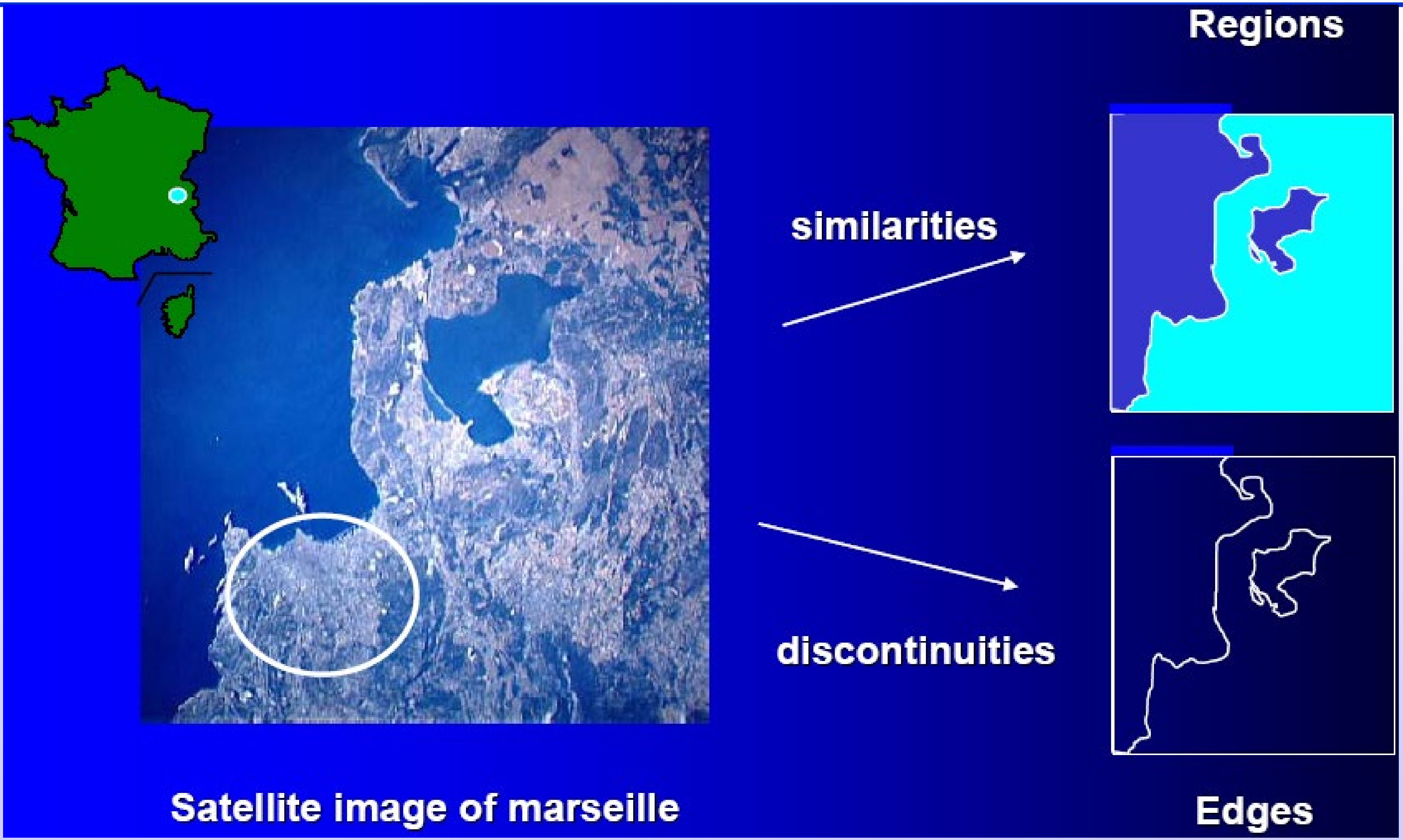
- Segmentation is to subdivide an image into its constituent regions or objects.
- Segmentation should stop when the objects of interest in an application have been isolated.

➢ Segmentation algorithms generally are based on one of 2 basic properties of intensity values

similarity: to partition an image into regions that are similar according to a set of predefined criteria. (Thresholding)

discontinuity: to partition an image based on abrupt changes in intensity (Point, Line and Edge Detection)

6 Image Analysis –Segmentation



6 Image Analysis –Segmentation by Thresholding

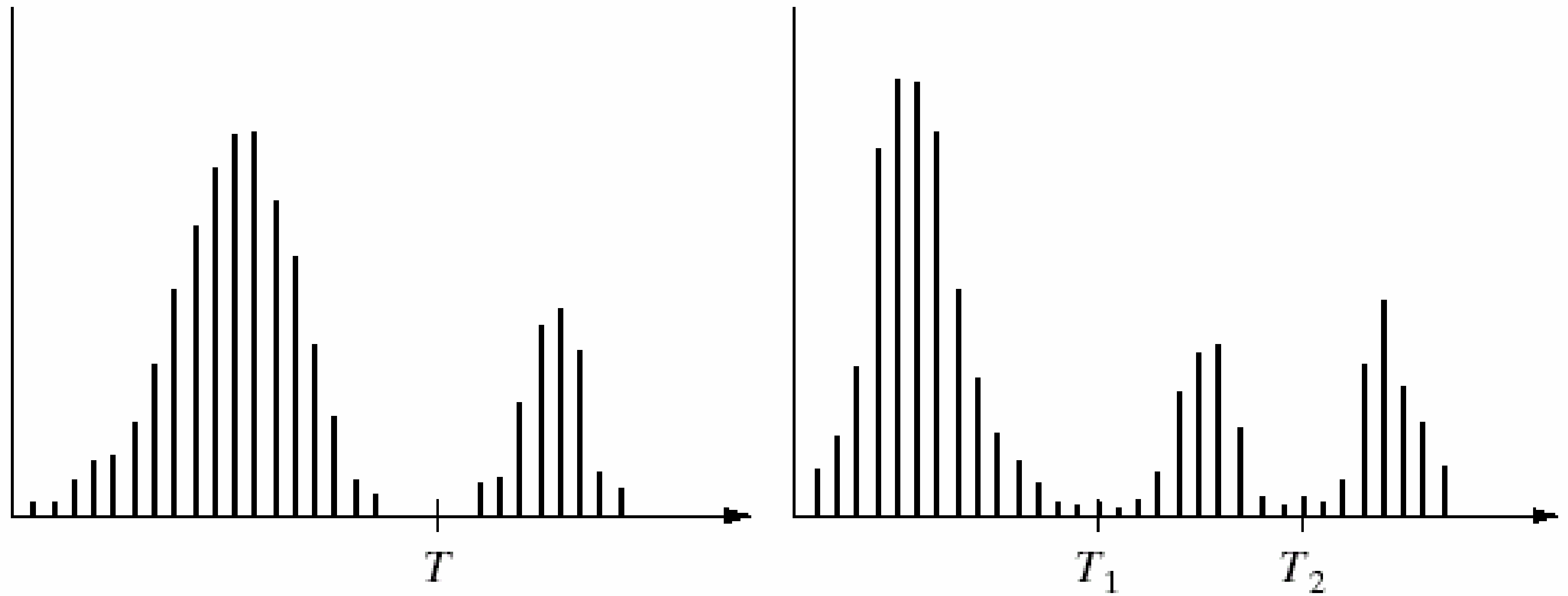
- Many images contain some objects of interest of uniform brightness placed against a background of differing brightness.

- Typical examples include handwritten and typewritten text, microscopic biomedical samples, fingerprints, and airplanes on a runway.

6 Image Analysis –Segmentation by Thresholding

image with dark background and a light object

image with dark background and two light objects



a b

(a) Gray-level histograms that can be partitioned by (a) a single threshold, and (b) multiple thresholds.

6 Image Analysis –Segmentation by Thresholding

- A thresholded image $g(x, y)$ is defined as

$$g(x,y)=\begin{cases}1 & \text{if } f(x,y)>T\\ 0 & \text{if } f(x,y)\le T\end{cases}$$

where T is the threshold given by

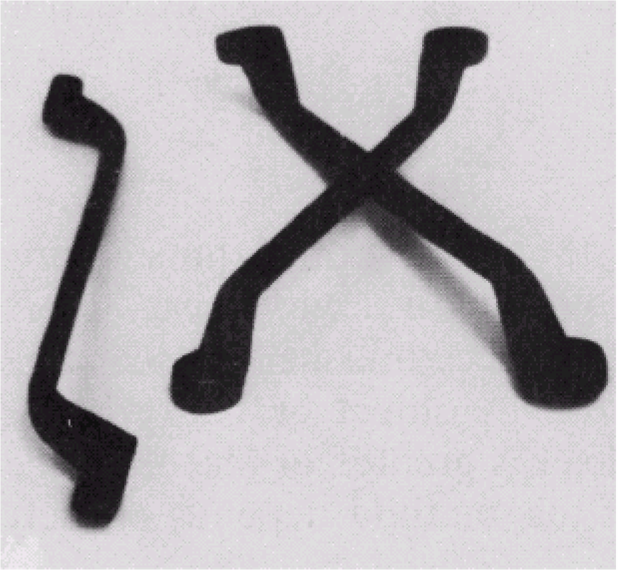
$$T = T[x, y, p(x,y), f(x,y)]$$

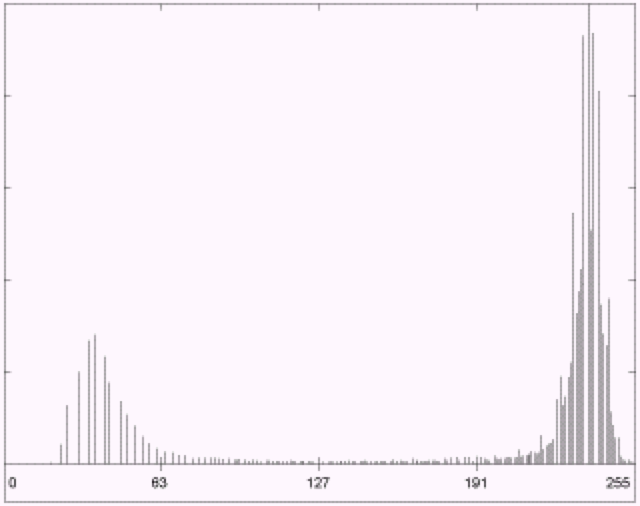
- Global threshold: T depends on gray-level values $f(x, y)$ of the whole image alone
- Local threshold: T depends on both $f(x, y)$ and its local neighbors property $p(x, y)$
- Adaptive threshold: T depends on x and y coordinates

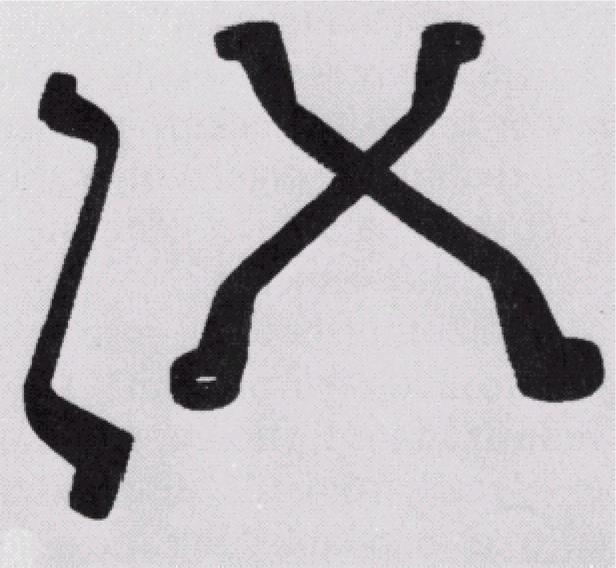
6 Image Analysis –Segmentation by Thresholding

Basic Global Thresholding

Use T midway between the max and min gray levels generate binary image







6 Image Analysis –Segmentation by Thresholding

- Heuristic approach to get global threshold T:

1. Select an initial estimate for T.
2. Segment the image using T. This will produce two groups of pixels: G_1 consisting of all pixels with gray level values > T and G_2 consisting of pixels with gray level values $\leq T$
3. Compute the average gray level values μ_1 and μ_2 for the pixels in regions G_1 and G_2
4. Compute a new threshold value $T = 0.5\ (\mu_1 + \mu_2)$
5. Repeat steps 2 through 4 until the difference in T in successive iterations is smaller than a predefined parameter T_o.

6 Image Analysis –Segmentation by Thresholding

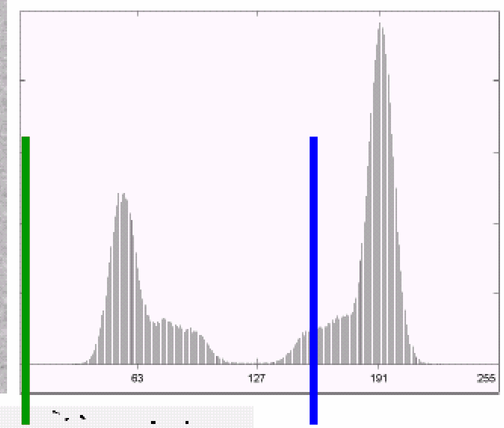
Example of

Heuristic approach:

$T_0 = 0$

3 iterations

with result $T = 125$



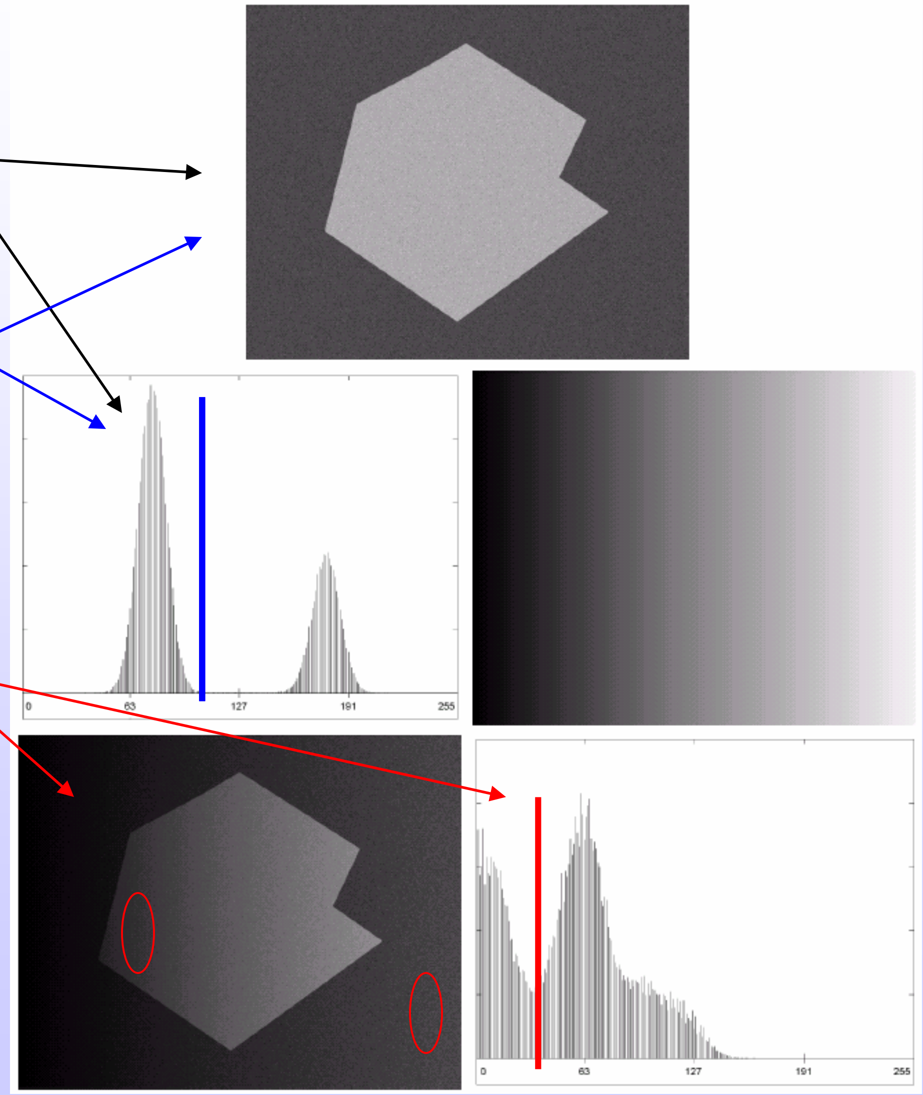
6 Image Analysis –Segmentation by Thresholding

If Object and background are separated in the grey value,

easily use global thresholding

If the grey value of object and background are overlapped, Difficult to segment using global thresholding

Solution: Adaptive local thresholding



6 Image Analysis –Segmentation by Thresholding

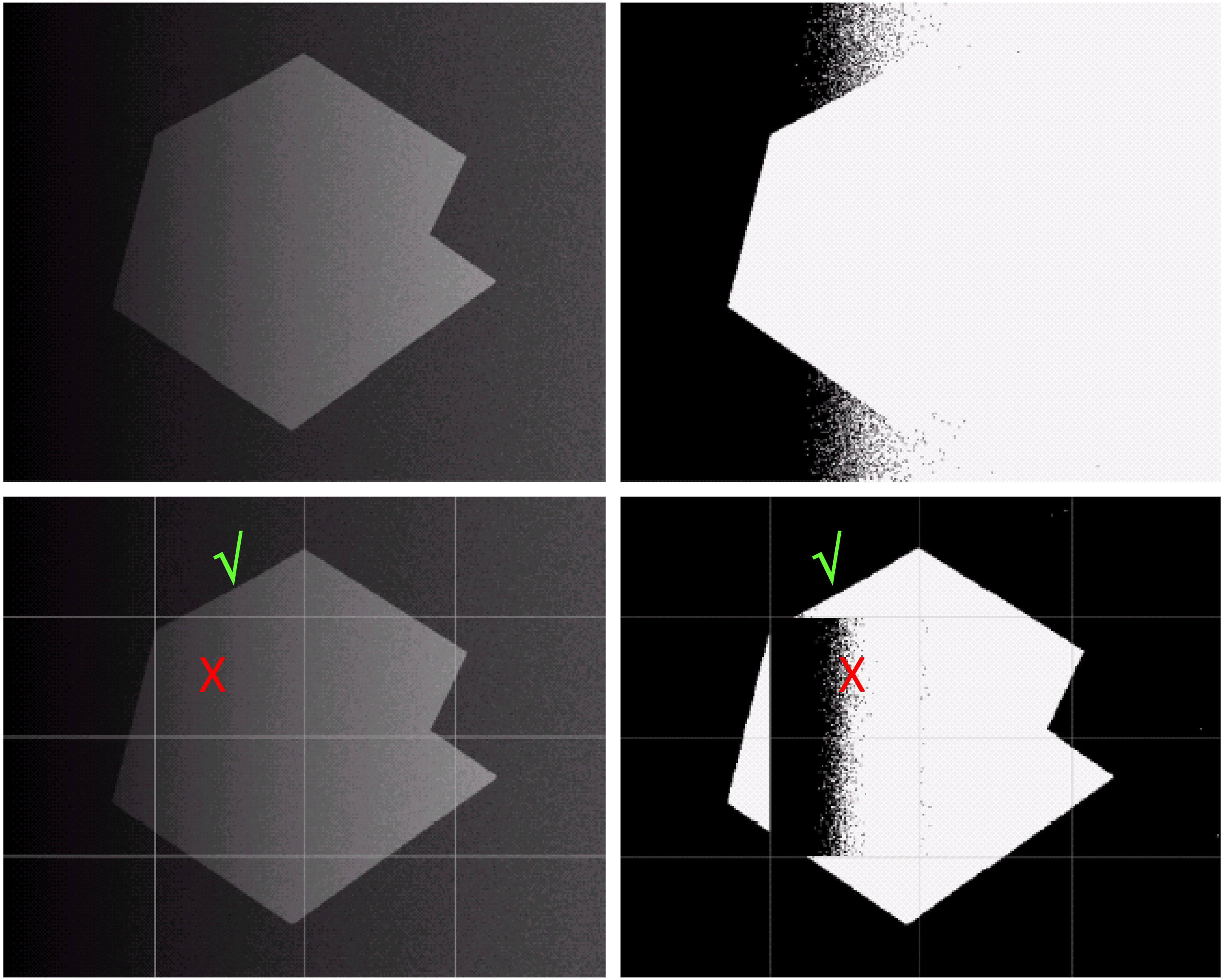
- Adaptive local thresholding:

- Subdivide original image into small areas.
- Utilize a different threshold to segment different sub-images.
- Since the threshold used for each pixel depends on the location of the pixel in terms of the sub-images, this type of thresholding is adaptive.

6 Image Analysis –Segmentation by Thresholding

a b
c d

FIGURE 10.30 (a) Original image. (b) Result of global thresholding. (c) Image subdivided into individual subimages. (d) Result of adaptive thresholding.



6 Image Analysis –Segmentation by Thresholding

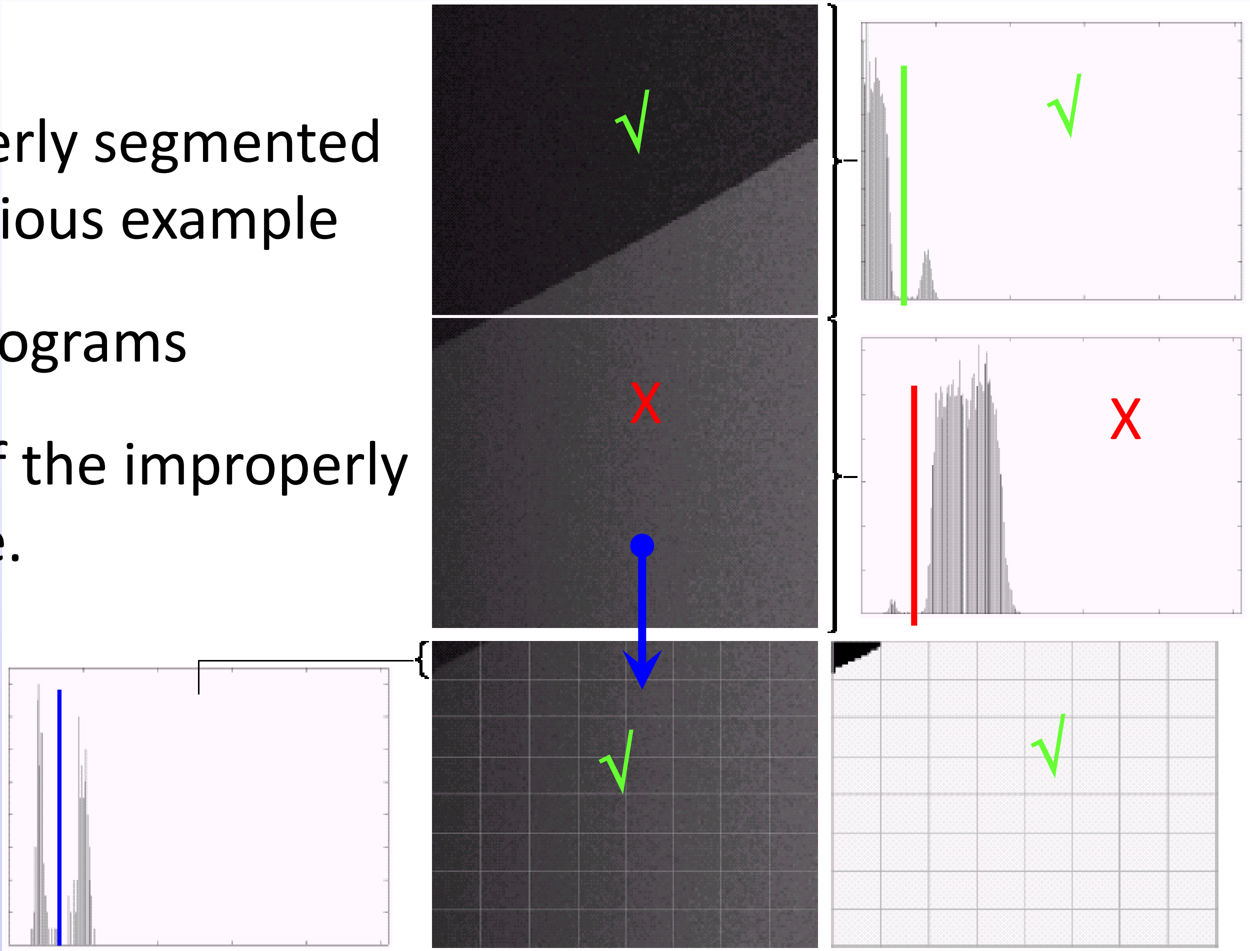
Further subdivision:

a) properly and improperly segmented subimages from previous example

b)-c) corresponding histograms

d) further subdivision of the improperly segmented subimage.

e) histogram of small subimage at top



f) result of adaptively segmenting d)

6 Image Analysis –Optimal Thresholding

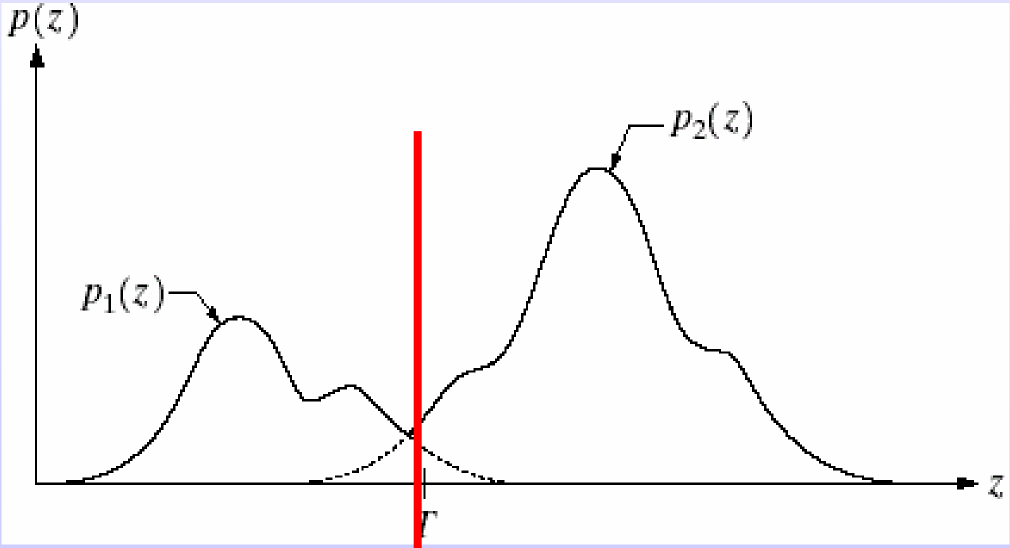
- Objective:

  Minimize the average error in making decisions that a given pixel belongs to an object or the background

  - Assumptions:

    Image contains only 2 gray-level regions.

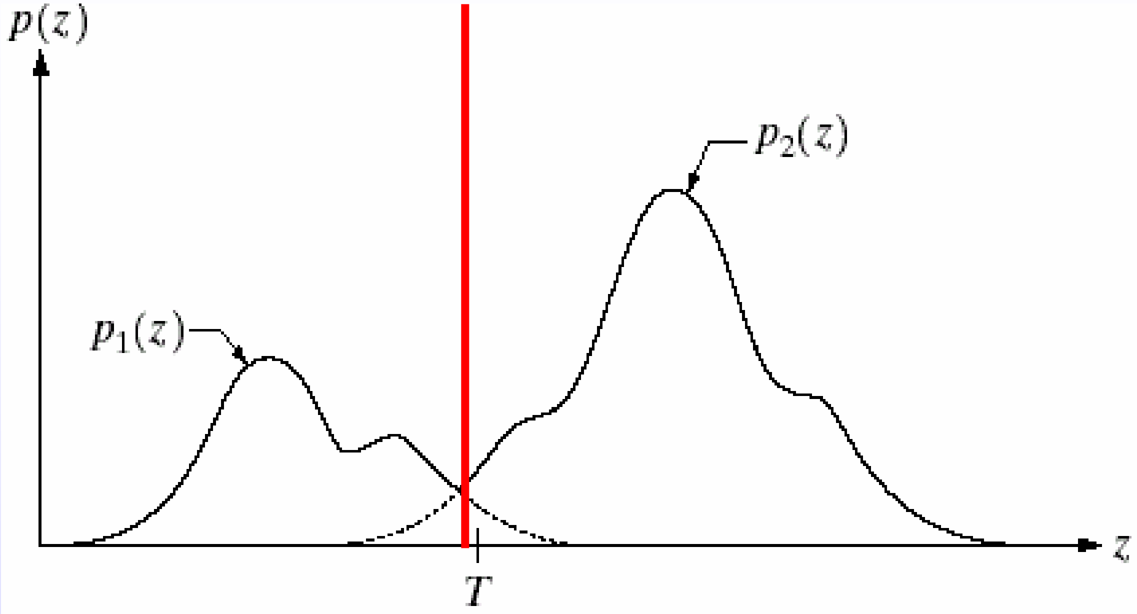
    $p_1(z)$ and $p_2(z)$ are the probability density functions of grey level z for region 1 (object) and 2 (background) respectively



6 Image Analysis –Optimal Thresholding

- Probability of error in classifying a background point as an object:

$$E_1(T) = \int_{-\infty}^{T} p_2(z)dz$$



- Probability of error in classifying an object point as background:

$$E_2(T) = \int_{T}^{\infty} p_1(z)dz$$

6 Image Analysis –Optimal Thresholding

Mixture pdf of the overall image:

$$p(z) = P_1 p_1(z) + P_2 p_2(z)$$

- Assume any pixel belongs to either object or background:

$$P_1 + P_2 = 1$$

- Overall probability of error:

$$E(T) = P_2 E_1(T) + P_1 E_2(T)$$

6 Image Analysis –Optimal Thresholding

- To minimize the error, differentiate $E(T)$ with respect to T and let the result equal to 0

$$\frac{dE(T)}{dT} = \frac{d(P_2 E_1(T) + P_1 E_2(T))}{dT} = 0$$

- Find T which makes $$P_1 p_1(T) = P_2 p_2(T)$$

- If $P_1 = P_2$, the optimum threshold is where the curve $p_1(z)$ and $p_2(z)$ intersect.

6 Image Analysis –Optimal Thresholding

- Assuming both $p_1(z)$ and $p_2(z)$ follow Gaussian distribution:

$$p(z)=\frac{P_1}{\sqrt{2\pi}\sigma_1}e^{-\frac{(z-\mu_1)^2}{2\sigma_1^2}}+\frac{P_2}{\sqrt{2\pi}\sigma_2}e^{-\frac{(z-\mu_2)^2}{2\sigma_2^2}}$$

where
μ_1 and σ_1^2 are the mean and variance of the Gaussian density of one object.
μ_2 and σ_2^2 are the mean and variance of the Gaussian density of the other object.

6 Image Analysis –Optimal Thresholding

- The optimum T is obtained by solve:

$$P_1 p_1(T) = P_2 p_2(T) \Rightarrow \frac{P_1}{\sqrt{2\pi}\sigma_1} e^{-\frac{(T-\mu_1)^2}{2\sigma_1^2}} = \frac{P_2}{\sqrt{2\pi}\sigma_2} e^{-\frac{(T-\mu_2)^2}{2\sigma_2^2}}$$

- This results in a quadratic equation:

$$AT^2 + BT + C = 0$$

$$\text{where } A = \sigma_1^2 - \sigma_2^2, \quad B = 2(\mu_1\sigma_2^2 - \mu_2\sigma_1^2)$$

$$C = \sigma_1^2\mu_2^2 - \sigma_2^2\mu_1^2 + 2\sigma_1^2\sigma_2^2 \ln(\sigma_2 P_1 / \sigma_1 P_2)$$

6 Image Analysis –Optimal Thresholding

- If $\sigma_1 = \sigma_2 = \sigma$, the optimum threshold is simply obtained by

$$T = \frac{\mu_1 + \mu_2}{2} + \frac{\sigma^2}{\mu_1 - \mu_2} \ln\left(\frac{P_2}{P_1}\right)$$

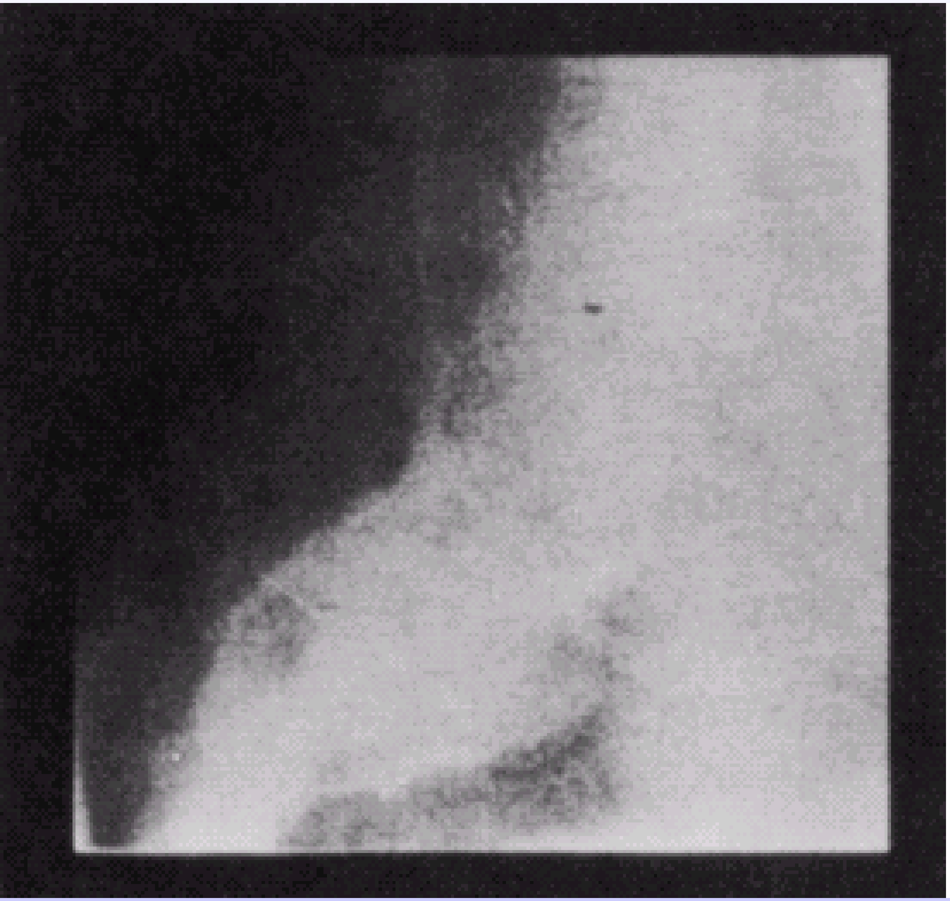
- if $P_1 = P_2$, then the optimal threshold is the average of the two means

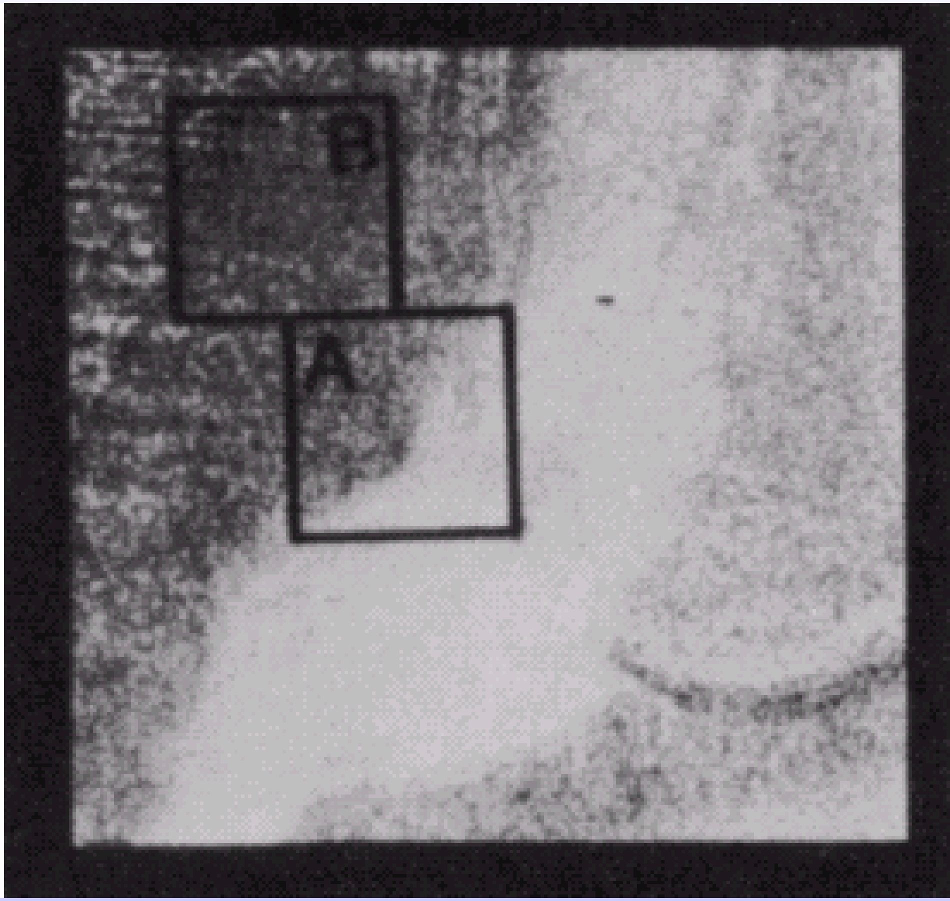
$$T = \frac{\mu_1 + \mu_2}{2}$$

6 Image Analysis –Optimal Thresholding

- Example:

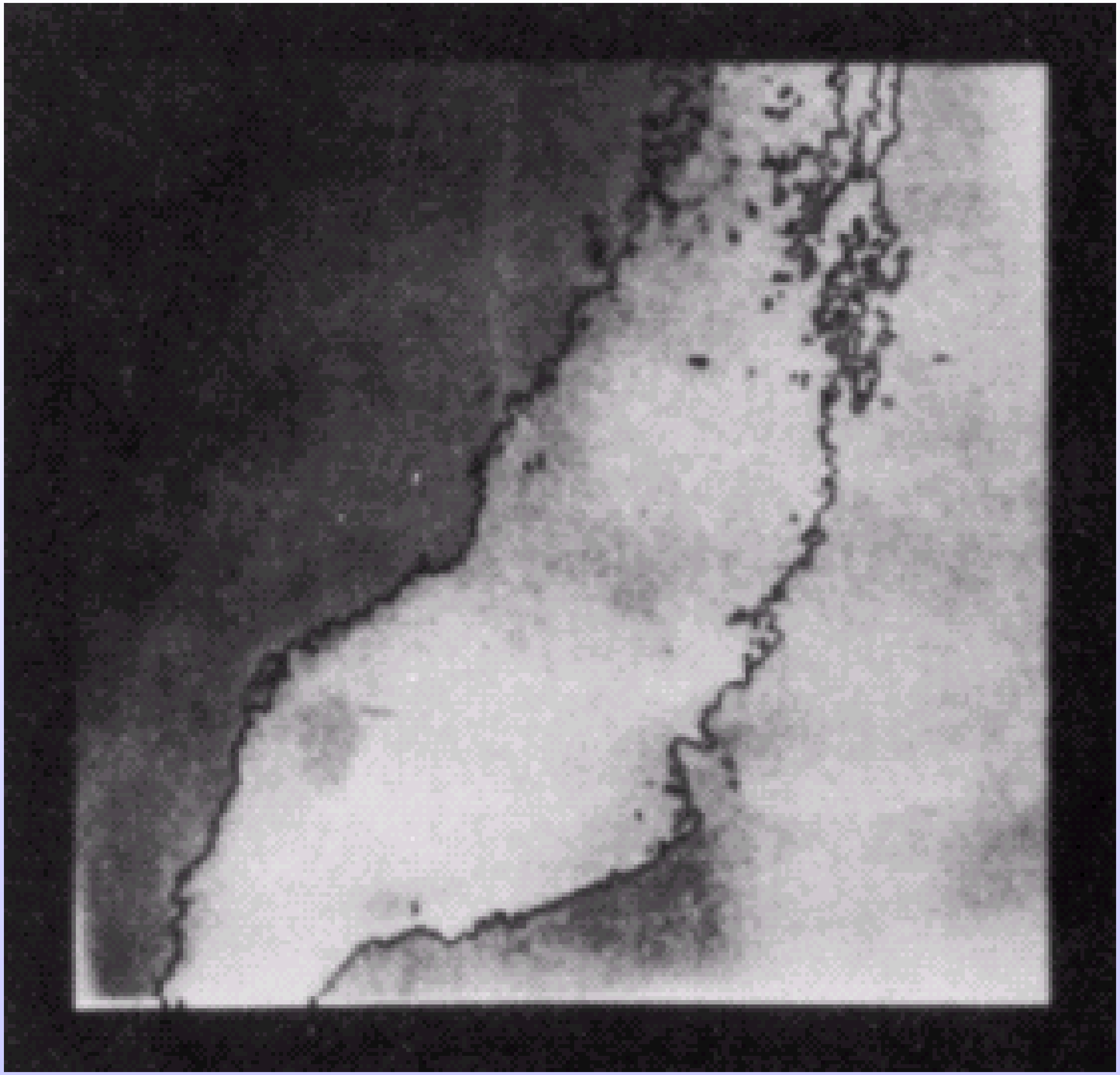
cardioangiogram before and after preprocessing.





6 Image Analysis –Optimal Thresholding

Example:



Cardioangiogram showing superimposed boundaries.

6 Image Analysis –Detection of Discontinuities

- Detect the three basic types of gray-level discontinuities (abrupt changes in intensity)
  - – Point detection
  - – Line detection
  - – Edge detection

- The simple way is to run a mask through the image

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

6 Image Analysis –Point Detection

$$R = \sum_{i=1}^{9} w_i z_i$$

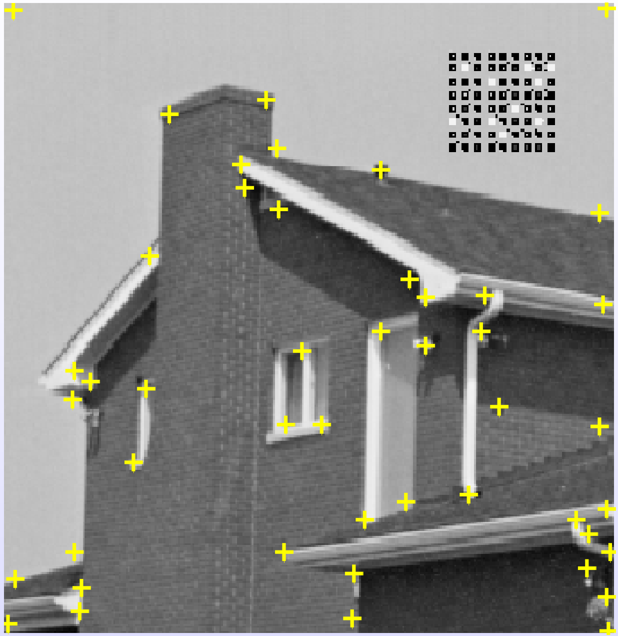
−1	−1	−1
−1	8	−1
−1	−1	−1

- The formulation measures the weighted difference between the center point and its neighbors.
- A point has been detected at the location on which the mask is centered if

$$|R| \geq T$$

6 Image Analysis –Point Detection example

- Point in larger scales– blob and corner detection: Multiscale interesting point detection with scale selection: refer to

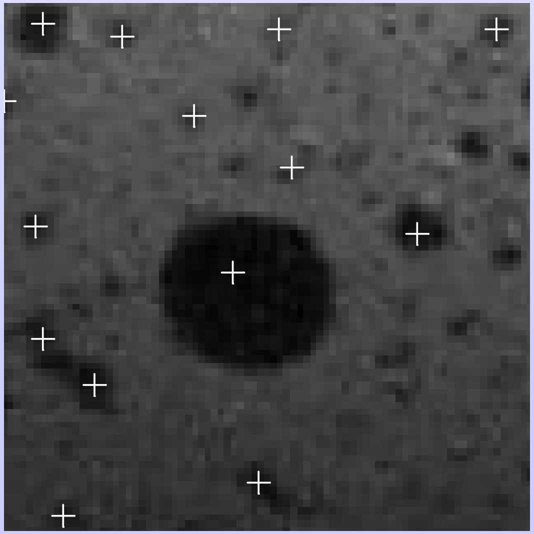


C. Geng and X.D. Jiang, "Face Recognition Based on the Multi-scale Local Image Structures," ***Pattern Recognition***, vol. 44, no. 10-11, pp. 2565-2575, October-November 2011.

C. Geng and X.D. Jiang, "Fully Automatic Face Recognition Framework based on Local and Global Features," ***Machine Vision and Applications***, vol. 24, no. 3, pp. 537-549, April 2013.

Z. Miao and X.D. Jiang, "Interest Point Detection Using Rank Order LoG Filter," ***Pattern Recognition***, vol. 46, no. 11, pp. 2890-2901, November 2013.

Z. Miao, X.D. Jiang and K. Yap, "Contrast Invariant Interest Point Detection by Zero-Norm LoG Filter," ***IEEE Transactions on Image Processing***, vol. 25, no. 1, pp. 331 - 342, January 2016.



6 Image Analysis –Line Detection

−1	−1	−1
2	2	2
−1	−1	−1

Horizontal

−1	−1	2
−1	2	−1
2	−1	−1

$+45^{\circ}$

−1	2	−1
−1	2	−1
−1	2	−1

Vertical

2	−1	−1
−1	2	−1
−1	−1	2

-45°

- Horizontal mask will result in maximum response when a line passed through the middle row of the mask with a constant background.
- The similar idea is used with other masks.
- note: The preferred direction of each mask is weighted with a larger coefficient (i.e.,2) than other possible directions.

6 Image Analysis –Line Detection

- Apply every mask on the image.
- let R_1, R_2, R_3, R_4 denotes the response of the horizontal, +45 degree, vertical and -45 degree masks, respectively.
- if, at a certain point in the image

$$|R_i| > |R_j|,$$

- for all $j \neq i$, that point is said to be more likely associated with a line in the direction of mask i.
- To detect all lines in an image in the direction defined by a given mask, we simply run the mask through the image and threshold the absolute value of the result.
- Points left are the strongest responses, which, correspond closest to the direction defined by the mask.

6 Image Analysis –Edge Detection

- How can an algorithm extract relevant information from an image to recognize objects?
- Most important information for the interpretation of an image (for both technical and biological systems) is the contour of objects.
- Contours are indicated by abrupt changes in brightness.
- We can use edge detection filters to extract contour information from an image.

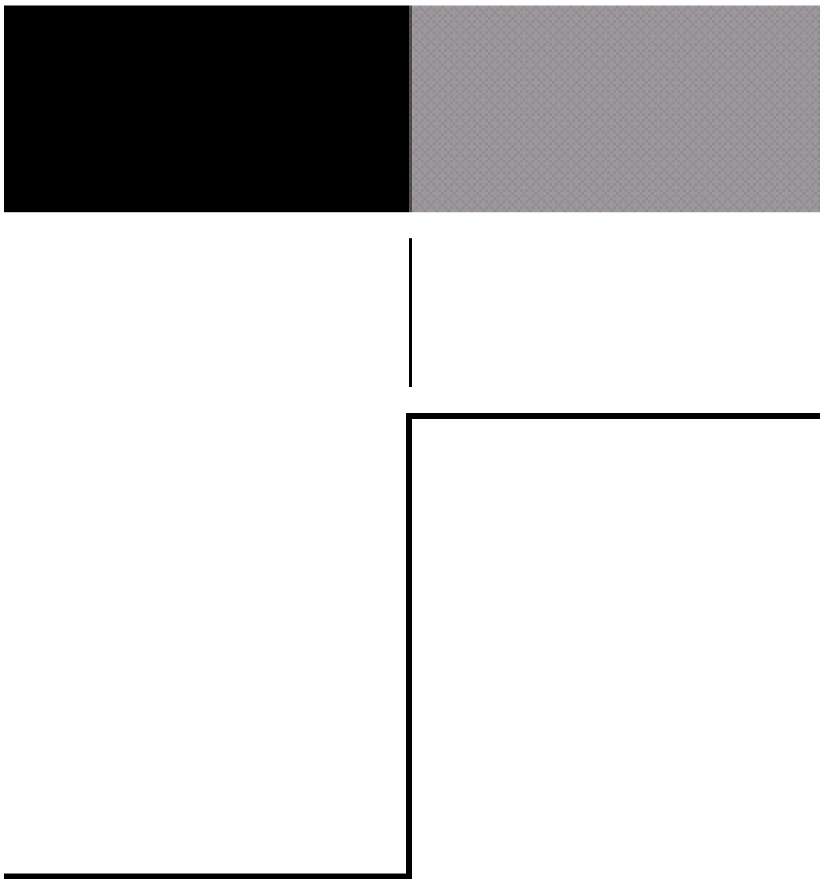
6 Image Analysis –Edge Detection

- Edge detection is the most common approach for detecting meaningful discontinuities in gray level.
- We will discuss approaches of
  - first-order derivative (Gradient operator)
  - second-order derivative (Laplacian operator)

- Intuitively, an edge is a set of connected pixels that lie on the boundary between two regions.
- Changes or discontinuities in image amplitude provide an indication of physical extent of object

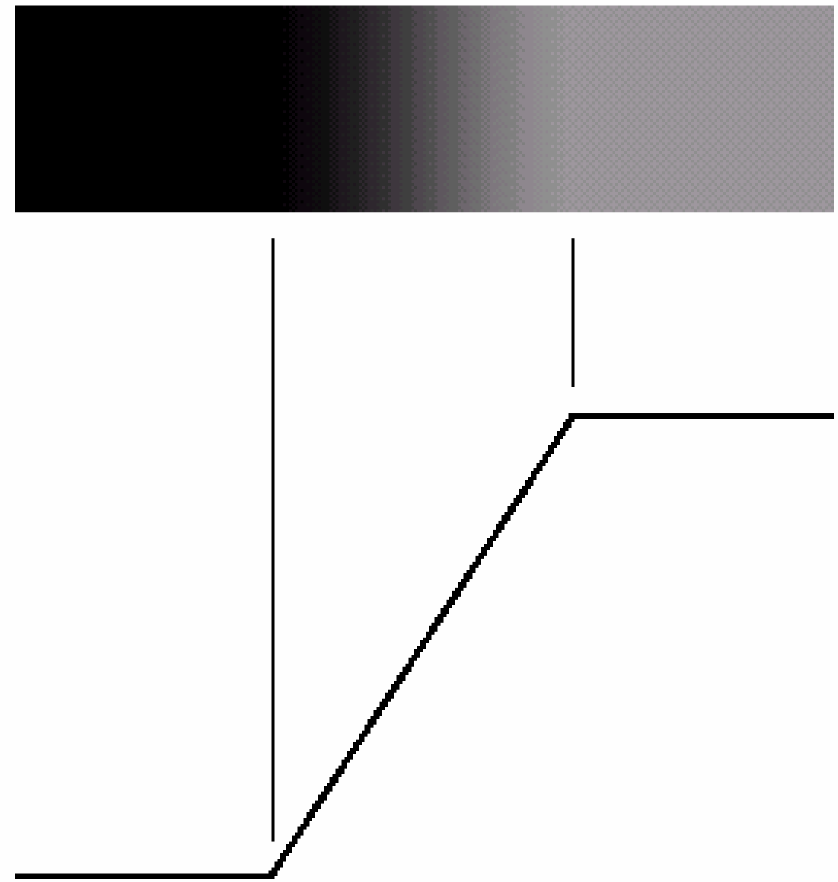
6 Image Analysis –Edge Detection

Model of an ideal digital edge



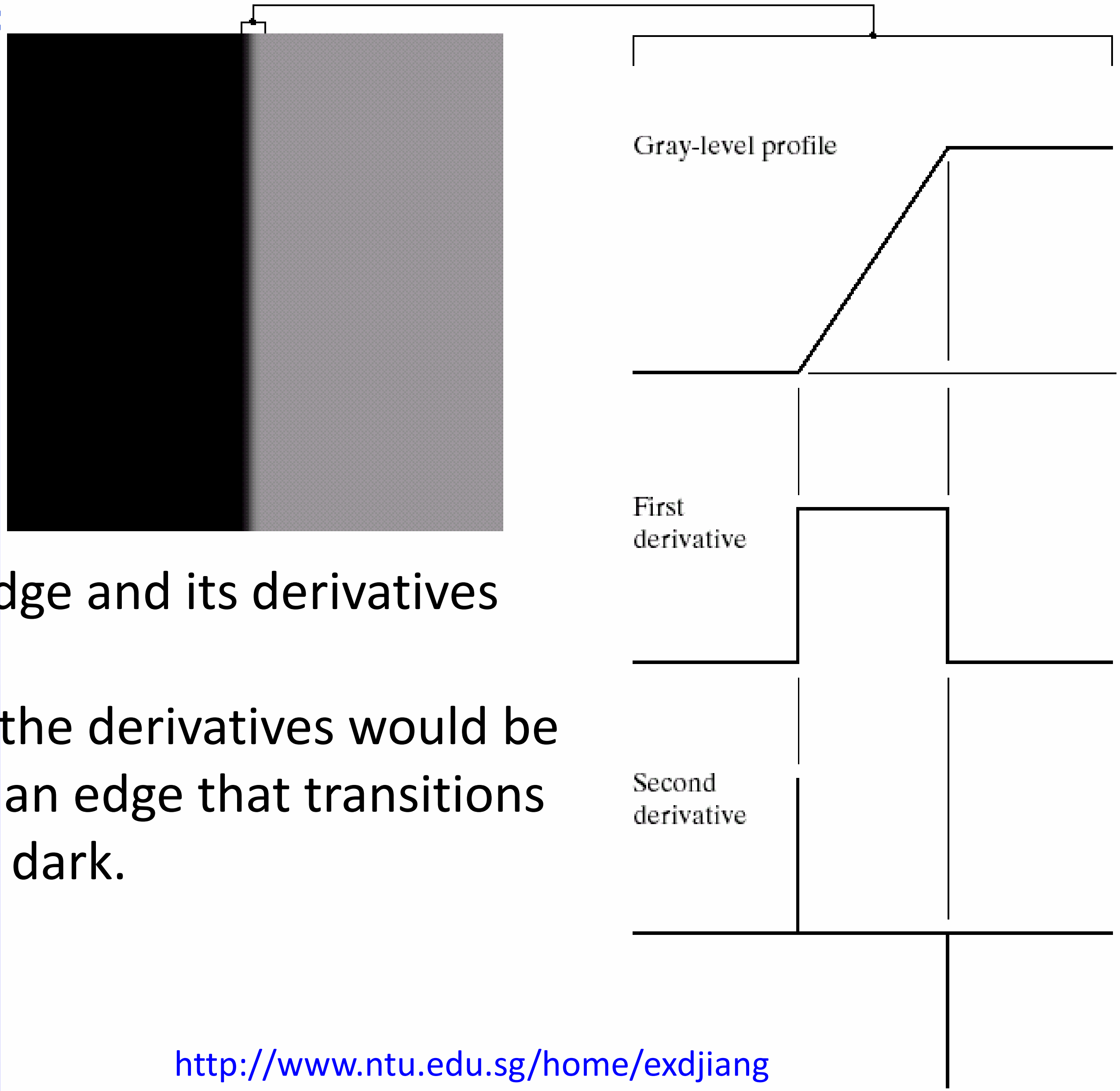
Gray-level profile
of a horizontal line
through the image

Model of a ramp digital edge



Gray-level profile
of a horizontal line
through the image

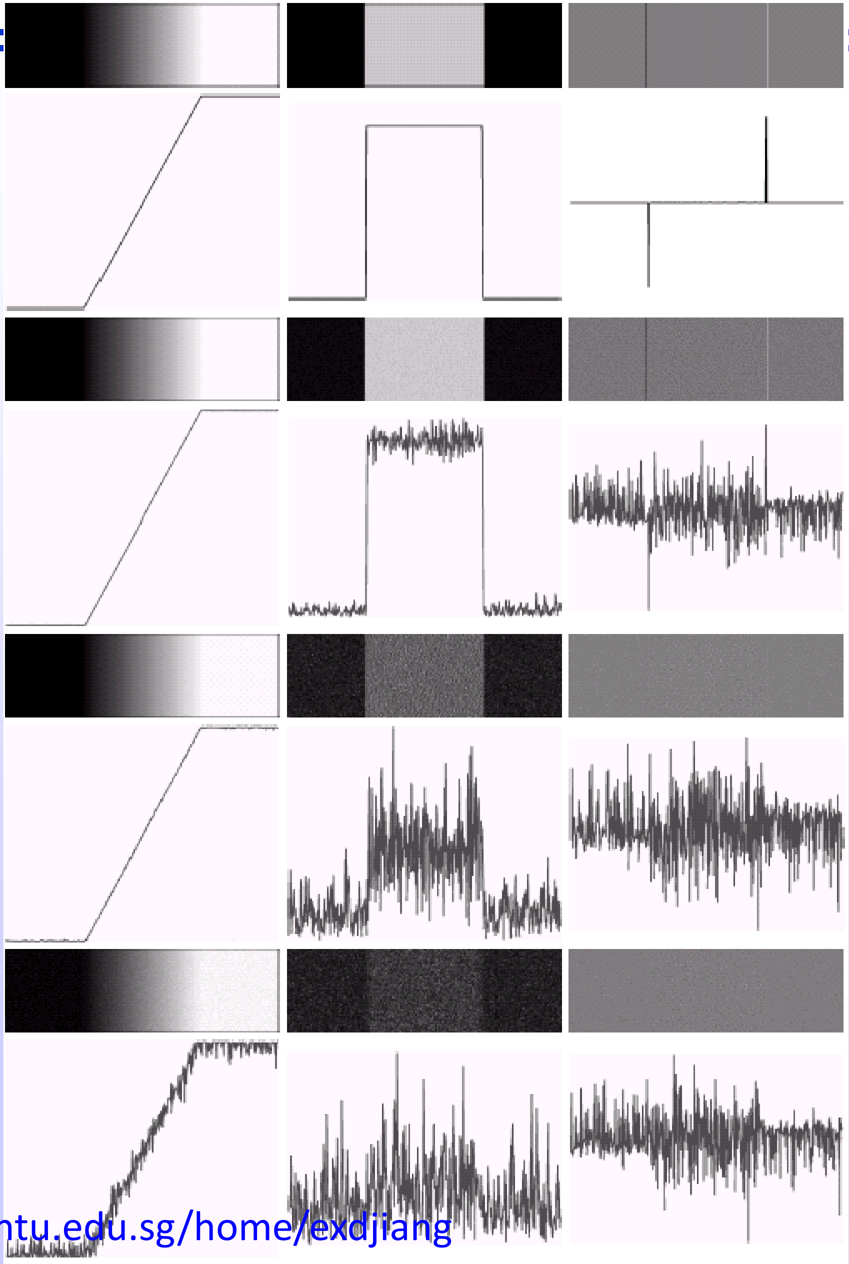
6 Image Analysis –Edge First & Second Derivatives



- Noise free edge and its derivatives
- The signs of the derivatives would be reversed for an edge that transitions from light to dark.

6 Image Analysis –Noisy Edge Derivatives

- First column: images and gray-level profiles of a ramp edge corrupted by random Gaussian noise of mean 0 and σ = 0.0, 0.1, 1.0 and 10.0, respectively.
- Second column: first-derivative images and gray-level profiles.
- Third column : second-derivative images and gray-level profiles..



6 Image Analysis –Noisy Edge Derivatives

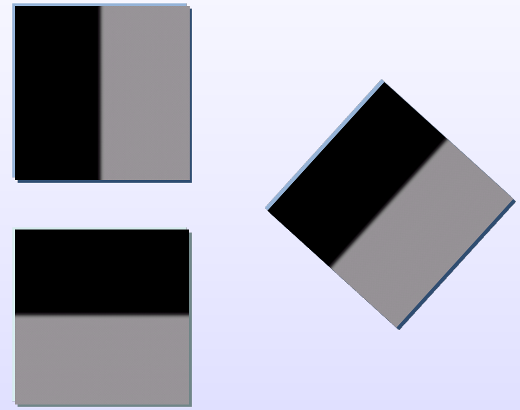
- Fairly little noise can have a significant impact on the two key derivatives used for edge detection in images.
- Image smoothing should be serious consideration prior to the use of derivatives in applications where noise is likely to be present.

- To determine a point as an edge point
  - The transition in grey level associated with the point has to be significantly stronger than the background at that point.
  - Use threshold to determine whether a value is “significant” or not.
  - The point’s two-dimensional first-order derivative must be greater than a specified threshold.

6 Image Analysis –Image First Derivative: Gradient

- The first-order derivative of image called gradient is a two-dimensional vector, which consists of x- and y-differentials.

$$\nabla f(x,y) = \begin{bmatrix} G_x(x,y) \\ G_y(x,y) \end{bmatrix} = \begin{bmatrix} \dfrac{\partial f(x,y)}{\partial x} \\ \dfrac{\partial f(x,y)}{\partial y} \end{bmatrix}$$



- The strength of the differentials is proportional to the degree of discontinuity of the image.
- Thus, image differentiation
  enhances edges and other discontinuities (noise)
  deemphasizes area with slowly varying gray-level values.

6 Image Analysis –Image First Derivative: Gradient

- A image gradient vector has magnitude

$$|\nabla f(x,y)| = [G_x^2(x,y) + G_y^2(x,y)]^{1/2}$$

And direction

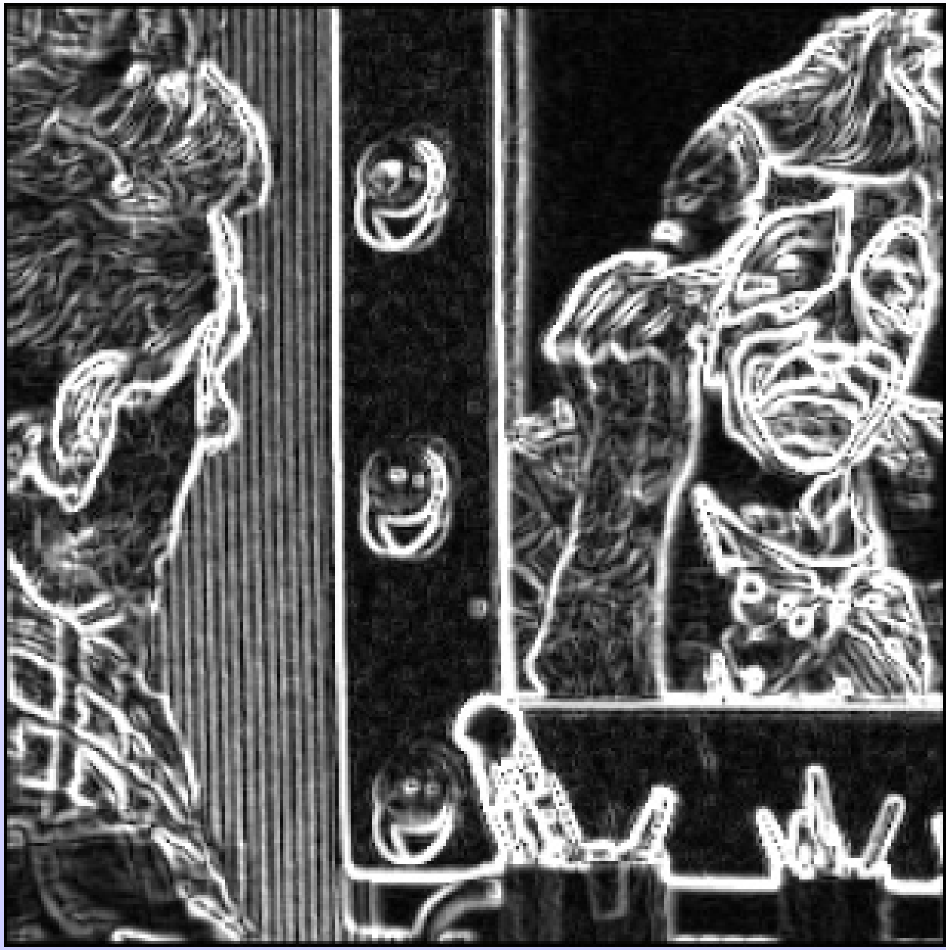
$$\varphi(x,y) = \tan^{-1}\left(\frac{G_y(x,y)}{G_x(x,y)}\right)$$

- Gradient vector points in the direction of maximum rate of change of f at coordinate (x, y).

- The directions (orientation) of an edge at (x, y) is perpendicular to the direction of the gradient vector at that point. In this orientation, the rate of change of f at coordinate (x, y) is minimum.

6 Image Analysis –Gradient Examples

Example

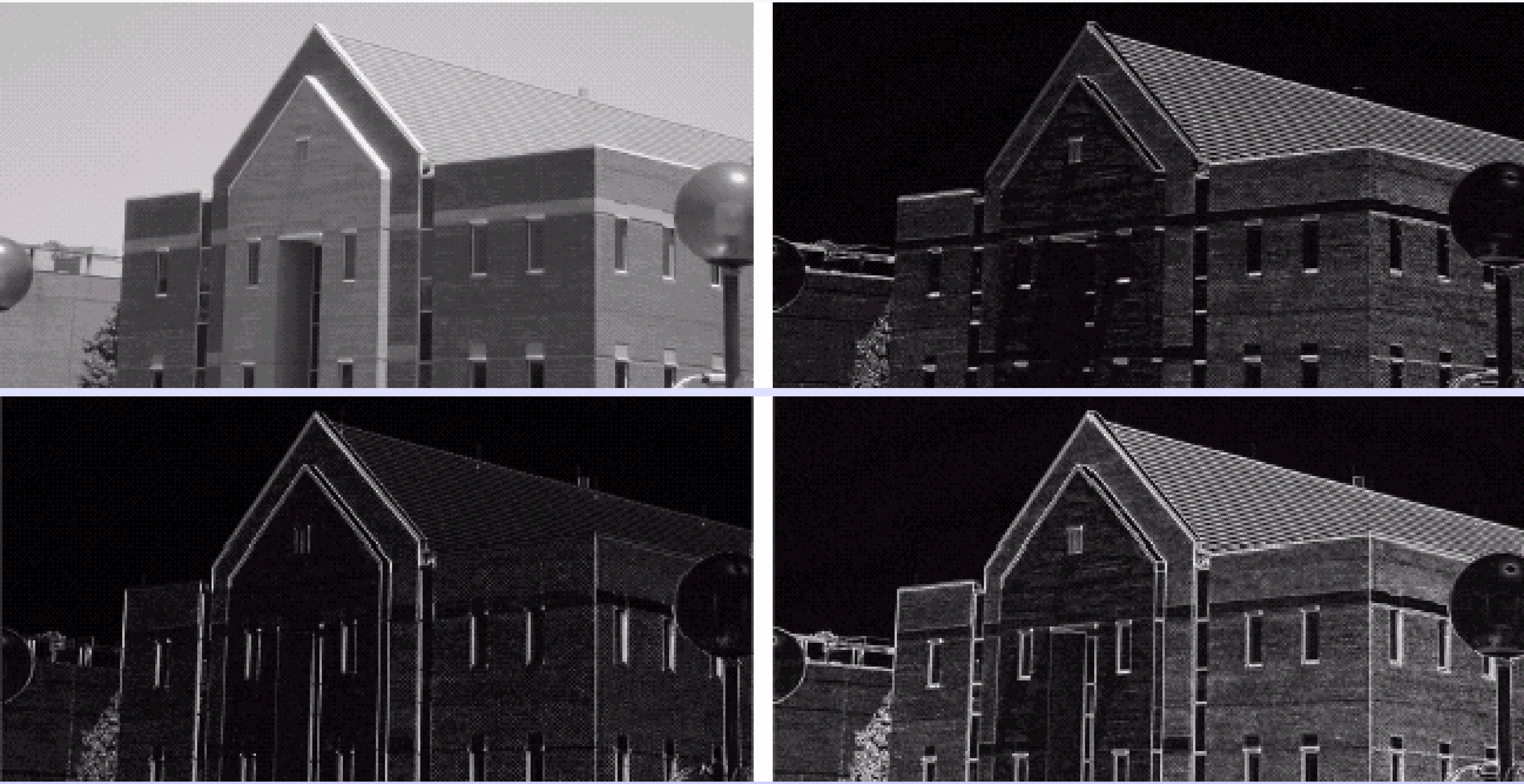


6 Image Analysis –Gradient Examples

Example

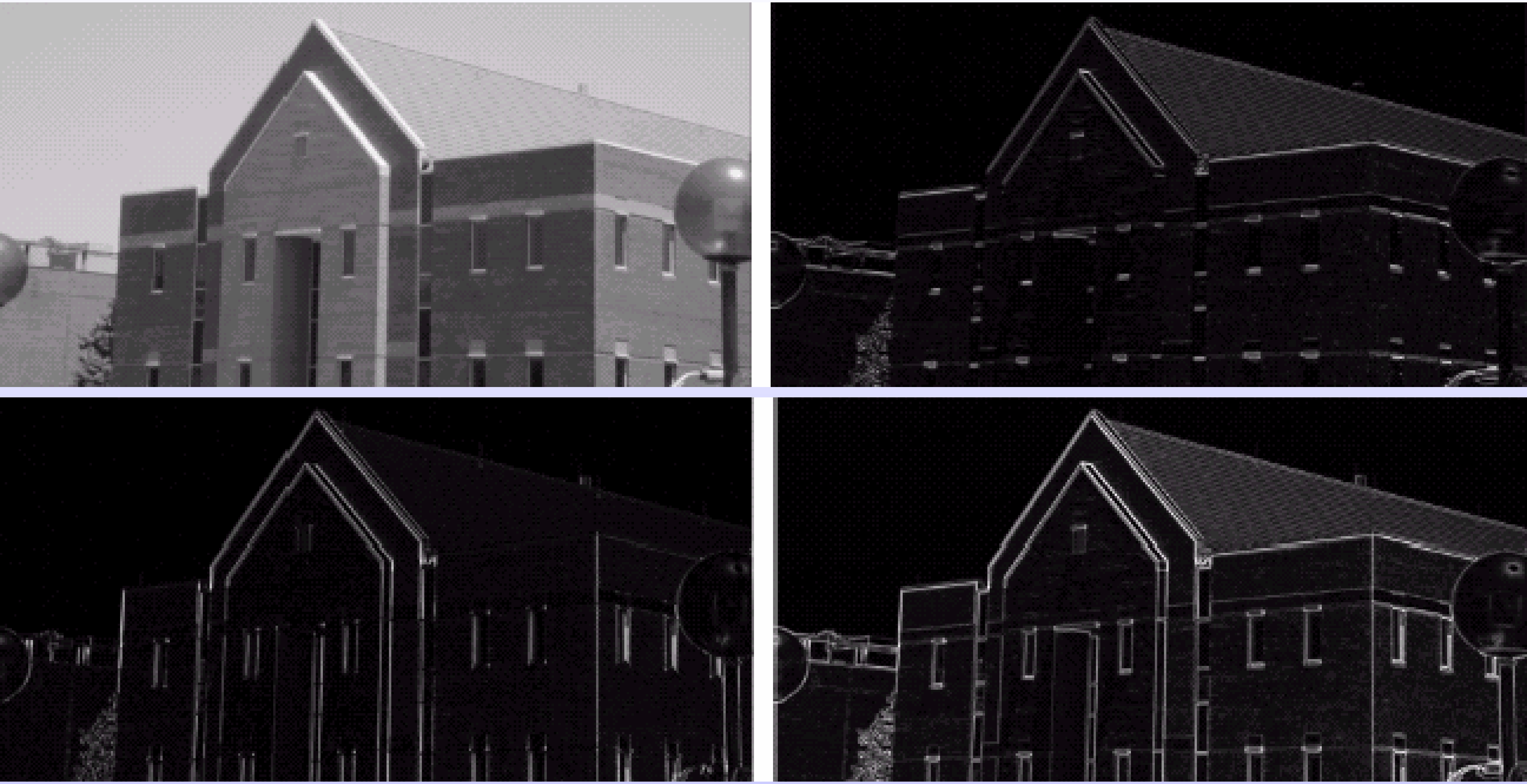
6 Image Analysis –Gradient Examples

original, x- and y-differentials and gradient images with 3×3 masks



6 Image Analysis –Gradient Examples

original, x- and y-differentials and gradient images with 5×5 masks



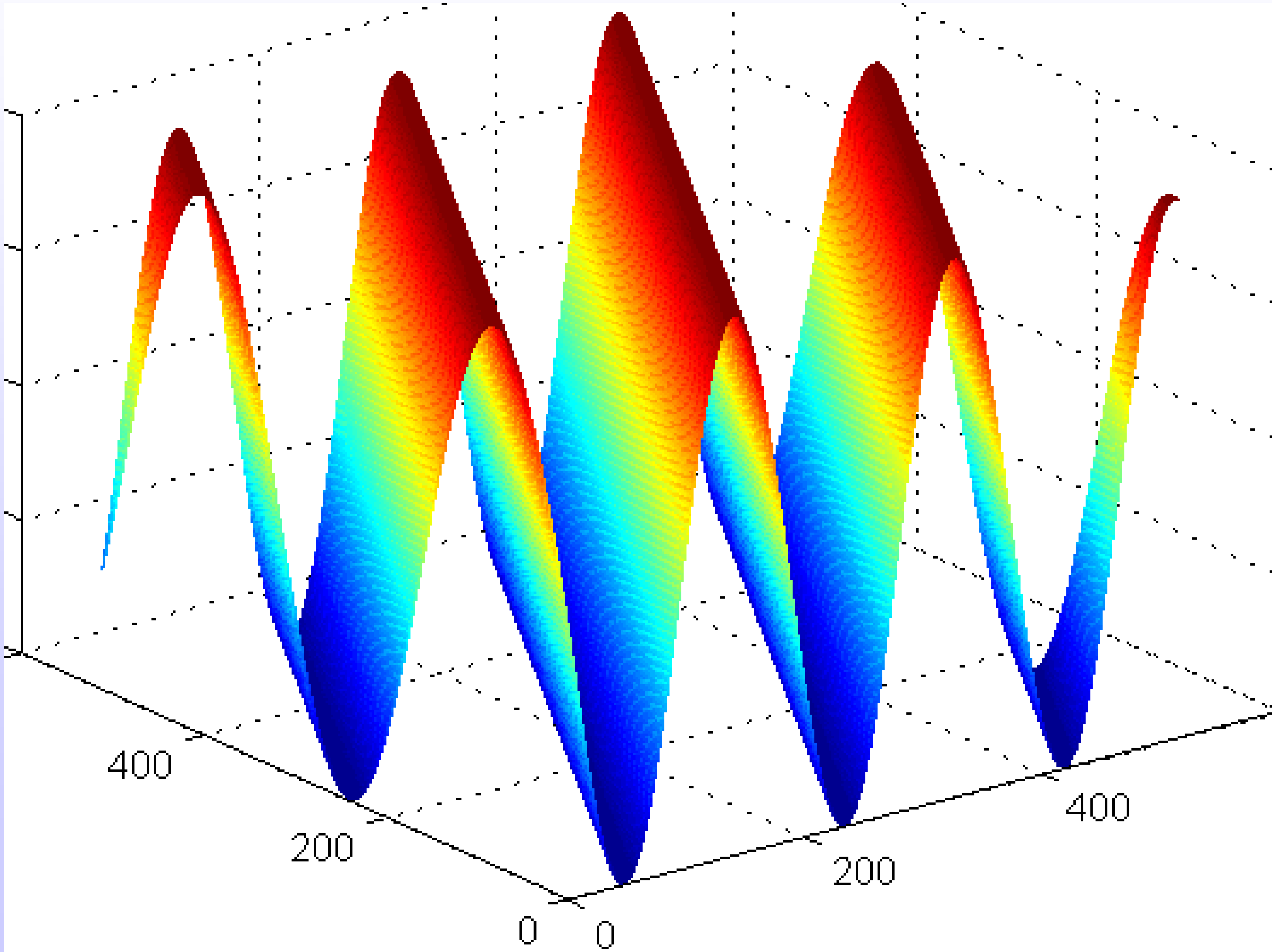
6 Image Analysis –Image First Derivative: Gradient

$$f(x, y) = \cos(ax + by)$$

$$= \cos(t)$$

$$ax + by = t$$

Understand edge, line orientation and frequency



6 Image Analysis –Image First Derivative: Gradient

Understand edge, line orientation and frequency

$$f(x,y) = h(ax+by) = \cos(ax+by)$$

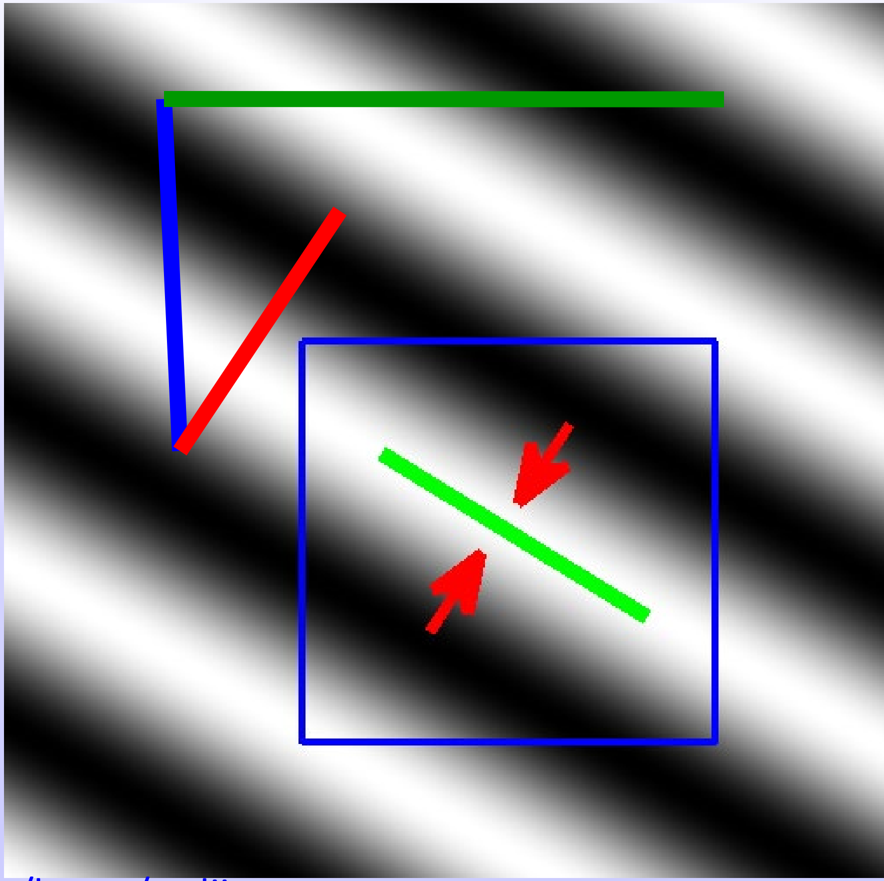
$$= \cos(\omega_1 x + \omega_2 y)$$

$$= \cos(2\pi u_o x + 2\pi v_o y) =$$

$$\cos[2\pi f \sin(\theta) x + 2\pi f \cos(\theta) y]$$

$$= \cos(\frac{2\pi}{T_x} x + \frac{2\pi}{T_y} y) =$$

$$\cos[\frac{2\pi}{T}\sin(\theta)x + \frac{2\pi}{T}\cos(\theta)y]$$



6 Image Analysis –Image First Derivative: Gradient

An image is modeled locally by

$$f(x, y) = h(ax + by)$$

$ax + by = t$ is the line represent the orientation of $f(x, y)$

$$y = -\frac{a}{b}x + t = \tan\theta x + t$$

$$\nabla f(x, y) = \begin{bmatrix} G_x(x, y) \\ G_y(x, y) \end{bmatrix} = \begin{bmatrix} \dfrac{\partial f(x, y)}{\partial x} \\ \dfrac{\partial f(x, y)}{\partial y} \end{bmatrix} = \begin{bmatrix} ah'(t) \\ bh'(t) \end{bmatrix}$$

$$\varphi(x, y) = \tan^{-1}\left(\frac{G_y(x, y)}{G_x(x, y)}\right) = \tan^{-1}\frac{b}{a} = -\tan^{-1}\frac{a}{b} \pm \frac{\pi}{2} = \theta \pm \frac{\pi}{2}$$

6 Image Analysis –Image First Derivative: Gradient

- A simple approximation of the x- and y-differentials for digital image.

$$G_x(x,y) = f(x+1,y) - f(x-1,y)$$
$$G_y(x,y) = f(x,y+1) - f(x,y-1)$$

- This is equivalent to run the two simple 3X3 masks through the image:

$$\begin{pmatrix} 0 & 0 & 0 \\ -1 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}, \quad \begin{pmatrix} 0 & -1 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$$

$$h_x(x,y), \qquad h_y(x,y)$$

$$G_x(x,y) = h_x(x,y) * f(x,y)$$
$$G_y(x,y) = h_y(x,y) * f(x,y)$$

- However, it is very sensitive to noise.

6 Image Analysis –Image First Derivative: Gradient

- Therefore, certain smoothing is desirable prior to application of differentiation.

$$\nabla[h(x,y) * f(x,y)] = [\nabla h(x,y)] * f(x,y)$$

- Due to linearity of differentiation, differentiate the image convolved (smoothed) with h is same as convolving an image with $\nabla h(x,y)$. So we have gradient operator (Mask) $\nabla h(x,y)$.

- Different design of the smooth filter $h(x,y)$ leads to various different gradient operators (Masks) $\nabla h(x,y)$.

6 Image Analysis –Image First Derivative: Gradient

- Some gradient masks

$$G_y = (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)$$

$$G_x = (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)$$

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

−1	−1	−1
0	0	0
1	1	1

−1	0	1
−1	0	1
−1	0	1

Prewitt

−1	−2	−1
0	0	0
1	2	1

−1	0	1
−2	0	2
−1	0	1

Sobel

0	1	1
−1	0	1
−1	−1	0

−1	−1	0
−1	0	1
0	1	1

Prewitt

0	1	2
−1	0	1
−2	−1	0

−2	−1	0
−1	0	1
0	1	2

Sobel

6 Image Analysis –Image First Derivative: Gradient

- Gradient operator (mask) should have noise suppression characteristics, which is important for robust edge detection.
- However, gradient operator may cause biased gradient direction due to the discrete image and the smoothing filter. Accurate gradient direction is desirable for some applications.
- This issue is analyzed and solution was proposed in the publication:

  X.D. Jiang, "Extracting Image Orientation Feature by Using Integration Operator," *Pattern Recognition*, Vol. 40, No. 2, pp. 705-717, February 2007.

6 Image Analysis –Image First Derivative: Gradient

A digital image is modeled locally by

$$f(x,y)=\cos(ax+by)$$

$$\begin{aligned}\hat{\nabla}h(x,y) &= f(x+1,y)-f(x-1,y) \\ &+j[f(x,y+1)-f(x,y-1)] \\ &=\cos(a(x+1)+by)-\cos(a(x-1)+by) \\ &+j[\cos(ax+b(y+1))-\cos(ax+b(y-1))]\end{aligned}$$

$$\varphi(x,y)=\arctan\frac{\mathrm{Im}\{\hat{\nabla}f(x,y)\}}{\mathrm{Re}\{\hat{\nabla}f(x,y)\}}\neq\arctan\frac{b}{a}$$

6 Image Analysis –Image First Derivative: Gradient

Prewitt, isotropic, Sobel, circular $\begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}, \begin{bmatrix} 1 & 0 & -1 \\ \sqrt{2} & 0 & -\sqrt{2} \\ 1 & 0 & -1 \end{bmatrix}, \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}, \begin{bmatrix} .464 & 0 & -.464 \\ .959 & 0 & -.959 \\ .464 & 0 & -.464 \end{bmatrix}$

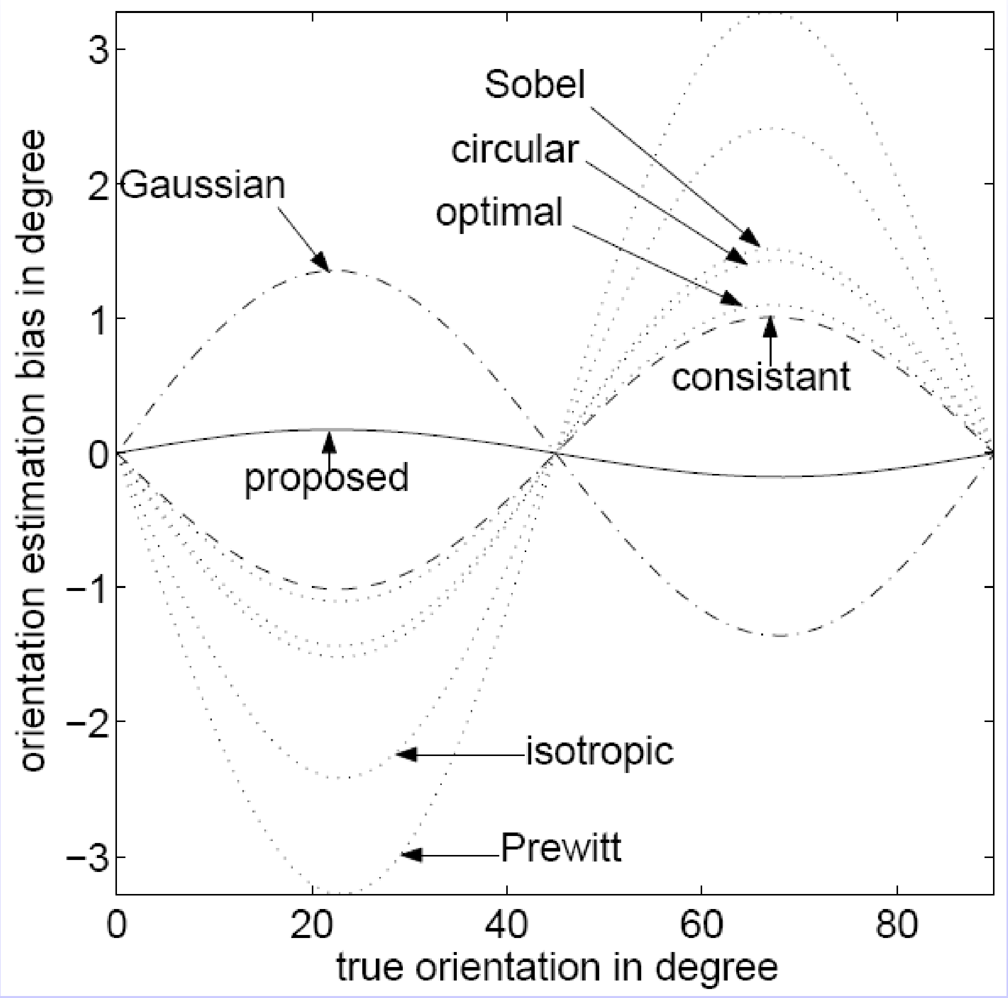
consistent operator (PAMI 00)

optimal operator (IEEE TIP 04)

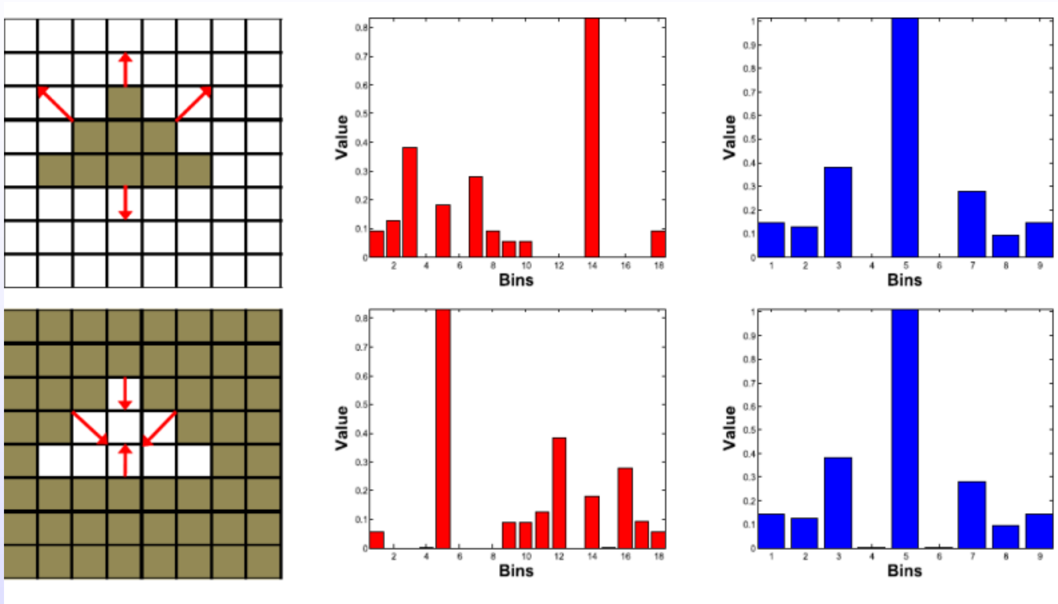
Proposed operator

$$\begin{bmatrix} 1 & 0 & -1 \\ 4 & 0 & -4 \\ 1 & 0 & -1 \end{bmatrix}$$

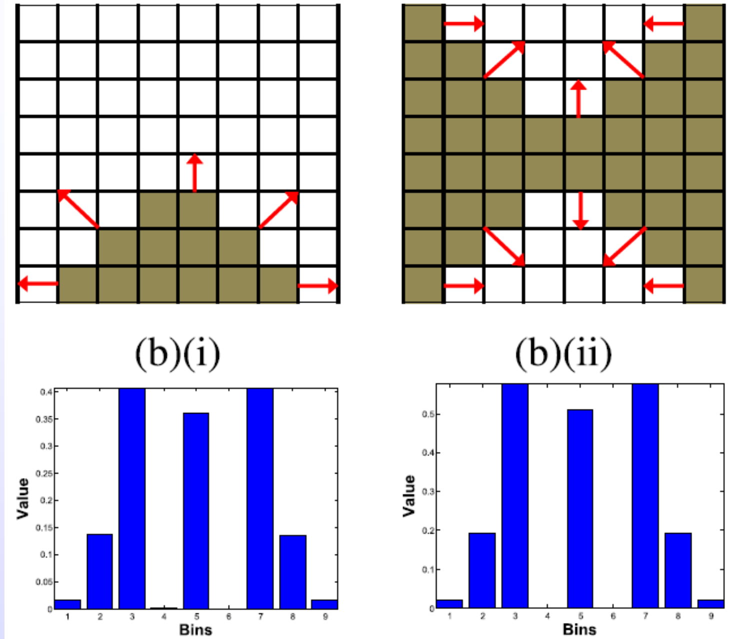
X. Jiang, Extracting Image Orientation Feature by Using Integration Operator, ***Pattern Recognition***, vol. 40, no. 2, pp. 705-717, Feb. 2007.



6 Image Analysis –Histogram of Gradient or Unsigned Gradient?



(a) HG (b) HOG



(b)(i) (b)(ii)

A. Satpathy, X. Jiang and H. Eng, "Human Detection by Quadratic Classification on Subspace of Extended Histogram of Gradients," ***IEEE Trans. Image Processing***, vol. 23, no. 1, pp. 287-297, Jan, 2014.

6 Image Analysis –Second-Order Derivative

- Second-order derivative is also called Laplacian operator defined by

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

- Approximation in discrete domain:

$$\frac{\partial^2 f}{\partial x^2} = [f(x+1,y) - f(x,y)] - [f(x,y) - f(x-1,y)]$$

$$= f(x+1,y) + f(x-1,y) - 2f(x,y)$$

- Similarly: $$\frac{\partial^2 f}{\partial y^2} = f(x,y+1) + f(x,y-1) - 2f(x,y)$$

6 Image Analysis –Second-Order Derivative

- This yield

$$\nabla^2 f = f(x+1,y) + f(x-1,y) + f(x,y+1) + f(x,y-1) - 4f(x,y)$$

- This can be implemented by a four-neighbor Laplacian mask or an eight-neighbor Laplacian mask

0	−1	0
−1	4	−1
0	−1	0

−1	−1	−1
−1	8	−1
−1	−1	−1

$$h_L(x,y)$$

$$\nabla^2 f(x,y) = h_L(x,y) * f(x,y)$$

6 Image Analysis –Laplacian of Gaussian (LoG)

- Laplacian is sensitive to noise
- Therefore, certain smoothing is desirable prior to application of Laplacian
- Solution: Employ Gaussian-shaped smoothing

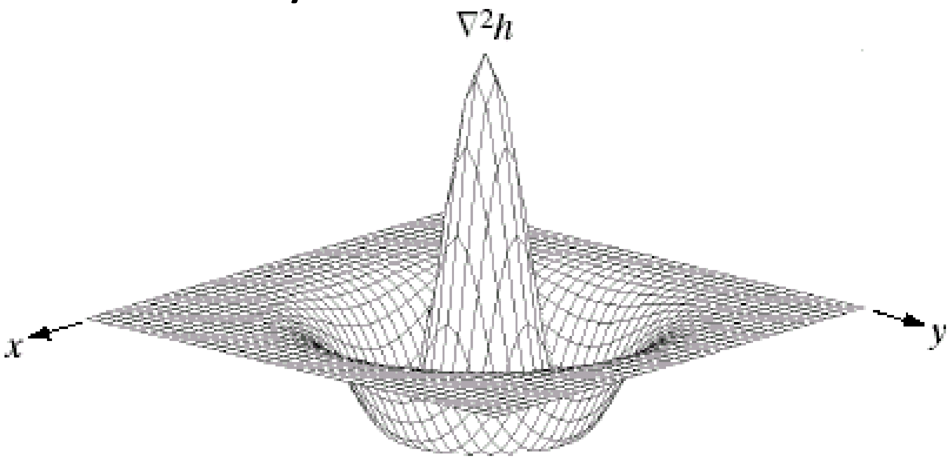
$$h(x, y) = -e^{-\frac{x^2+y^2}{2\sigma^2}} = -e^{-\frac{r^2}{2\sigma^2}}$$

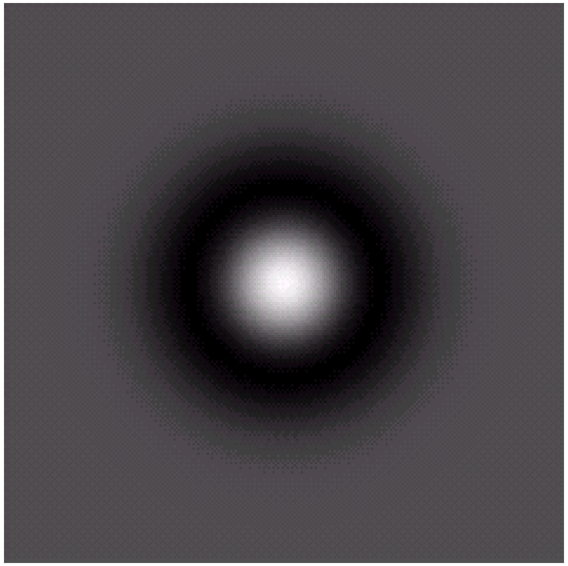
- Due to linearity of second derivative, taking the Laplacian of the image convolved (smoothed) with h is same as convolving an image with $\nabla^2 h$.

$$\nabla^2 (h * f) = (\nabla^2 h) * f$$

6 Image Analysis –Laplacian of Gaussian (LoG)

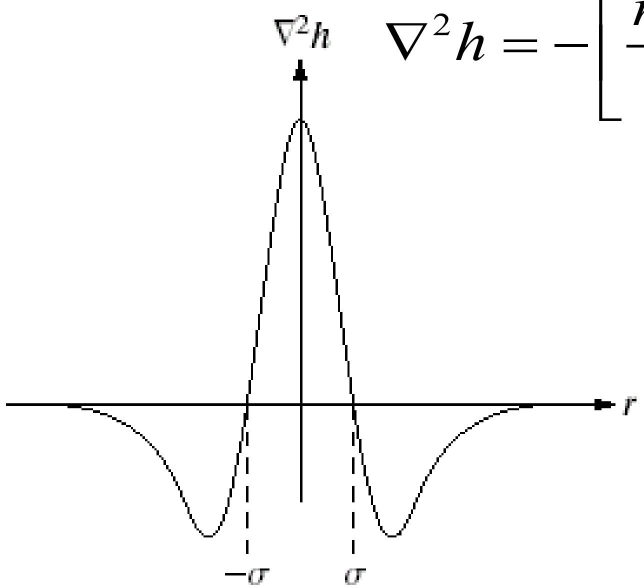
- It is easy to have





Mexican hat

$$\nabla^2 h = -\left[\frac{r^2 - \sigma^2}{\sigma^4}\right] e^{-\frac{r^2}{2\sigma^2}}$$



0	0	−1	0	0
0	−1	−2	−1	0
−1	−2	16	−2	−1
0	−1	−2	−1	0
0	0	−1	0	0

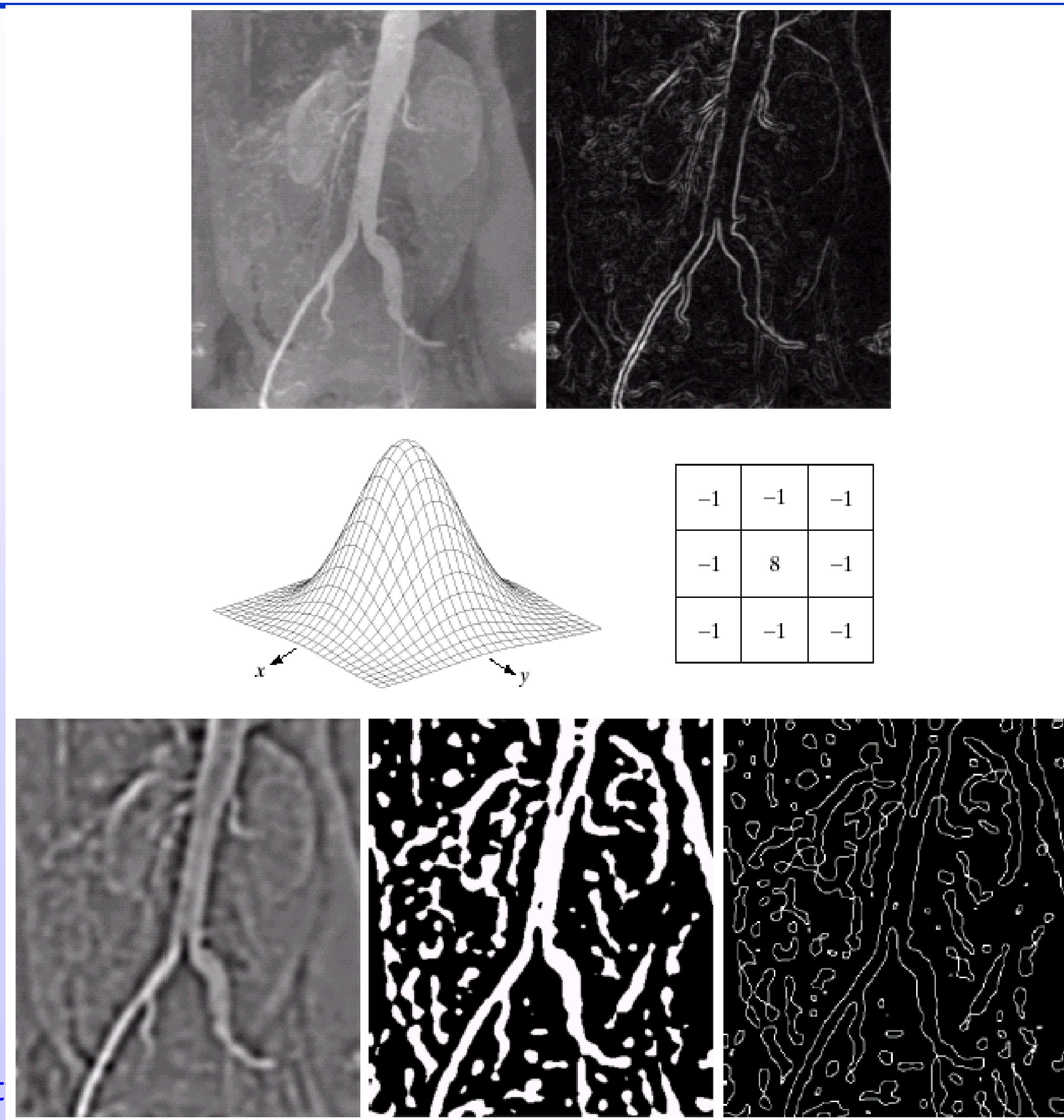
a b
c d

FIGURE 10.14 Laplacian of a Gaussian (LoG). (a) 3-D plot. (b) Image (black is negative, gray is the zero plane, and white is positive). (c) Cross section showing zero crossings. (d) 5 × 5 mask approximation to the shape of (a).

the coefficient must be sum to zero

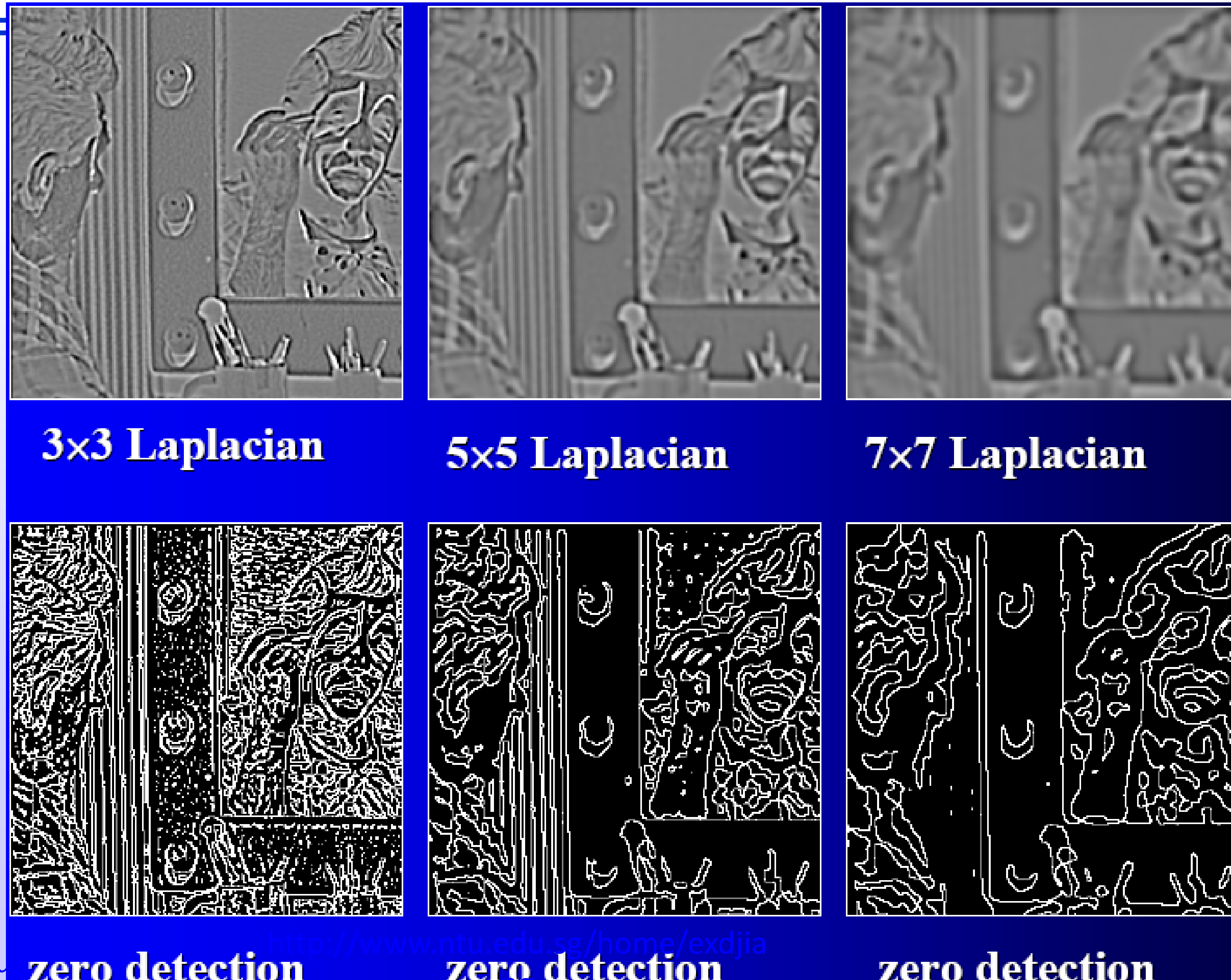
6 Image Analysis –Example of LoG

a) original image
b) Sobel gradient
c) spatial Gaussian
smoothing function
d) Laplacian mask
e) LoG
f) threshold LoG
g) zero crossing



6 Image Analysis –Example of LoG

Example



7 Image Analysis –Edge Linking & Boundary Detection

- Edge detection algorithm are followed by linking procedures to assemble edge pixels into meaningful edges.

- Basic approaches

  Local Processing

  Global Processing via the Hough Transform

7 Image Analysis –Local Processing

- Analyze the characteristics of pixels in a small neighborhood (say, 3x3, 5x5) about every edge pixels (x,y) in an image.

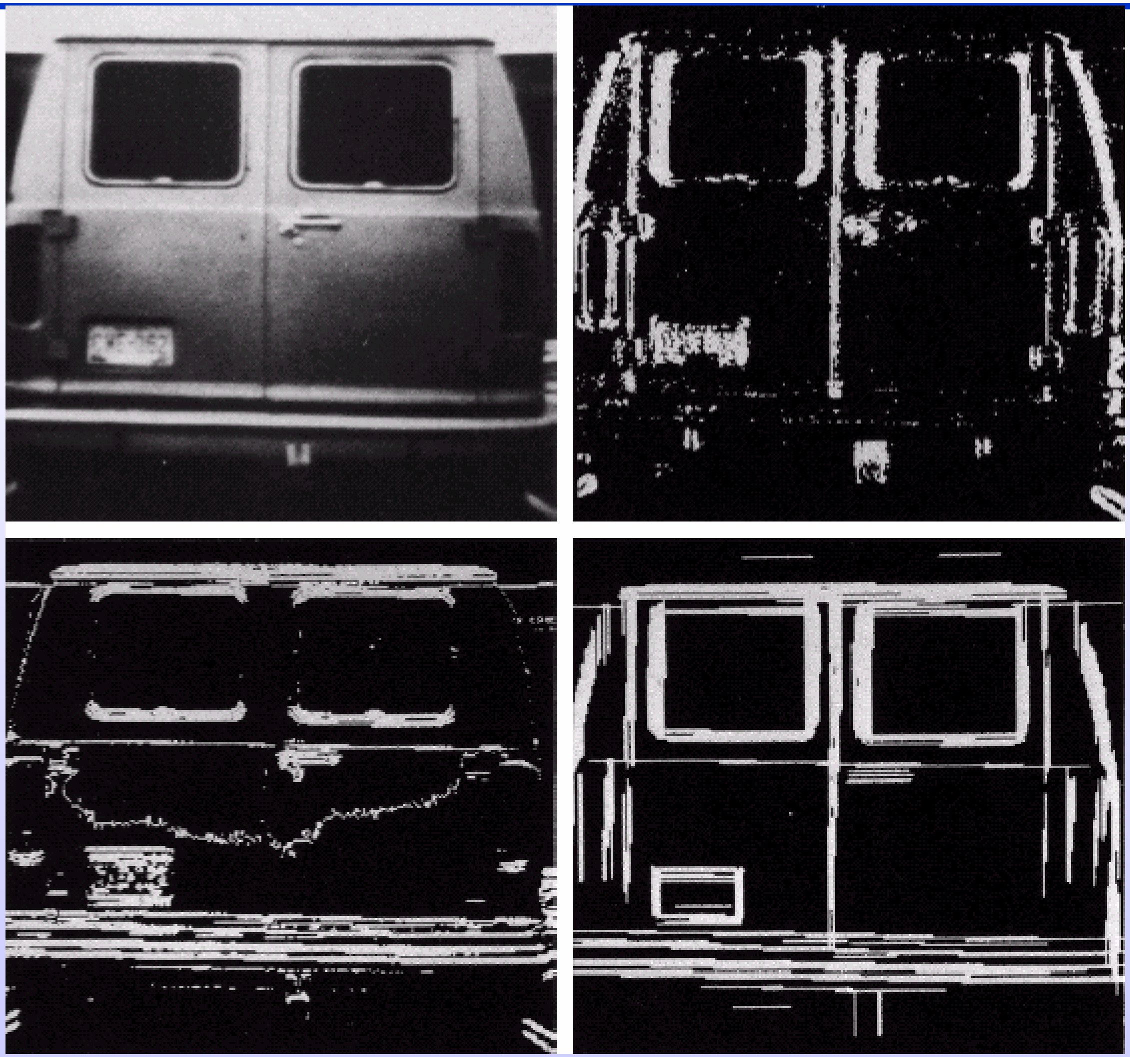
- All points that are similar according to a set of predefined criteria are linked, forming an edge of pixels that share those criteria.

7 Image Analysis –Local Processing

1. The strength of the gradient vector
   An edge pixel with coordinates (x_0,y_0) in a predefined neighborhood of (x,y) is similar in magnitude to the pixel at (x,y) if $|\nabla f(x,y) - \nabla f(x_0,y_0)| \leq E$
2. The direction of the gradient vector
   An edge pixel with coordinates (x_0,y_0) in a predefined neighborhood of (x,y) is similar in angle to the pixel at (x,y) if $|\theta(x,y) - \theta(x_0,y_0)| < A$

A point in the predefined neighborhood of (x_0,y_0) is linked to the pixel at (x,y) if both magnitude and direction criteria are satisfied.

7 Image Analysis –Local Processing



Example

use horizontal and vertical Sobel operators

eliminate isolated short segments

link conditions:
gradient value > 25
gradient direction difference < 15°

7 Image Analysis –Hough Transform

- Hough transform is a technique that can be used to detect (link) regular curves such as lines, circles, and ellipses in an image.

  Line segment in spatial space: $y = ax + b$

  If the line passes through a point (pixel) (x_i, y_i), we obtain:

$$y_i = ax_i + b$$

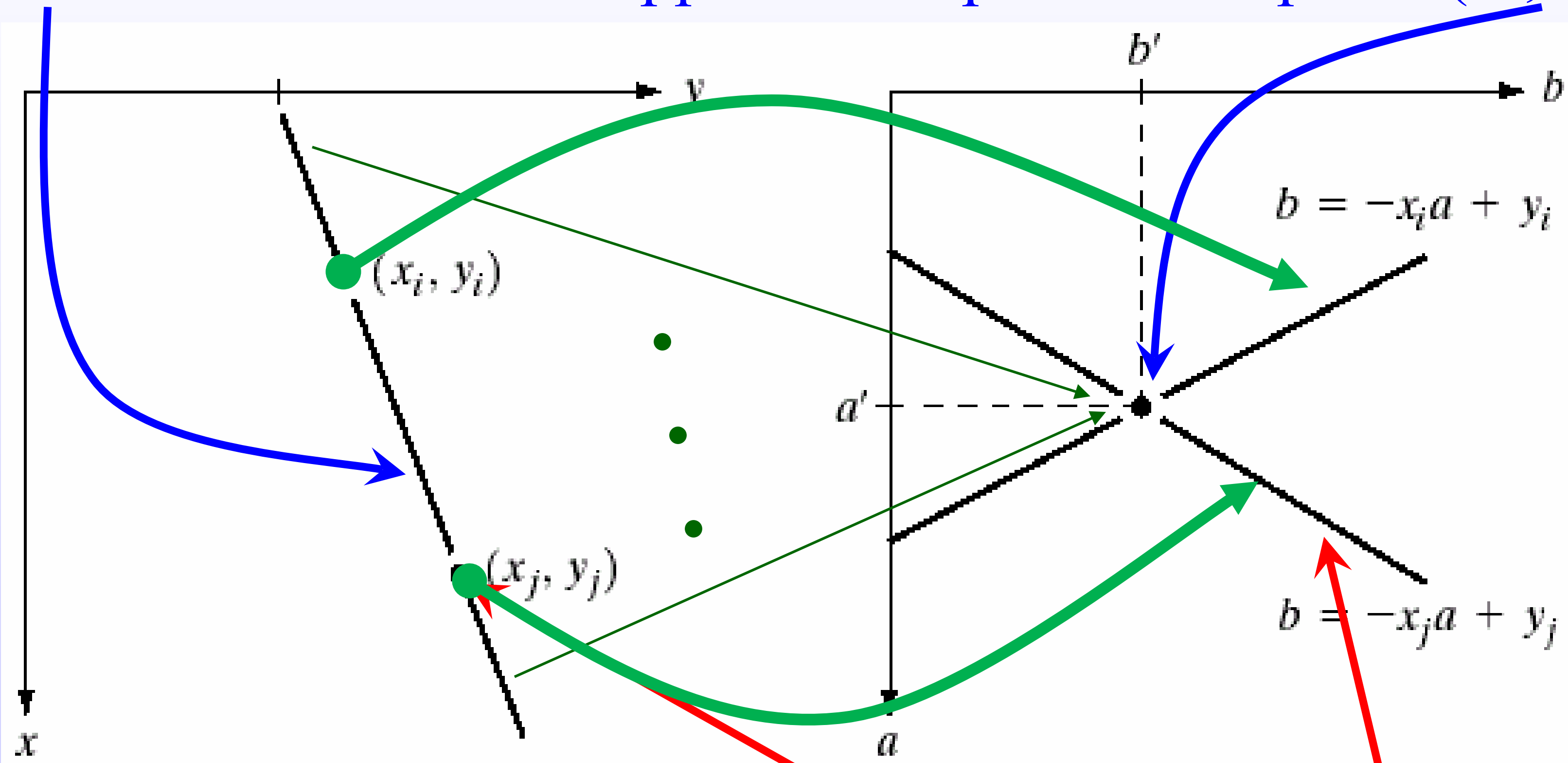
- Rewrite it in ab-plane or parametric space:

$$y_i = ax_i + b$$

$$b = y_i - ax_i$$

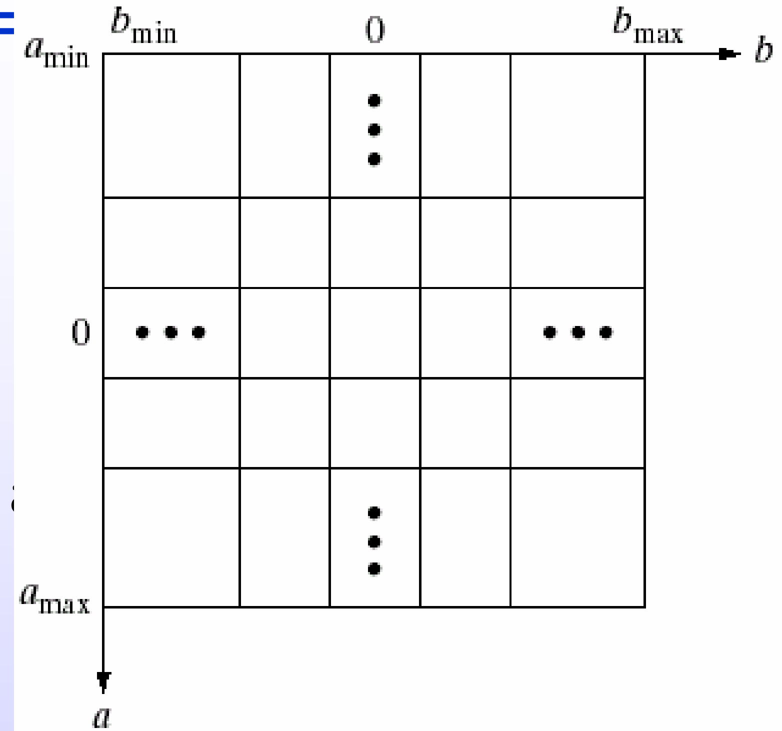
7 Image Analysis –Hough Transform

- A line in *xy*-plane is a set of points (x, y) that satisfy equation: $y = a'x + b'$ which is mapped into a point in ab-plane (a', b')



- A point in *xy*-plane may have many lines go through it, which is mapped into a line in *ab*-plane. $(x_j, y_j) \Leftrightarrow b = y_j - ax_j$

7 Image Analysis –Hough Transform



- All points (x_i, y_i) on a same line in the image must fall into a same point (a_i, b_i) in the parametric space.

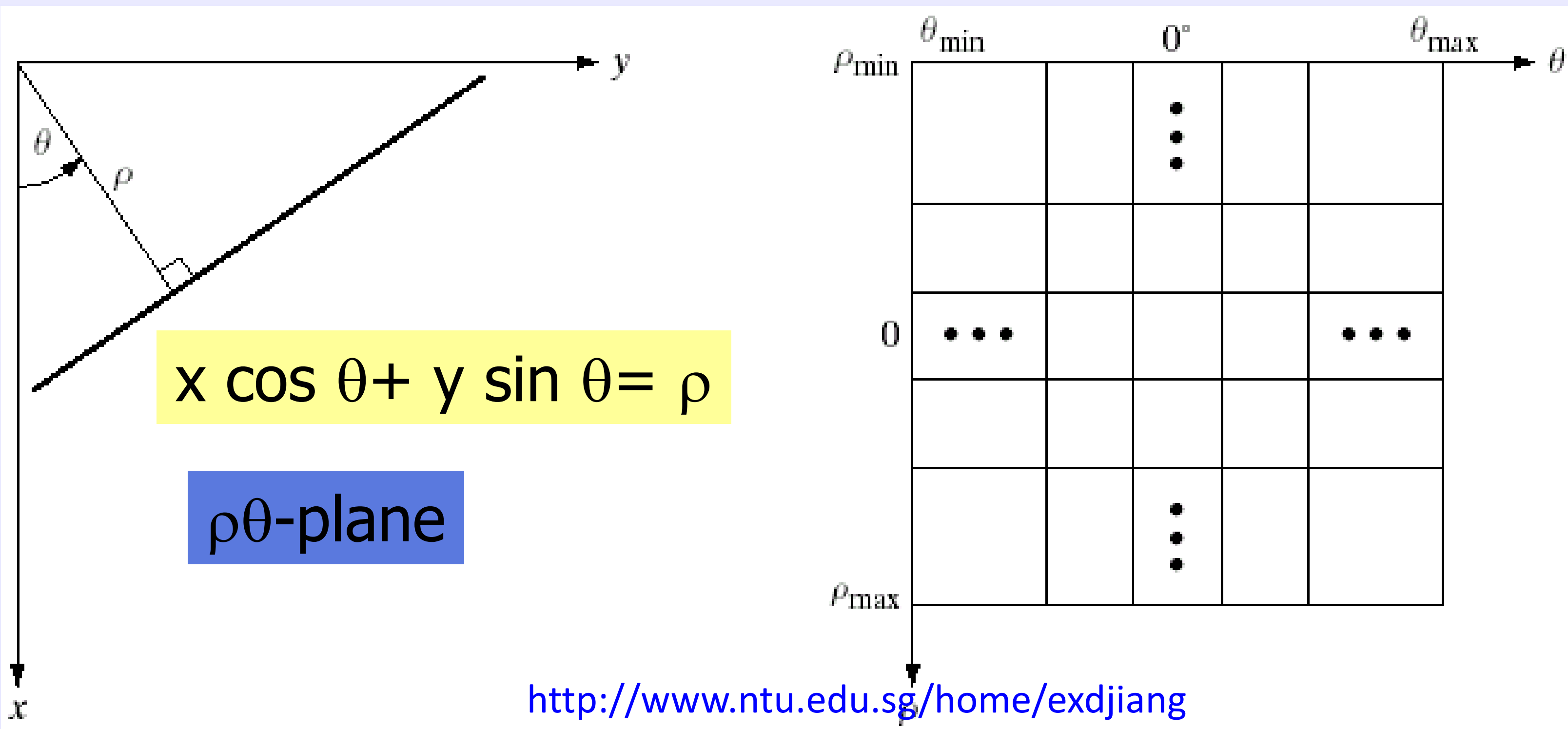
  - Hough transform:
    1. Ddivision of parameter space into a cells (a, b).
    2. All cells are initialized to zero, $$A(a,b)=0$$
    3. For each detected point (x_i, y_i) in the image:

$$A(a,b)+1 \Rightarrow A(a,b) \text{ for all } a \text{ and } b \text{ satisfying } b = y_i - ax_i$$

- At the end of the procedure, value $A(a, b)$ corresponds to the number of points in image lying on the line $y = ax+b$

7 Image Analysis –Hough Transform

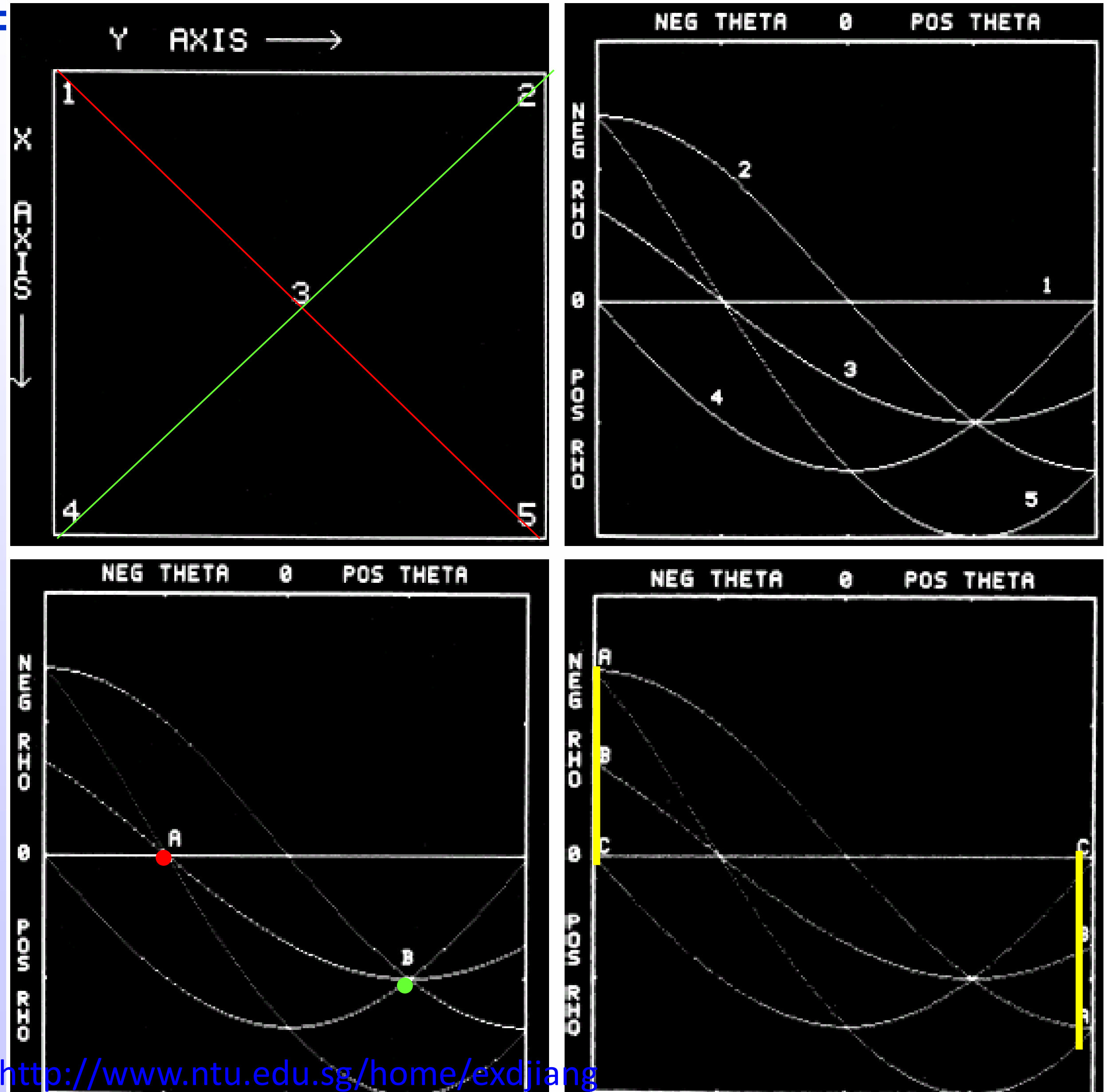
- Problem of using $y=ax+b$ is that a is infinite for a vertical line.
- To avoid the problem, use equation $x\cos\theta + y\sin\theta = \rho$ to represent a line instead.
- Vertical line has $\theta = 90°$ with ρ equals to the positive y-intercept or $\theta = -90°$ with ρ equals to the negative y-intercept.



7 Image Analysis –Hough Transform

- Example

5 points in the image



ρθ-plane

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

7 Image Analysis –Hough Transform

- Generalized Hough transform can be used for any function of the form

$$g(v,c) = 0$$

v is a vector of coordinates, c is a vector of coefficients

- For example a circle is represented by equation:

$$(x-c_1)^2 + (y-c_2)^2 = c_3^2$$

- three parameters (c_1, c_2, c_3)
- cube like cells, accumulators of the form $A(c_1, c_2, c_3)$
- For each point in the image, update the value of $A(c_1, c_2, c_3)$ $\{A(c_1, c_2, c_3) +1 \rightarrow A(c_1, c_2, c_3)\}$ that satisfies the equation $(x-c_1)^2 + (y-c_2)^2 = c_3^2$.

7 Image Analysis –Edge Detection by Hough Transform

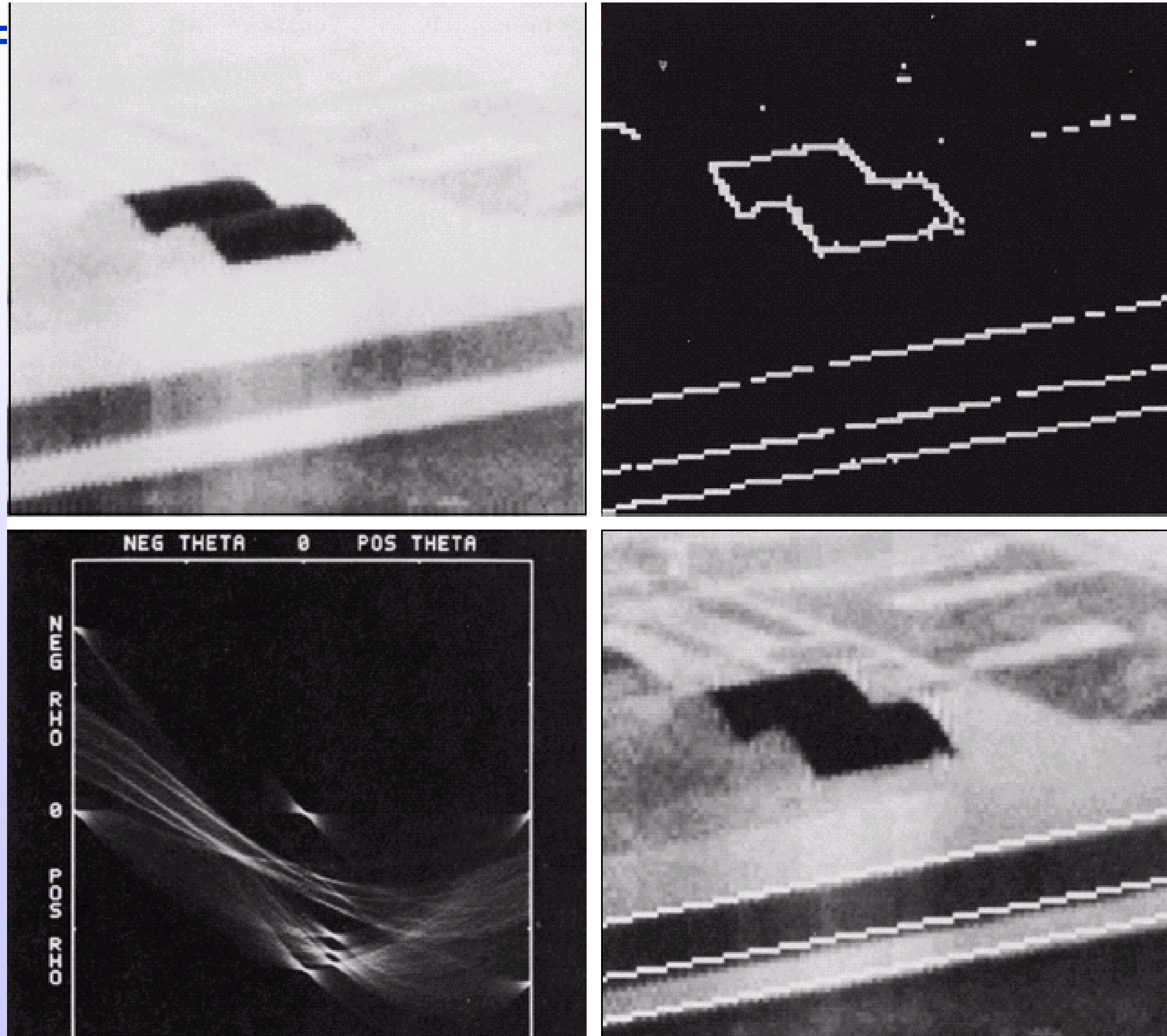
1. Compute the gradient of an image and threshold it to obtain a binary image.
2. Specify subdivisions in the $\rho\theta$-plane.
3. Examine the counts of the accumulator cells for high pixel concentrations.
4. Examine the relation (principally for continuity) between pixels in a chosen cell.
5. A gap at any point is linked if the distance between that point and its closet neighbor below a certain threshold.

7 Image Analysis –Edge Detection by Hough Transform

Example

link criteria:
pixels belonged to a set is linked according to the highest count.

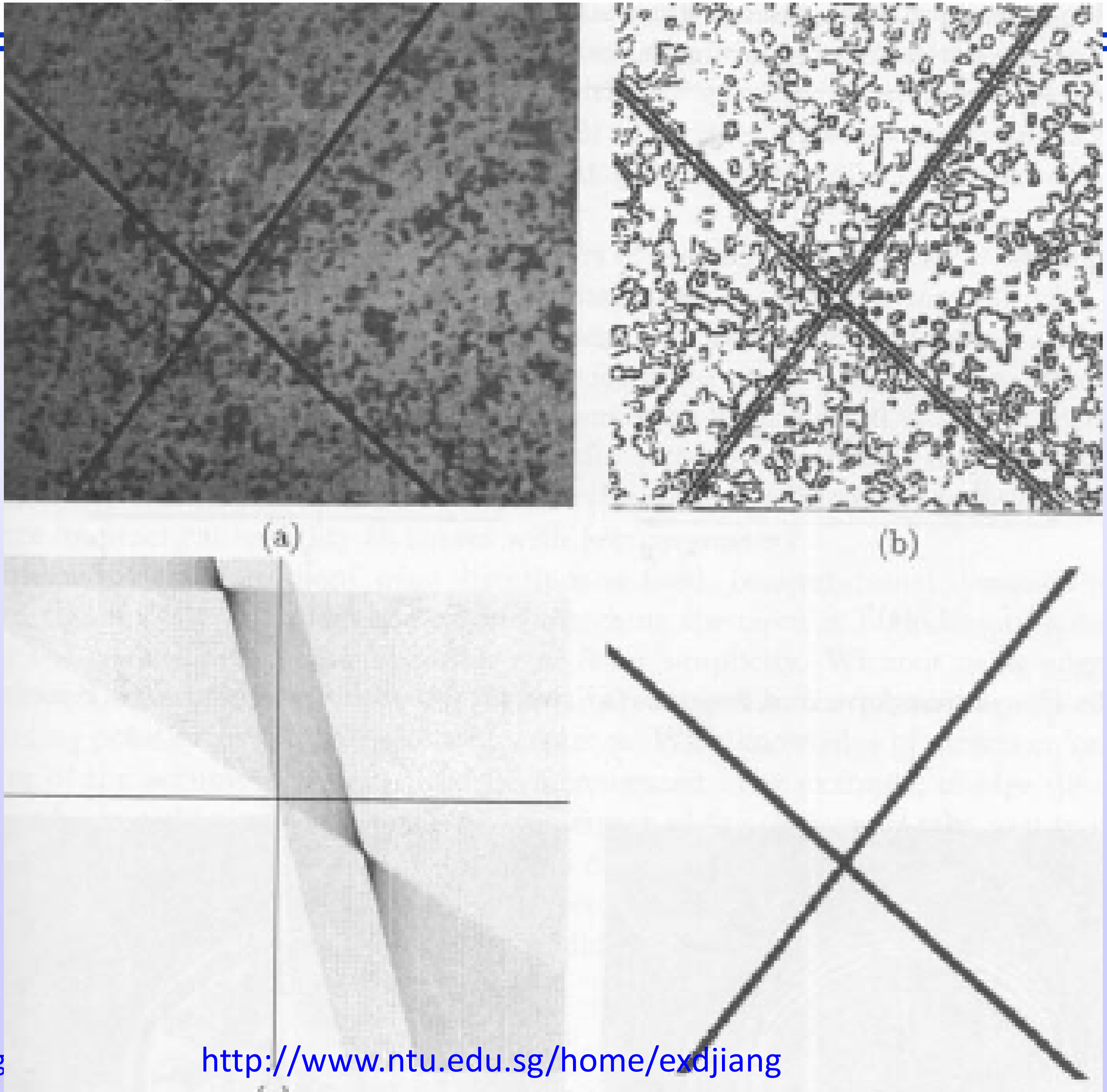
no gaps were longer than 5 pixels



exdjiang@ntu.edu.sg

7 Image Analysis –Edge Detection by Hough Transform

Further example

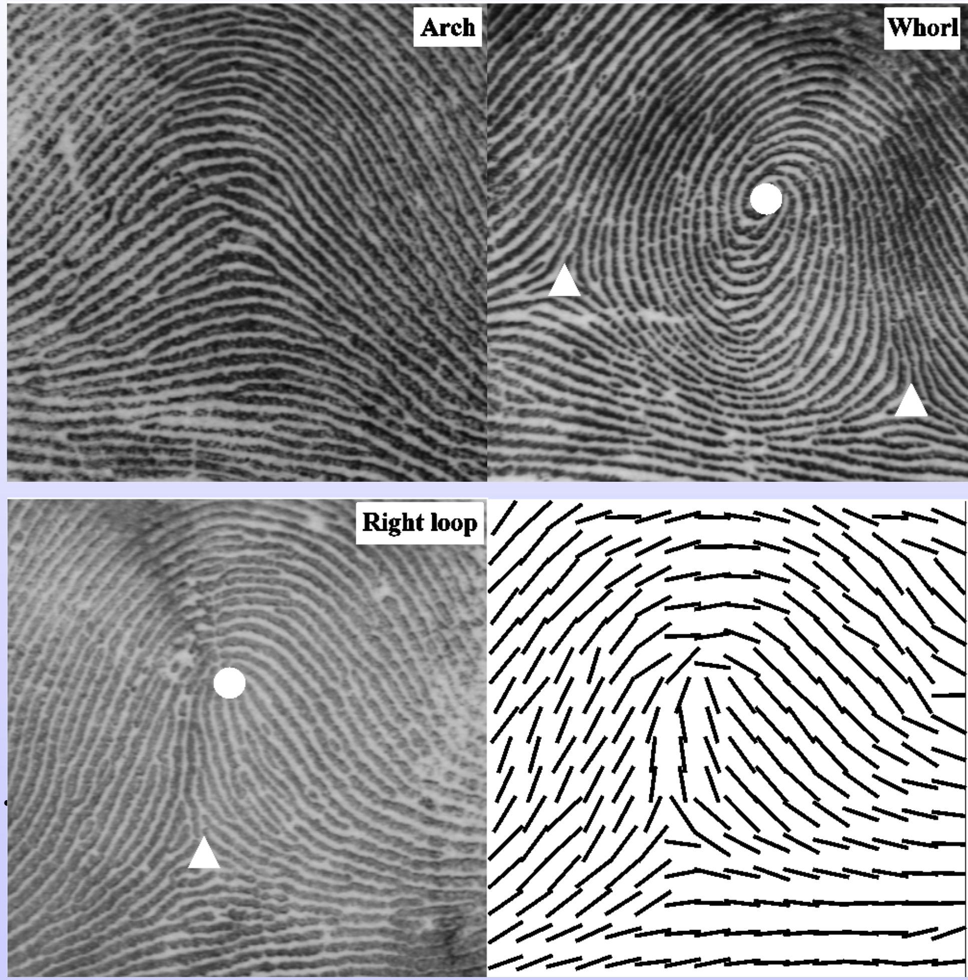


7 Image Analysis–Local Dominant Orientation

Fingerprint classification: for example, arch, whorl and loop

- Local orientations are intrinsic features for such task.

Orientations denoted by short lines.



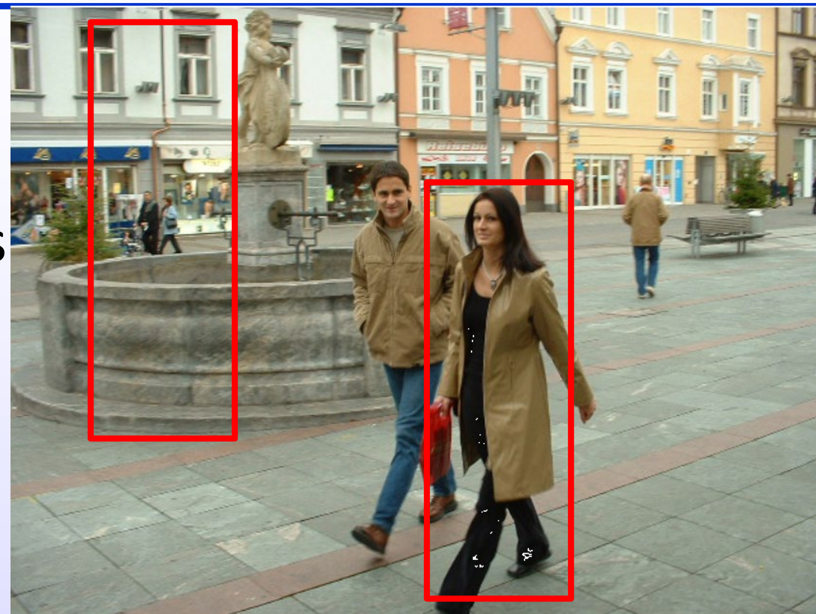
X.D. Jiang, M. Liu and A. Kot, Fingerprint Retrieval for Identification, *IEEE Transactions on Information Forensics and Security*, vol. 1, no. 4, pp. 532-542, December 2006.

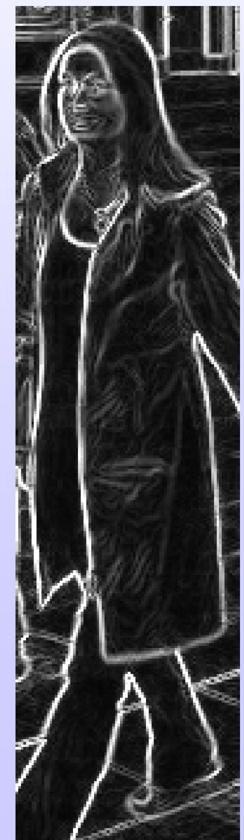
M. Liu, X.D. Jiang and A. Kot, Efficient Fingerprint Search Based on Database Clustering, *Pattern Recognition*, vol. 40, no. 6, pp. 1793-1803, June 2007.

7 Image Analysis–Local Dominant Orientation

Human Detection: Local orientations are intrinsic features for such task.

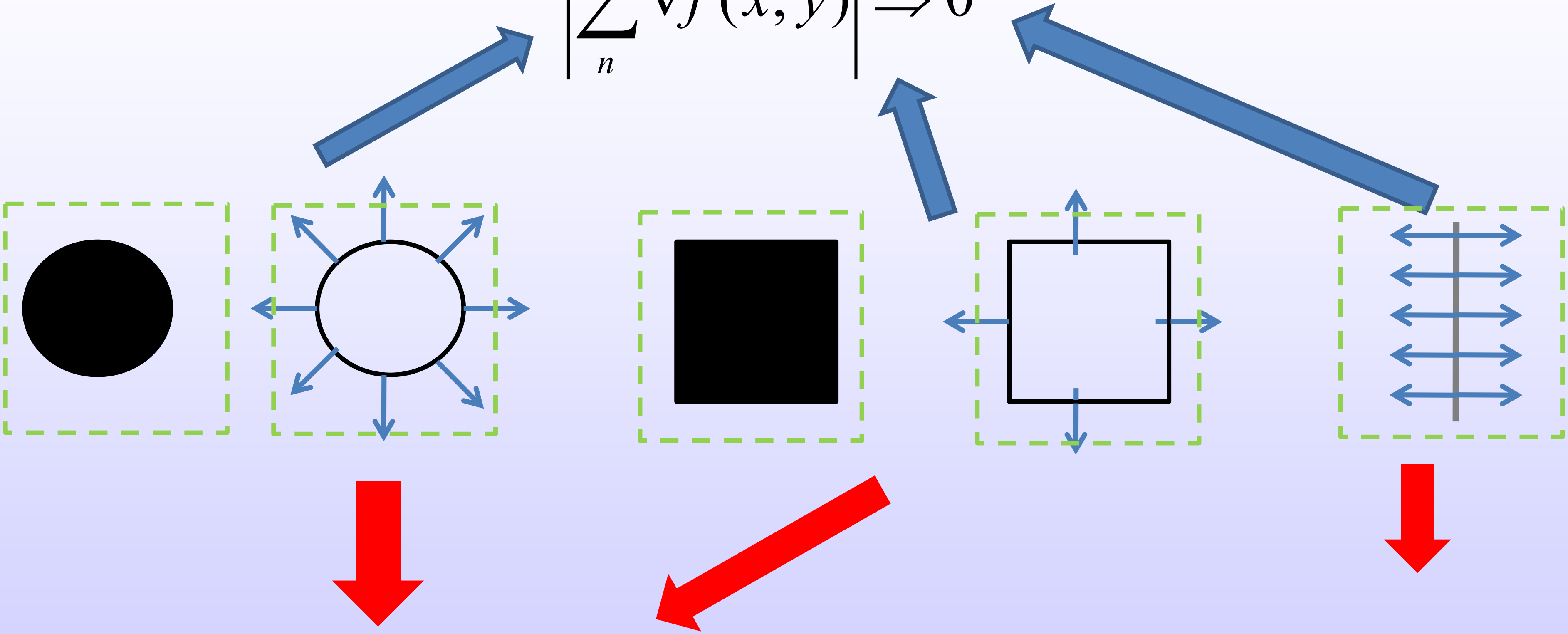
A. Satpathy, X.D. Jiang and H. Eng, "Human Detection by Quadratic Classification on Subspace of Extended Histogram of Gradients," ***IEEE Transactions on Image Processing***, vol. 23, no. 1, pp. 287-297, January, 2014.





7 Image Analysis–Local Dominant Orientation

$$\left|\sum_n \nabla f(x,y)\right| \Rightarrow 0$$

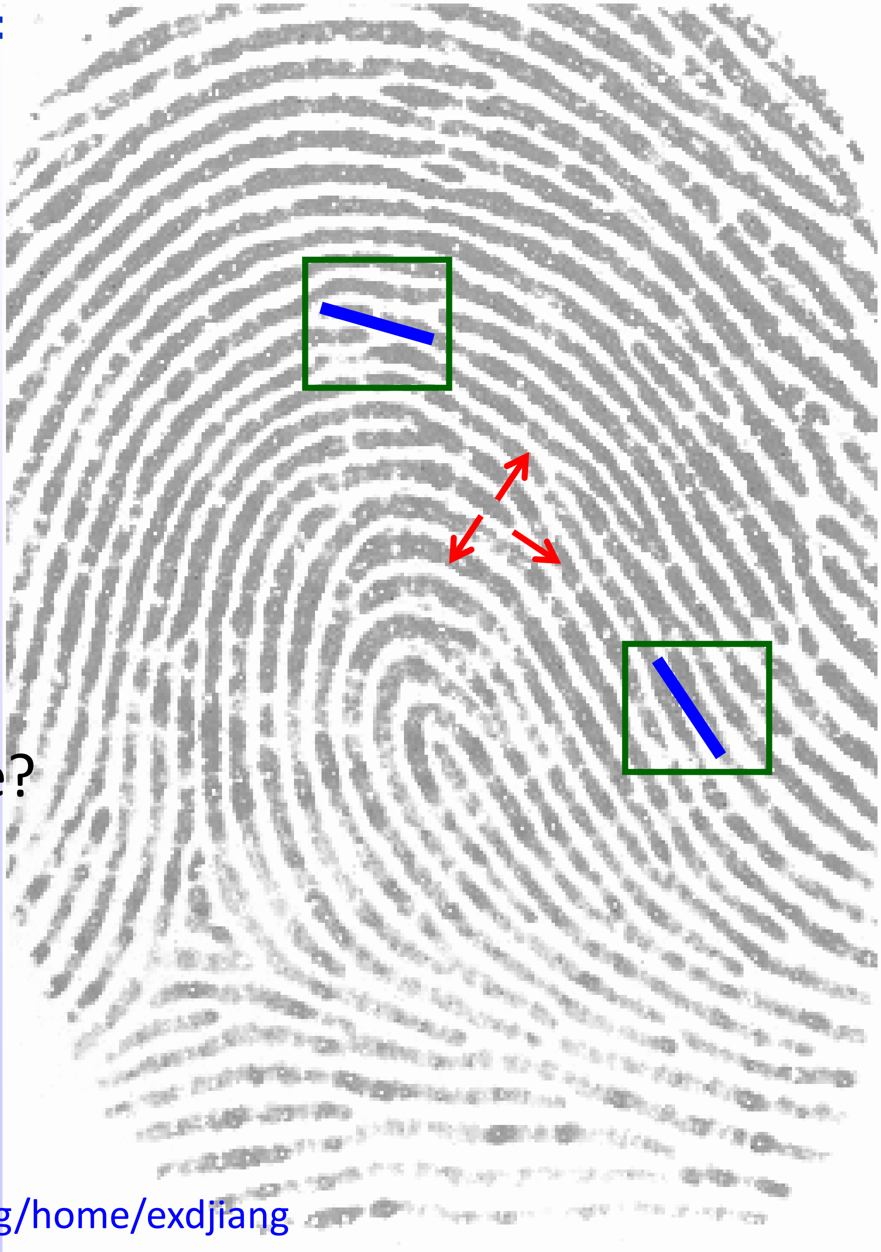


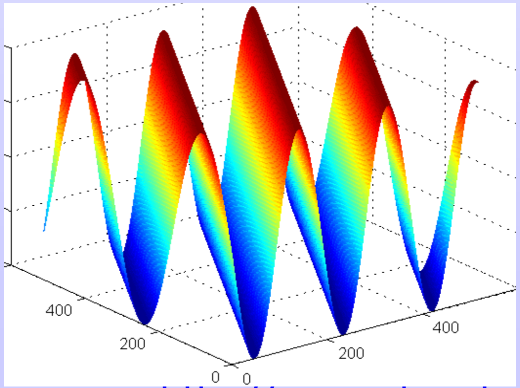
$$\left|\sum_n [\nabla f(x,y)]^2\right| \Rightarrow 0$$

$$\left|\sum_n [\nabla f(x,y)]^2\right| \Rightarrow n[\nabla f(x,y)]^2$$

7 Image Analysis–Local Dominant Orientation

- Local orientations $\varphi(x,y)$, $0° < \varphi(x,y) \leq 180°$ of edges and lines are important image features.
- The gradient vectors are very noisy and a same orientation may be represented by gradients with opposite directions.
- How to extract stable and robust dominant (main) orientation information of a local area of image?



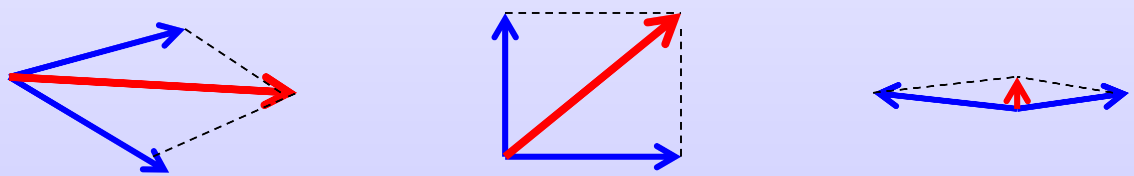


7 Image Analysis–Local Dominant Orientation

- Should we smooth the image first then take gradient or smooth the gradient image?

$$\nabla(h(x,y) * f(x,y)) = h(x,y) * \nabla f(x,y) = \sum_{(s,t)\in S_{xy}} w(s,t)\nabla f(s,t)$$

- The output of any linear filter is a weighted average of all inputs. What is the problem of average the gradient vector?



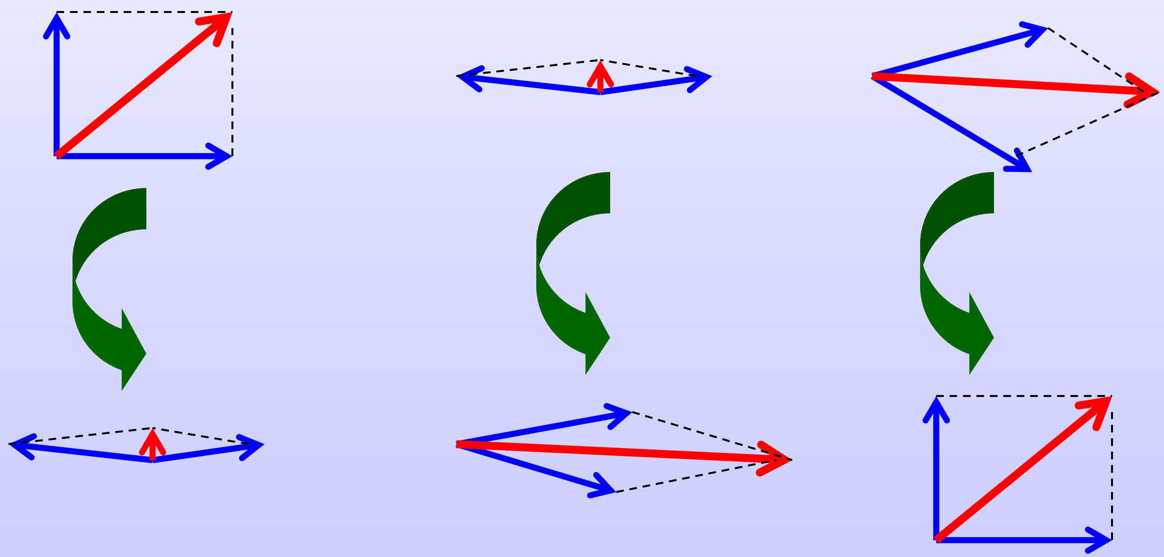
- Problem is the directions of gradient $\theta(x,y)$, $0° < \theta(x,y) \leq 360°$ but the orientations of edges or lines $\varphi(x,y)$, $0° < \varphi(x,y) \leq 180°$.

7 Image Analysis–Local Dominant Orientation

- One solution is to smooth or average the squared gradient vectors.
  What is the square of a vector?
- The vector direction angle will be doubled. Thus:

Average:

Squared average:



7 Image Analysis–Local Dominant Orientation

- Mathematically, let's represent the gradient vector by a complex variable.

$$\nabla f(x, y) = G_x(x, y) + jG_y(x, y)$$

$$= |\nabla f(x, y)| e^{j\theta(x,y)}, \quad \text{where } j = \sqrt{-1}$$

- Its square is then

$$[\nabla f(x, y)]^2 = G_x^2(x, y) - G_y^2(x, y) + j2G_x(x, y)G_y(x, y)$$

$$= |\nabla f(x, y)|^2 \, e^{j2\theta(x,y)}$$

- Obviously:

$$\theta(x, y) = \frac{1}{2}\tan^{-1}\left(\frac{2G_x(x, y)G_y(x, y)}{G_x^2(x, y) - G_{yy}^2(x, y)}\right)$$

7 Image Analysis–Local Dominant Orientation

- Now, the average of the squared gradient is

$$\overline{[\nabla f(x,y)]^2} = \sum_{(s,t)\in S_{xy}} [G_x^2(s,t) - G_y^2(s,t) + j2G_x(s,t)G_y(s,t)]$$

$$= A - B + j2C$$

Therefore, the dominant orientation is determined by

$$\overline{\theta(x,y)} = \frac{1}{2}\tan^{-1}\left(\frac{2C}{A-B}\right)$$

where

$$A = \sum_{(s,t)\in S_{xy}} G_x^2(s,t), \quad B = \sum_{(s,t)\in S_{xy}} G_y^2(s,t), \quad C = \sum_{(s,t)\in S_{xy}} G_x(s,t)G_y(s,t)$$

7 Image Analysis–Local Dominant Orientation

An continuous image is modeled locally by $f(x,y)=\sin(ax+by)$

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

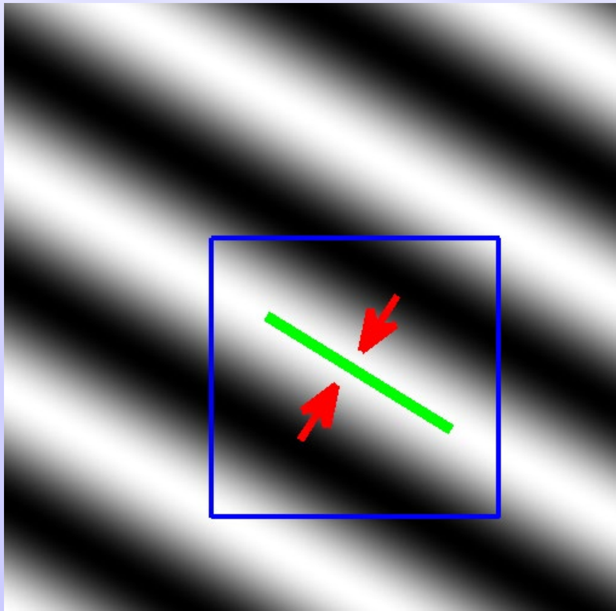
$$=a\cos(ax+by)+jb\cos(ax+by)$$

$$\theta(x,y)=\arctan\frac{\operatorname{Im}\{\nabla f(x,y)\}}{\operatorname{Re}\{\nabla f(x,y)\}}$$

$$=\arctan\frac{b}{a}=\varphi+90^\circ$$

$$\frac{\operatorname{Im}\left\{\int_y^{y+w}\int_x^{x+w}\nabla f(u,v)dudv\right\}}{\operatorname{Re}\left\{\int_y^{y+w}\int_x^{x+w}\nabla f(u,v)dudv\right\}}=\frac{0}{0}$$

$$\frac{\operatorname{Im}\left\{\int_y^{y+w}\int_x^{x+w}[\nabla f(u,v)]^2\,dudv\right\}}{\operatorname{Re}\left\{\int_y^{y+w}\int_x^{x+w}[\nabla f(u,v)]^2\,dudv\right\}}=\frac{2ab}{a^2-b^2}=\tan 2\theta(x,y)$$



7 Image Analysis–Local Dominant Orientation

- Obviously, the magnitude of the average squared gradient is

$$\left|\overline{[\nabla f(x,y)]^2}\right| = \sqrt{(A-B)^2 + 4C^2}$$

- And the average magnitude of squared gradients is

$$\overline{|\nabla f(x,y)|^2} = A + B$$

- The following amount indicates how all gradients in a local area point to a same orientation, called anisotropy or coherence of an image local area.

$$coh(x,y) = \frac{\left|\overline{[\nabla f(x,y)]^2}\right|}{\overline{|\nabla f(x,y)|^2}} = \frac{\sqrt{(A-B)^2 + 4C^2}}{A+B}$$

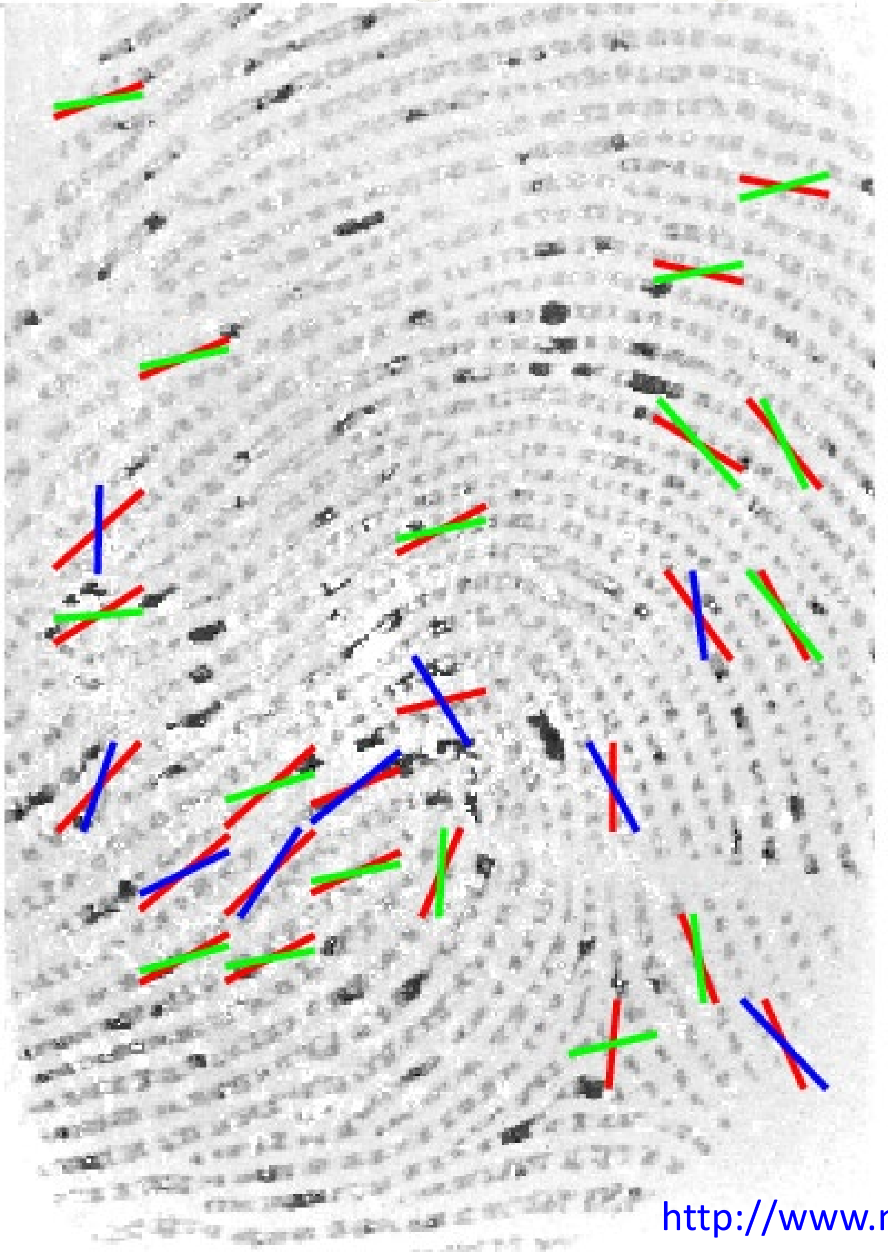
7 Image Analysis–Local Dominant Orientation

- What is the minimum and maximum values of $coh(x,y)$?
- In what cases $coh(x,y)$ reaches to the minimum or maximum value?
- How to choose the appropriate size of the average window $S(x,y)$?
- What factor will affects the selection of the $S(x,y)$?
- What problems of this technique may have?

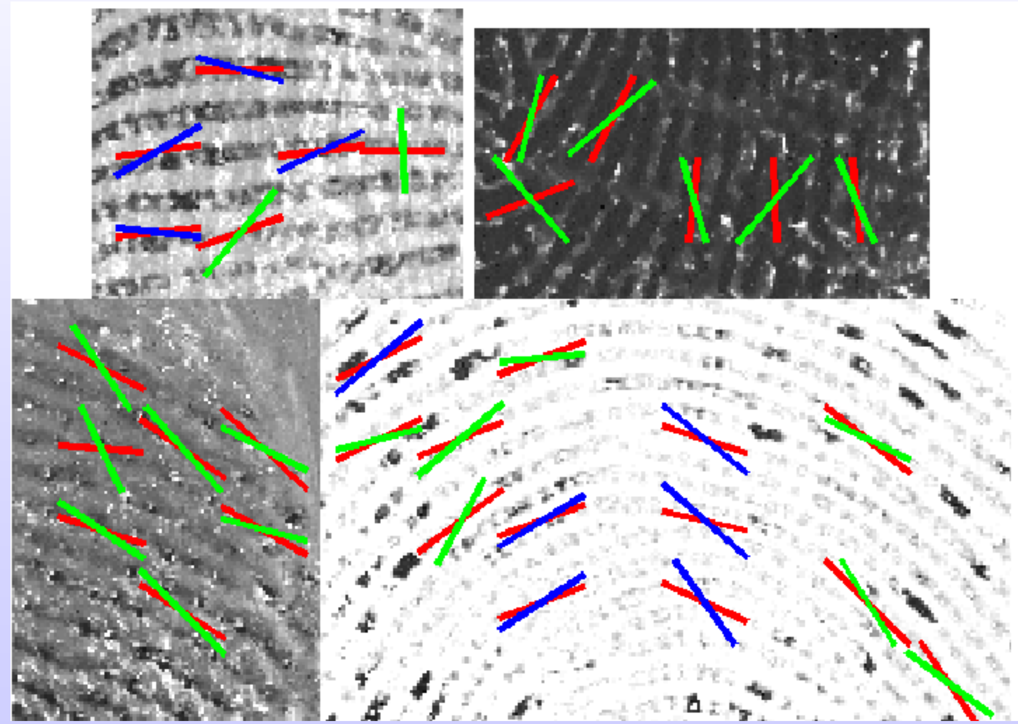
- Rresearch on this topic can be found in the research publication:

X.D. Jiang, "On Orientation and Anisotropy Estimation for Online Fingerprint Authentication," *IEEE Transactions on Signal Processing,* Vol. 53, No. 10, pp. 4038- 4049, October 2005.

7 Image Analysis–Local Dominant Orientation



Examples of different methods



7 Image Analysis–Local Dominant Orientation

A noise free face image and its noising version

7 Image Analysis–Local Dominant Orientation

Extracted significant orientations represented by short lines of the noise free image by two different gradient operators

7 Image Analysis–Local Dominant Orientation

Extracted significant orientations represented by short lines of the noising face image by two different gradient operators