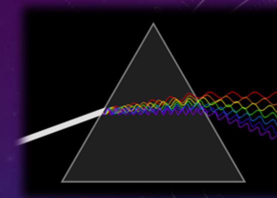
DIGITAL IMAGE PROCESSING REVIEW

CHANG SHU

2022-12-09

BASIC CONCEPTS

BASIC CONCEPTS



$$I = f(x, y, \lambda, t)$$

Intensity — I; **Coordinates** — x, y; **Wavelength** — λ; **Time** — t

- *I*(.) represents **the spatial energy distribution** of an image source of **radiant**.
- An image can be seen as a distributed amplitude of color(s).

BASIC CONCEPTS

- **Image**
  - **Still / Sequential**
  - **Analog / Digital**
  - **Two-dimensional / Three-dimensional**
  - **Binary / Gray / Color**
  - …

BASIC CONCEPTS

$$I = f(x, y, \lambda, t)$$

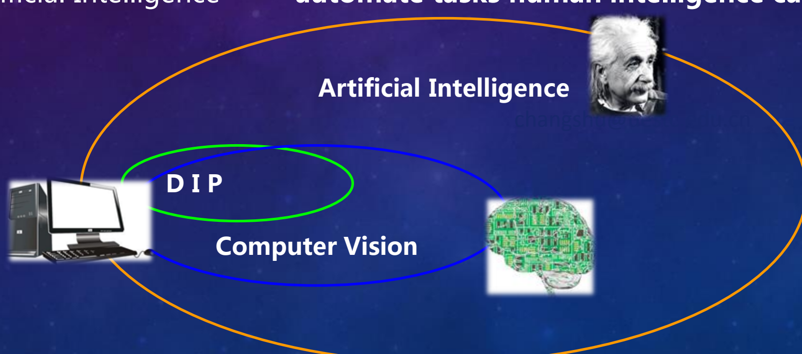
- **Digital image (Two-dimensional)**
  - **When the value of spatial coordinates and amplitude of I are both finite, discrete, it is called a digital image.**
  - **Digital images are constructed of finite elements (pixels) .**
  - **Each pixel has a particular location and value.**

$$f(x, y) \longrightarrow I(m, n)$$

$$I(m,n)=\begin{bmatrix} I(0,0) & I(0,1) & \cdots & I(0,N-1) \\ I(1,0) & I(1,1) & \cdots & I(1,N-1) \\ \vdots & & & \\ I(M-1,0) & I(M-1,1) & \cdots & I(M-1,N-1) \end{bmatrix}$$

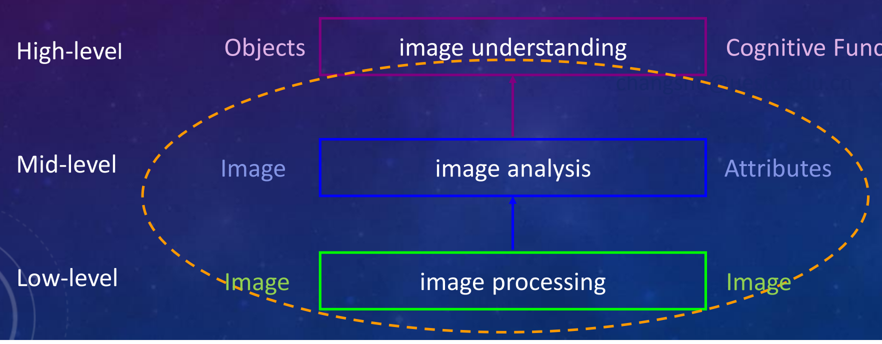
WHAT – ORIENTATION

- **Related areas**
  - Computer Vision —— **automate tasks human vision system can do**
  - Artificial Intelligence —— **automate tasks human intelligence can do**



WHAT – ORIENTATION

- **DIP domain**
  - **Narrow sense —— improvement of image quality for human interpretation, processing of image data for storage, transmission, display and representation for machine perception**
  - **Broad sense —— includes fields where image processing ends and image analysis and image understanding starts.**



TASKS AND GOALS

- **Improve image quality**
- **Extract target features**
  - human
  - machine
- **Information visualization**
- **Information security**

IMAGE PROCESSING SYSTEM

Interactive control

Input

Processor

Output

Storage

IMAGE PROCESSING SYSTEM

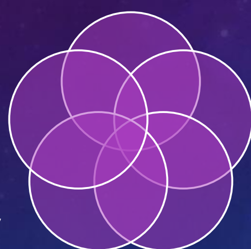
- **Image Input**
  - **Sensors**
    - **CCD**（Charge Coupled Device）
    - **CMOS（Complementary Metal-Oxide Semiconductor）**
    - **InfraRed**
  - **Ways of input**
    - **Fly-point scanning**
    - **Line sensor**
    - **2D sensor**

APPLICATION

Astronomy

Industrial inspection

Remote Sensing



Military affairs

Medical imaging

OCR

Robot vision

Biometrics

Intelligent monitoring

Image/video Communication

……

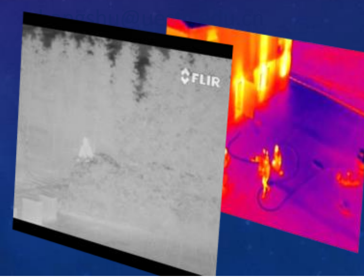
SUMMARY

- Digital image/ Digital image processing
- History of digital image processing
- Components of an image processing system
- Applications of digital image processing

VISION - LOW LEVEL

- ElectroMagnetic Spectrum
- Visual band 350 nm ~ 780 nm

Photon
光子



VISION AND VISION PROPERTIES

Vision perception

- Low-level
- High-level

Vision Properties

- Intensity property
- Frequency properties
  - Space
  - Wavelength
- Temporal property
- Color response

VISION - LOW LEVEL

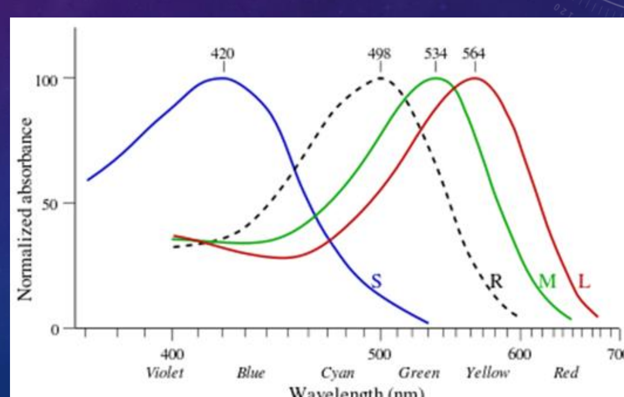
Photoelectrochemical reaction
光电化学反应

- Retina translates 2D image into impulses
- Optical nerve sends those impulses into the brain to create visual perception
- Two receptors – Rods & Cones
  - Rods： thin and long， ~120 million， Scotopic Vision， high sensitivity， achromatic， low acuity（spatial acuity）
  - Cones： thick and short， 6~7 million， Photopic Vision， lower absolute sensitivity， chromatic， high acuity

VISION - LOW LEVEL (REVIEW)

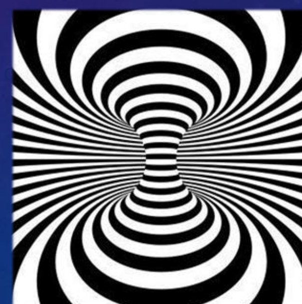
trichromatic theory
三原色理论

- **Two receptors – Rods & Cones**
  - Type： Rods(1) v.s. Cones(3)



VISION - HIGH LEVEL

- Some researchers believe that "Thinking is part of perception."
- Visual perception is also "visual thinking", not a simple copy action of stimulus, our past experience involves in our current visual perception



SUMMARY

- EM spectrum, visible band
- Eye physiology
- Retina, Rods & Cones
- Blind spot, periphery vision, color
- Eye v.s. Digital camera

SUMMARY

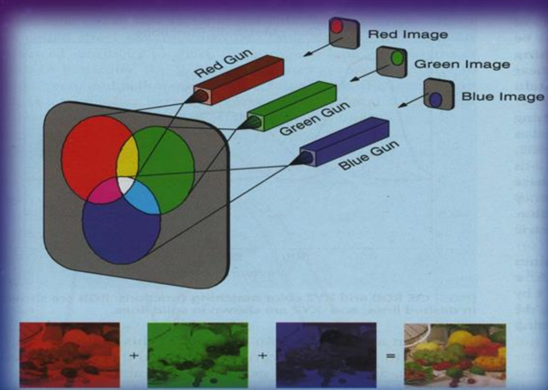
- Vision properties
  - Contrast sensitivity
  - Simultaneous Contrast
  - Chromatic Adaption
  - Modulation Transfer Function
  - Mach Band
  - Critical Fusion Frequency
  - Relative luminous efficiency function

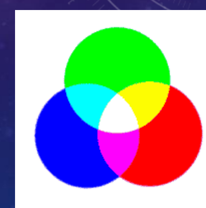
19

PHOTOMETRY & COLORIMETRY

- Basic concepts of colorimetry
  - Color = brightness + chromaticity
  - chromaticity = hue + saturation
  - Color = brightness + hue + saturation
  - Radiance, luminance, brightness, -> energy, perceived energy, a subjective descriptor
  - Hue is the attribute of light.
  - saturation refers to the relative purity or the amount of white light mixed with a hue.

ADDITIVE COLOR MATCHING





21

dyestuffs染料

SUBTRACTIVE COLOR MATCHING

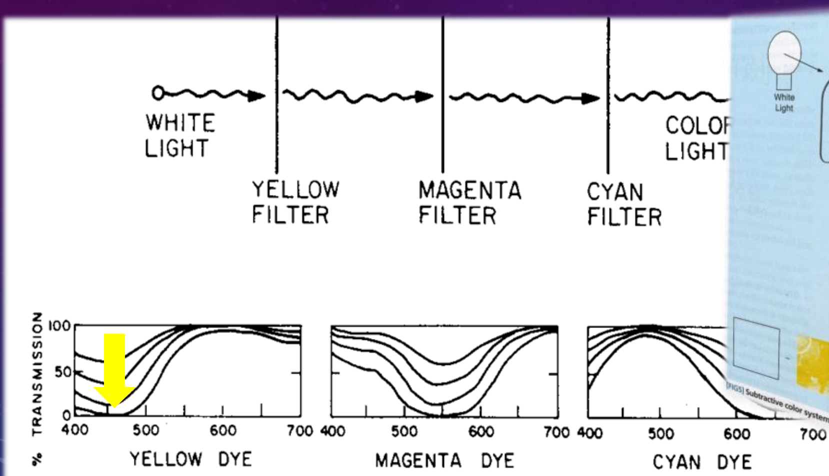
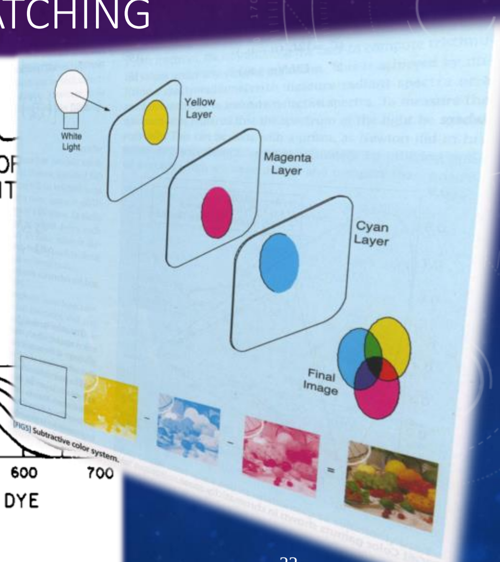


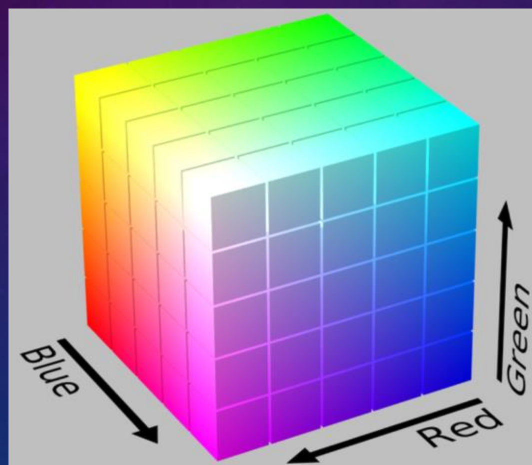
FIGURE 3.2-3. Subtractive color matching.



22

COLOR SPACE

- RGB
- CMY/CMYK
- HSI/HSV



23

COLOR SPACE

CMY

- CMY(cyan/magenta/yellow) model is for color printing.
- The conversion from RGB to CMY is :

  C=1.0-R; M=1.0-G; Y=1.0-B

Where the R,G,B values are in the range [0.0,1.0]

- CMYK

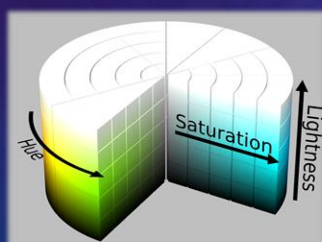
$$C = 1.0 - R - uK_b \qquad M = 1.0 - G - uK_b \qquad Y = 1.0 - B - uK_b \qquad K = bK_b$$

$$Where \;\; K_b = \min\{1.0 - R, 1.0 - G, 1.0 - B\} \qquad 0 \le u, b \le 1$$

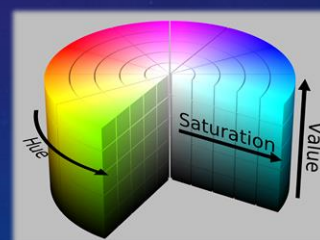
24

COLOR SPACE
HSI/HSV

- RGB alternatives
  - HSI(hue, Saturation, Intensity/Lightness)
  - HSV(hue, Saturation, Value)



HSI cylinder



HSV cylinder

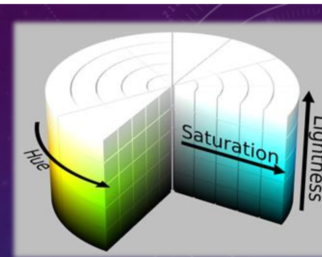
25

COLOR SPACE
HSI

- RGB alternatives
- Decouples the intensity component
- Natural and intuitive to humans

26

COLOR SPACE

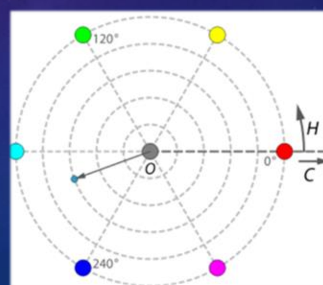


- Converting colors from RGB to HSI

$$H=\begin{cases}\theta & if\ \ B\le G\\ 360-\theta & if\ \ B>G\end{cases}$$

$$\theta=\cos^{-1}\left\{\frac{\frac{1}{2}(R-G+R-B)}{\sqrt{(R-G)^2+(R-B)(G-B)+\varepsilon}}\right\}$$

$$S=1-\frac{3}{R+G+B}[\min(R,G,B)]$$

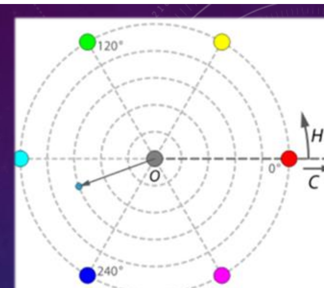


$$0\le R,G,B\le 1$$

$$I=\frac{1}{3}(R+G+B)$$

27

COLOR SPACE



- Converting colors from HSI to RGB

$$X=I\left[1+\frac{S\cos H'}{\cos(60^\circ-H')}\right]$$

$$H'=H\ \bmod 120 \qquad H\in[0^\circ,360^\circ]$$

$$Y=I(1-S)$$

$$quotient=\begin{cases}0 & X,Y,Z\ \ ->R,B,G\\ 1 & X,Y,Z\ \ ->G,R,B\\ 2 & X,Y,Z\ \ ->B,G,R\end{cases}$$

$$Z=3I-(X+Y)$$

$$H\div 120=quotient\cdots H'$$

28

SUMMARY

- Eye physiology
  - Retina, Rods & Cones
  - Blind spot, periphery vision, optic illusion
- Vision Properties
  - Contrast sensitivity
  - Simultaneous Contrast
  - Chromatic Adaption
  - Modulation Transfer Function
  - Mach Band
  - Critical Fusion Frequency
  - Relative luminous efficiency function
- Color & Color Matching & Color Space

IMAGE DIGITALIZATION

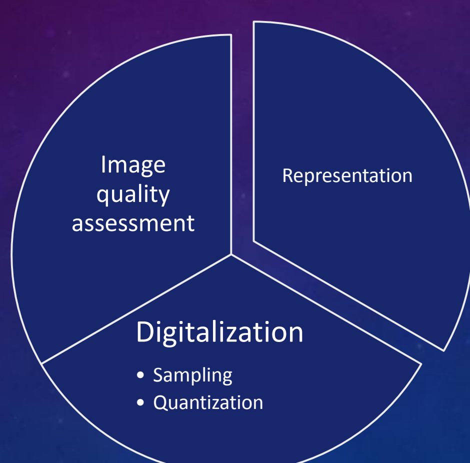


IMAGE QUALITY ASSESSMENT

- Fidelity，processed image v.s. original image
- Intelligibility，information conveyed

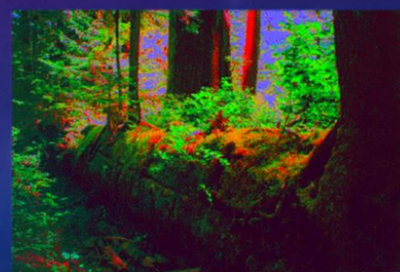


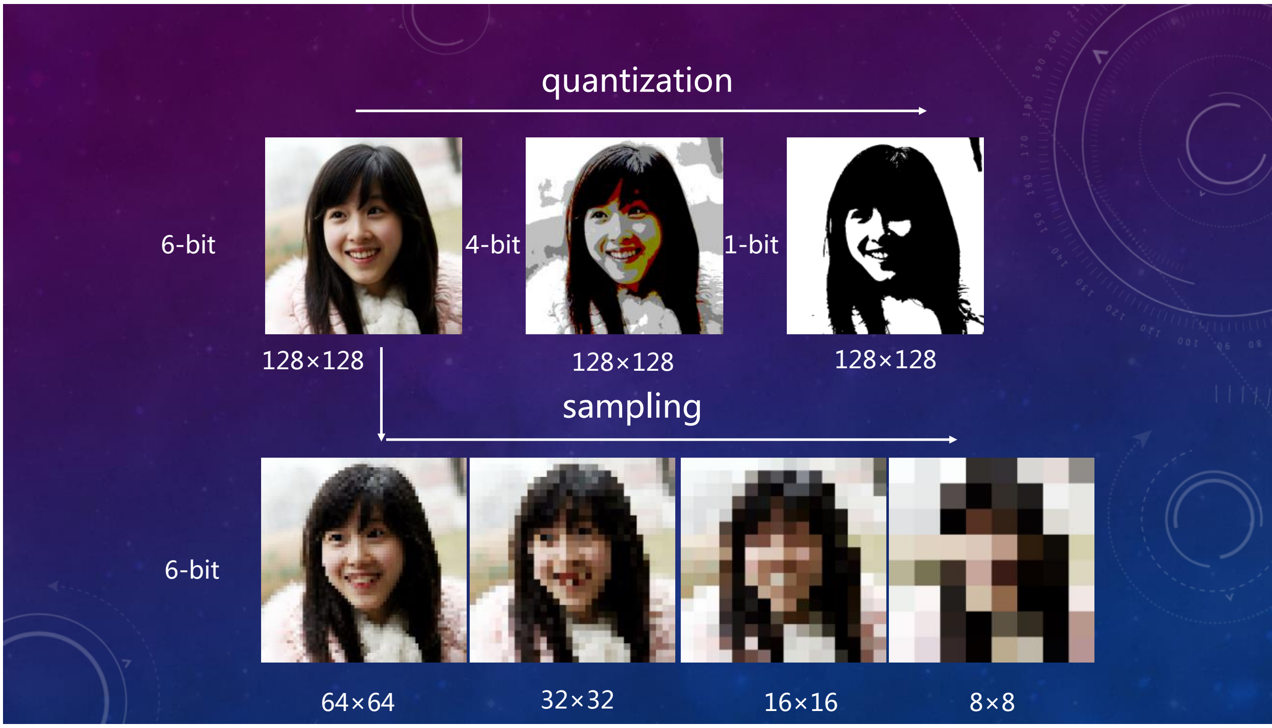
IMAGE QUALITY ASSESSMENT

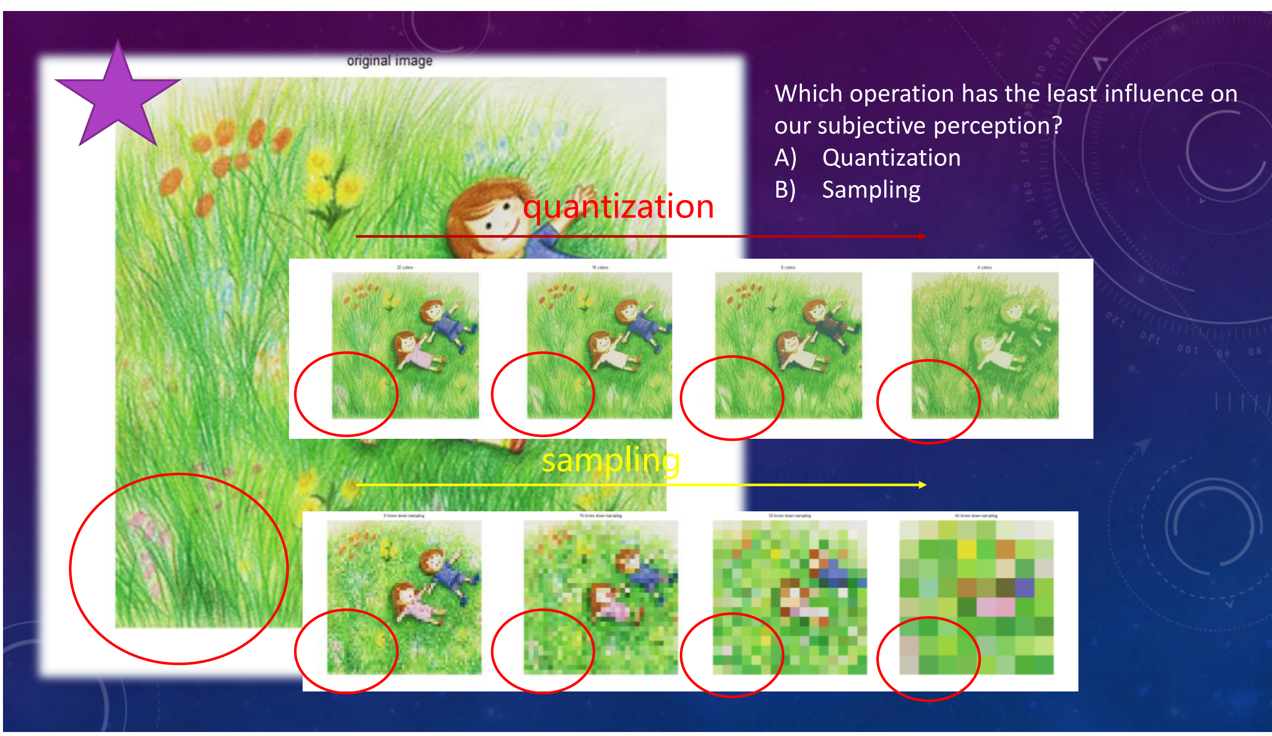
- Objective way

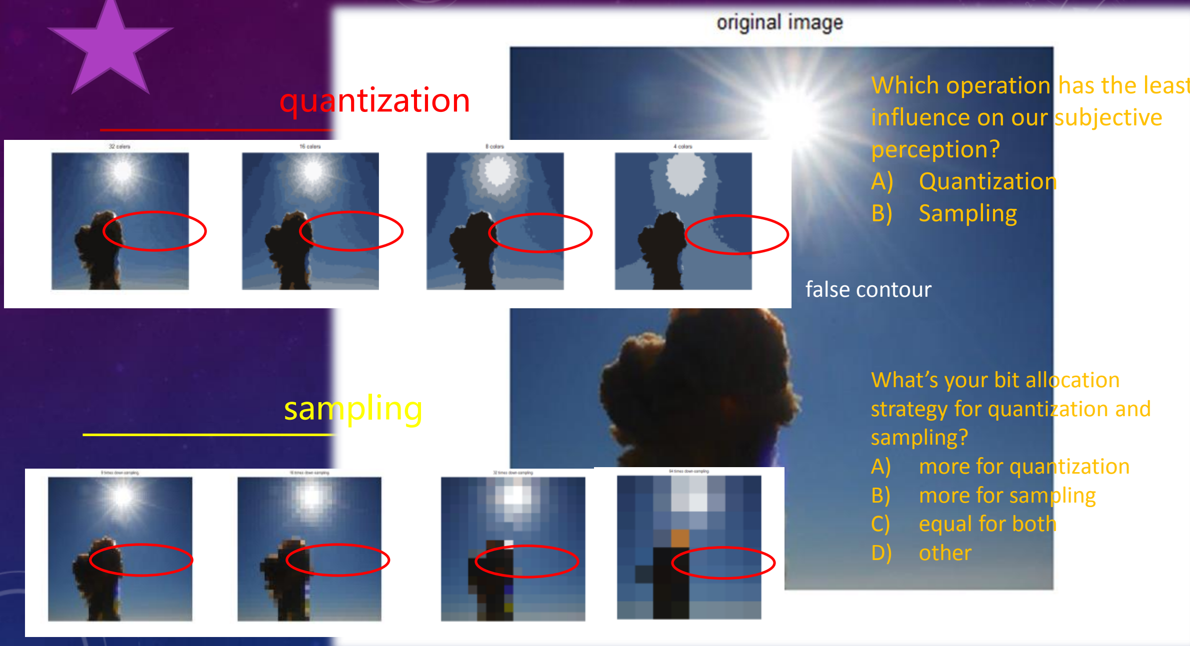
For $M \times N$ digital images，*root-mean-square error* (均方根误差)and *mean-square signal-to-noise ratio* (均方信噪比) are often used

$$e_{rms} = \sqrt{\frac{1}{MN}\sum_{i=0}^{N-1}\sum_{j=0}^{M-1}\left\{\underset{\uparrow\ \text{Preprocessing function}}{Q}[\overset{\text{original}}{\overbrace{f(i,j)}}] - Q[\overset{\text{processed}}{\overbrace{\hat{f}(i,j)}}]\right\}^2}$$

$$Q(f) = K_1 \log_b [K_2 + K_3 f(i,j)]$$


quantization
6-bit
4-bit
1-bit
128×128
128×128
128×128
sampling
6-bit
64×64
32×32
16×16
8×8


original image
quantization
sampling
Which operation has the least influence on our subjective perception?
A) Quantization
B) Sampling

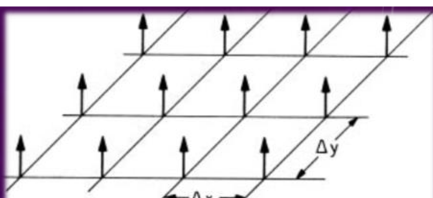


Which operation has the least influence on our subjective perception?

A) Quantization
B) Sampling

What's your bit allocation strategy for quantization and sampling?

A) more for quantization
B) more for sampling
C) equal for both
D) other



2D SAMPLING

Spatical domain

$$f(x,y)$$

$$\times \; comb(x,y,\Delta x,\Delta y) = \sum_m \sum_n \delta(x - m\Delta x, y - n\Delta y)$$

$$f_s(x,y) = \sum_m \sum_n f(m\Delta x, n\Delta y)\delta(x - m\Delta x, y - n\Delta y)$$

Frequency domain

$$F(u,v)$$

$$\otimes \; COMB(u,v) = \frac{1}{\Delta x \Delta y} \sum_k \sum_l \delta(u - k\frac{1}{\Delta x}, v - l\frac{1}{\Delta y})$$

$$F_s(u,v) = \frac{1}{\Delta x \Delta y} \sum_k \sum_l F(u - k\frac{1}{\Delta x}, v - l\frac{1}{\Delta y})$$

2D SAMPLING

$f_s(x,y) =$

$$\sum_m \sum_n f(m\Delta x, n\Delta y)\delta(x - m\Delta x, y - n\Delta y)$$

$\otimes$

$$h(x,y) = \sin c(f_{sx}x)\sin c(f_{sy}y)$$

$f_r(x,y) =$

$$\sum_m \sum_n f(m,n)Sinc(xf_{sx} - m)Sinc(yf_{sy} - n)$$

$$F_s(u,v) = \frac{1}{\Delta x \Delta y}\sum_k \sum_l F(u - k\frac{1}{\Delta x}, v - l\frac{1}{\Delta y})$$

$\times$

$$H(u,v) = \begin{cases} \Delta x \Delta y, & |u| < ½ f_{sx}, |v| \le ½ f_{sy} \\ 0 & otherwise \end{cases}$$

$$f_{sx} < 2 f_{mx}, \quad f_{sy} > 2 f_{my}$$

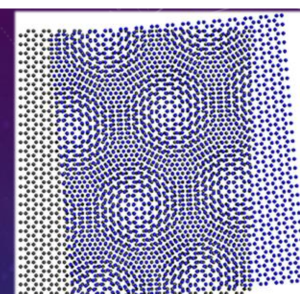
$$F_r(u,v)$$

Nyquist criterion

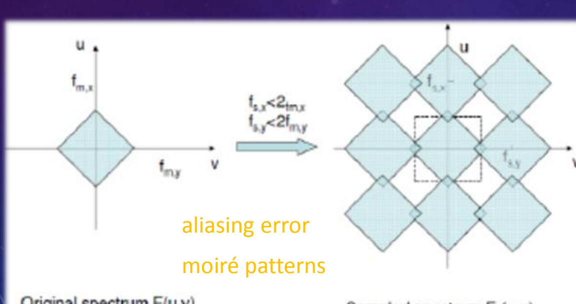
2D SAMPLING

$$F(u,v) = \begin{cases} F(u,v) & |u| < f_{mx}, |v| \le f_{my} \\ 0 & |u| > f_{mx}, |v| > f_{my} \end{cases}$$

$$F(u,v)$$



$$\otimes \; COMB(u,v) = \frac{1}{\Delta x \Delta y}\sum_k \sum_l \delta(u - k\frac{1}{\Delta x}, v - l\frac{1}{\Delta y})$$



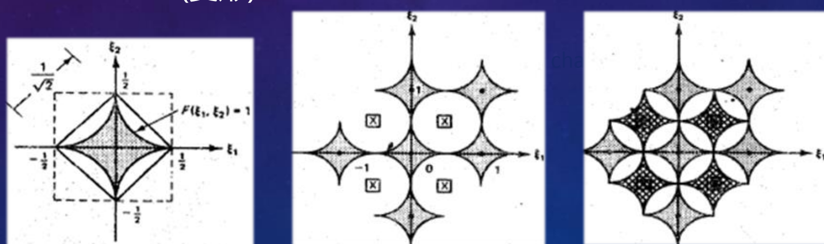
undersampled

$F_s(u,v) =$

$$\frac{1}{\Delta x \Delta y}\sum_k \sum_l F(u - k\frac{1}{\Delta x}, v - l\frac{1}{\Delta y})$$

SAMPLING EFFICIENCY

- When the nonzero area of the signal spectrum has a square shape , rectangular sampling is the most efficient way
- Can you find a different sampling strategy more efficient for other spectral shapes ?
  - Rhombus(菱形)



QUANTIZATION

- Uniform
- Symmetric
- With Memory
- Scalar

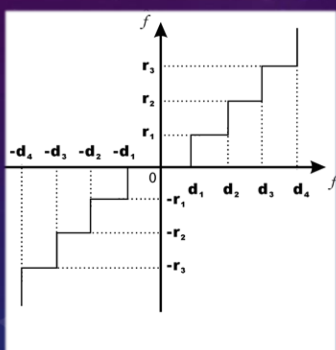
- Non-uniform
- Asymmetric
- Memoryless
- Vector

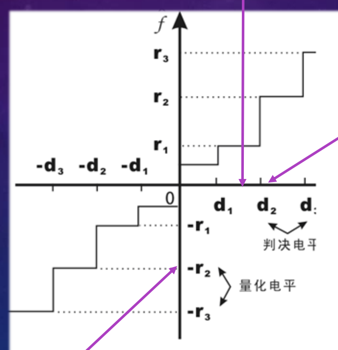
The conversion process between analog samples and discrete-valued samples is called quantization

QUANTIZATION

quantization interval $\Delta_i = d_{i+1} - d_i$

$d \in [d_i, d_{i+1})$ $\quad d \to r_i$





decision level d_i

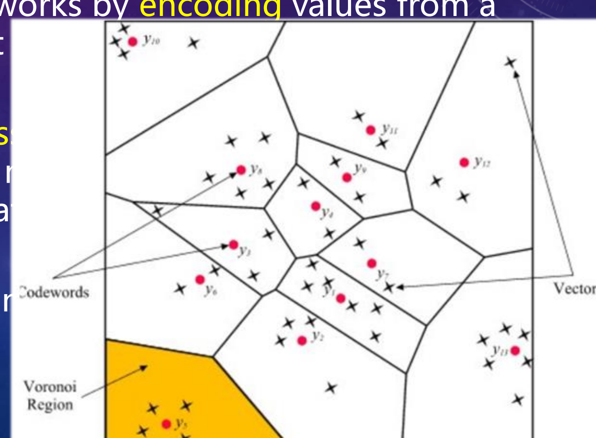
quantization error $e = d - Q(d)$

quantization noise $\varepsilon^2 = E[e^2]$

reconstruction level r_i

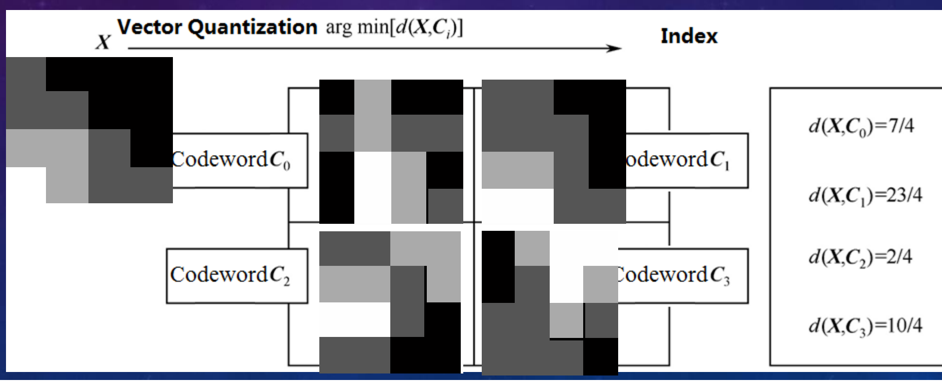
QUANTIZATION

- Vector Quantization(VQ) ---- Also called *block quantization* or *pattern matching quantization*, works by encoding values from a MD vector space into a finite set
subspace of lower dimension
- A vector in subspace requires les
compressed. Due to the density
quantization, the compressed da
proportional to density
- The transformation is usually do
codebook



QUANTIZATION

- Vector Quantization(VQ) ---- works by encoding values from a MD vector space into a finite set of values from a discrete subspace of lower dimension.



SUMMARY

- Digitalization
  - Sampling (rectangular/hexagonal, sampling density/efficiency)
  - Quantization (vector quantization, halftone, dither)
- To balance sampling and quantization, we have to take the image content into consideration

IMAGE TRANSFORMS

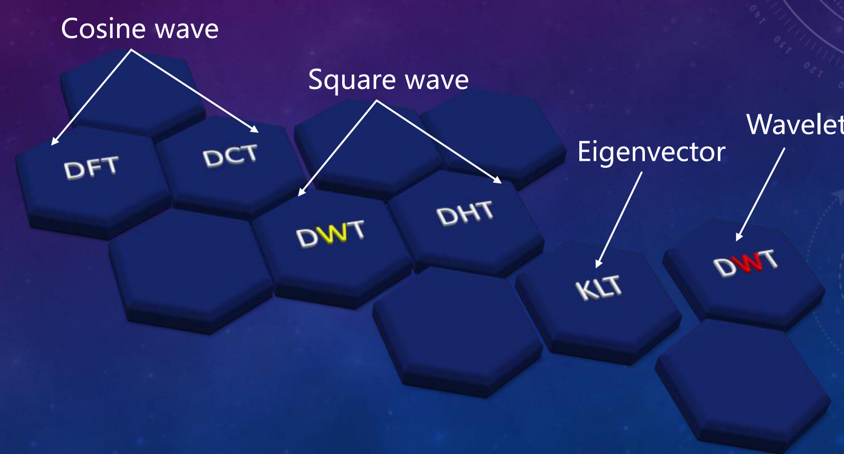


IMAGE TRANSFORMS – DFT2

- DFT for a discrete image with the size of $M \times N$:

$$F(u,v) = \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) e^{-j2\pi(ux/M + vy/N)}$$

$$F(u,v) = \frac{1}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) e^{-j2\pi\frac{(ux+vy)}{N}}$$

- IDFT :

$$f(x,y) = \frac{1}{\sqrt{MN}} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u,v) e^{j2\pi(ux/M + vy/N)}$$

$$f(x,y) = \frac{1}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} F(u,v) e^{j2\pi\frac{(ux+vy)}{N}}$$

IMAGE TRANSFORMS – DFT2

- $F(u, v)$ is usually a complex number : $F(u,v) = R(u,v) + jI(u,v) = |F(u,v)|e^{j\phi(u,v)}$

  Magnitude/Spectrum : $|F(u,v)| = [R^2(u,v) + I^2(u,v)]^{1/2}$

  Phase angle : $\phi(u,v) = \arctan\left[\frac{I(u,v)}{R(u,v)}\right]$

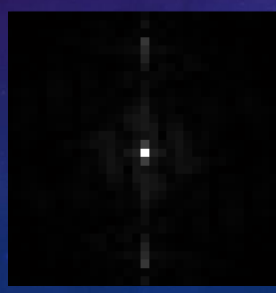
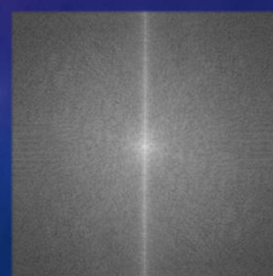
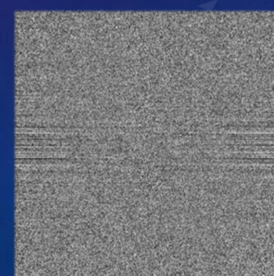





IMAGE TRANSFORMS – DFT2

- |F(u, v)| has even symmetry about the origin
- The DC component F(0, 0) corresponds to the average value of an image
- Translation has no effect on the spectrum of F(u, v)
- Applying DFT twice = image rotated 180 degrees
- Centering the Fourier transform:

$$DFT[f(x,y)(-1)^{x+y}] = F(u - M/2, v - N/2)$$

$$F(u - N/2) = \frac{1}{N}\sum_{x=0}^{N-1} f(x)e^{-j\frac{2\pi}{N}x(u-N/2)} = \frac{1}{N}\sum_{x=0}^{N-1}(-1)^x f(x)e^{-j\frac{2\pi}{N}xu}$$

IMAGE TRANSFORMS – DFT2

- Separability:

Computing 2D DFT by computing 1D DFT twice!

$$f(x,y) \rightarrow F_{\mathrm{col}}[f(x,y)] = F(x,v) \rightarrow F(x,v)^T$$
$$\rightarrow F_{col}[F(x,v)^T] = F(u,v)^T \rightarrow F(u,v)$$

IMAGE TRANSFORMS – DFT2 GENERALIZATION

- In the general form, image transform can be expressed as :
- 1-D linear transform :

Forward transform : $\underset{\text{Transform variable}}{T(u)} = \sum_{x=0}^{N-1} f(x)\underbrace{g(x,u)}_{\text{Forward transformation kernel}} \qquad u = 0,1,\ldots,N-1$

Inverse transform : $f(x) = \sum_{u=0}^{N-1} T(u)\underbrace{h(x,u)}_{\text{Inverse transformation kernel}} \qquad x = 0,1,\ldots,N-1$

- 2-D linear transform :

Forward transform : $T(u,v) = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)g(x,y,u,v) \qquad u = 0,1,\ldots,M-1, \; v = 0,1,\ldots,N-1$

Inverse transform : $f(x,y) = \sum_{u=0}^{M-1}\sum_{v=0}^{N-1} T(u,v)h(x,y,u,v) \qquad x = 0,1,\ldots,M-1, \; y = 0,1,\ldots,N-1$

IMAGE TRANSFORMS – DCT2

- DCT of a $M \times N$ *digital image* :

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

- IDCT :

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

IMAGE TRANSFORMS – DCT2

$$\begin{cases} f(n), 0 \le n \le N-1 \\ f(-n-1), -N \le n \le -1 \end{cases}$$

- The forward/inverse transform kernels of 2D DCT are the same
- The transform kernel of 2D DCT is separable $F = G \cdot f \cdot G^T$
- Flipping has no effect on the 2D DCT of an image

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

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

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

IMAGE TRANSFORMS – DCT2 APPLICATION

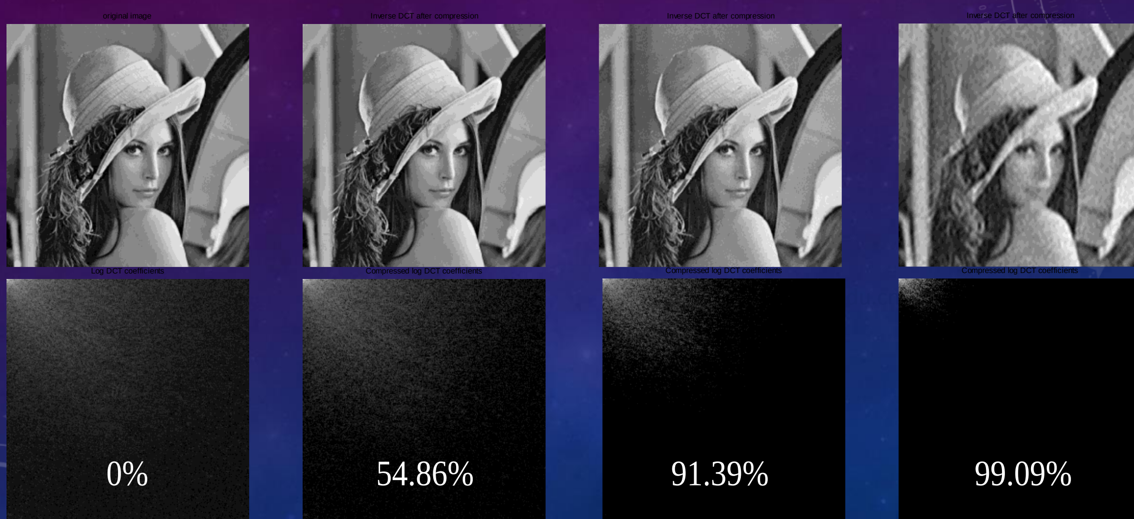


IMAGE TRANSFORMS – DCT2 APPLICATION

- DCT is often used in signal and image processing, especially for lossy compression, because it has a strong "energy compaction" property
- For strongly correlated Markov processes, the DCT can approach the compaction efficiency of the Karhunen-Loève transform (which is optimal in the decorrelation sense)
- The DCT is used in JPEG image compression, MJPEG, MPEG, DV, Daala(video coding format), and Theora video compression. It is applied to each row and column of the block, resulting an 8 × 8 transform coefficient array

IMAGE TRANSFORMS – DWT/DHT

- The Hadamard transform (also known as the Walsh–Hadamard transform, Walsh transform, or Walsh–Fourier transform) is an example of a generalized class of Fourier transforms
- It decomposes an arbitrary input signal into a superposition of Walsh functions.

IMAGE TRANSFORMS – DWT/DHT

$b_2(7) = 1$

$\cdots 00111$

- The DHT of a $N \times N$ digital image :

$$H(u,v) = \frac{1}{N}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)(-1)^{\sum_{i=0}^{n-1}[b_i(x)b_i(u)+b_i(y)b_i(v)]}$$

- IDHT :

$$f(x,y) = \frac{1}{N}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1} H(u,v)(-1)^{\sum_{i=0}^{n-1}[b_i(x)b_i(u)+b_i(y)b_i(v)]}$$

where $N = 2^n$ $b_k(z)$ is value of the kth bit of z in *binary representation*

IMAGE TRANSFORMS – DWT/DHT

- The forward and inverse transformation kernel of DHT are the same
- DHT is separable

$$g(x,u) = \frac{1}{\sqrt{N}}(-1)^{\sum_{i=0}^{n-1} b_i(x)b_i(u)} \quad , \quad N = 2^n$$

$$b_i(0) = 0$$

$$b_i(1) = \begin{cases} 1 & i = 0 \\ 0 & otherwise \end{cases}$$

$$b_i(2) = \begin{cases} 1 & i = 1 \\ 0 & otherwise \end{cases}$$

$$b_i(3) = \begin{cases} 1 & i = 0,1 \\ 0 & otherwise \end{cases}$$

$$H = G \cdot f \cdot G$$

$$f = G \cdot H \cdot G$$

$$G = \frac{1}{2}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}$$

IMAGE TRANSFORMS – DWT/DHT

- The naturally ordered Hadamard matrix can be defined by the recursive formula below

$$\frac{1}{\sqrt{N}} H_N = \frac{1}{\sqrt{N}} \begin{bmatrix} H_{N/2} & H_{N/2} \\ H_{N/2} & -H_{N/2} \end{bmatrix} \qquad H_2 = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

$$H_8 = \frac{1}{2\sqrt{2}} \begin{array}{c} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix} \end{array} \begin{array}{c} 0 \\ 7 \\ 3 \\ 4 \\ 1 \\ 6 \\ 2 \\ 5 \end{array}$$

the number of sign change

IMAGE TRANSFORMS – KARHUNEN-LOÈVE TRANSFORM

- Karhunen-Loève Transform

$$\boldsymbol{C}_x = \frac{1}{L}\sum_{i=1}^{L}(\boldsymbol{X}_i - \boldsymbol{m}_x)(\boldsymbol{X}_i - \boldsymbol{m}_x)^{\mathrm{T}} = \frac{1}{L}\left[\sum_{i=1}^{L}\boldsymbol{X}_i\boldsymbol{X}_i^{\mathrm{T}}\right] - \boldsymbol{m}_x\boldsymbol{m}_x^{\mathrm{T}}$$

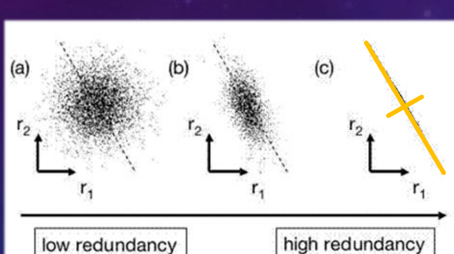
$$C_x \to \{\lambda_i, \vec{u}_i\} \qquad P = [\vec{u}_1, \ldots, \vec{u}_i, \ldots]^T$$

$$Y = P(X - m_x)$$

$$X = P^T Y + m_x$$

PRINCIPAL COMPONENT ANALYSIS

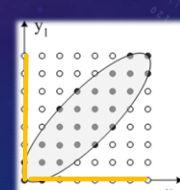
- Arrange the eigenvalues in descending order , the KL approximation is the one that minimizes the total mean square error(KL transform optimally compacts the energy)



$$X = P^T Y + m_x$$

$$\hat{X} = P_K^T Y_K + m_x$$

$$\varepsilon = \sum_{i=K+1}^{N\times N}\lambda_i$$



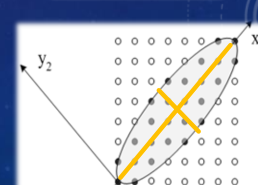


IMAGE TRANSFORMS – KARHUNEN-LOÈVE TRANSFORM

- The importance of the Karhunen–Loève theorem is that it yields the best basis in the sense that it minimizes the total mean squared error
- Pros : Completely decorrelates the original signal , data driven
- Cons : High computational cost, not suitable for data with a non-Gaussian/non-exponential probability distribution

compressed sensing

Short-Time Fourier Transform

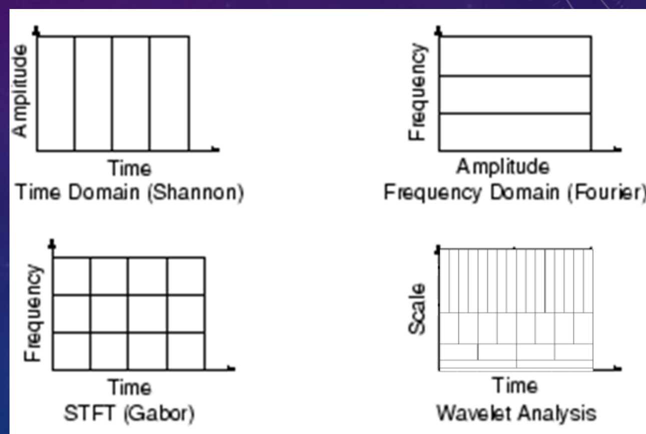
IMAGE TRANSFORMS – DWT

- Wavelets and wavelet transforms are relatively new imaging tools that are being rapidly applied to a wide variety of image processing problems. It is now often replacing the conventional Fourier transform
- Wavelet transforms are broadly divided into three classes: continuous, discrete and multiresolution-based
- Both Fourier and wavelet transforms are frequency-localized, but wavelets have an additional time-localization property.

IMAGE TRANSFORMS – DWT

Time-frequency tilings for the basis functions associated with

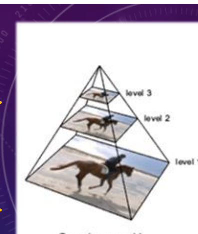
- (a)Sampled data
- (b)Fourier transform
- (c)STFT
- (d)Wavelet transform



mean pyramid

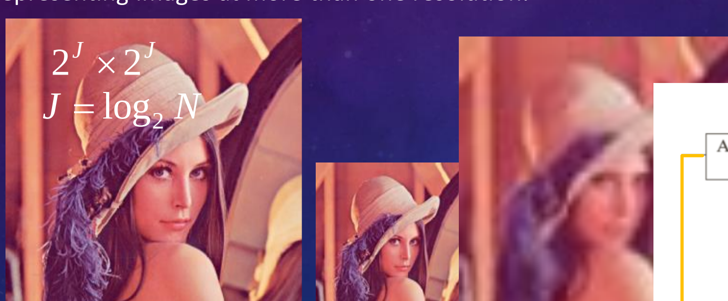
subsampling pyramid

IMAGE TRANSFORMS – DWT



- Image pyramid

A collection of decreasing resolution images arranged in the shape of a pyramid, used for representing images at more than one resolution.



Gaussian pyramid

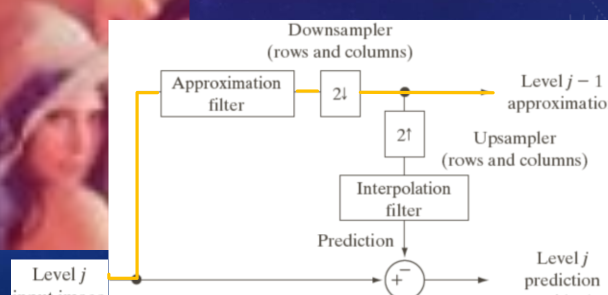
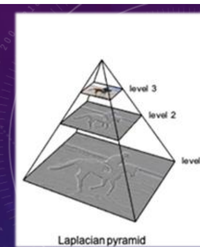


IMAGE TRANSFORMS – DWT



- Image pyramid

A collection of decreasing resolution images arranged in the shape of a pyramid, used for representing images at more than one resolution.

Laplacian pyramid

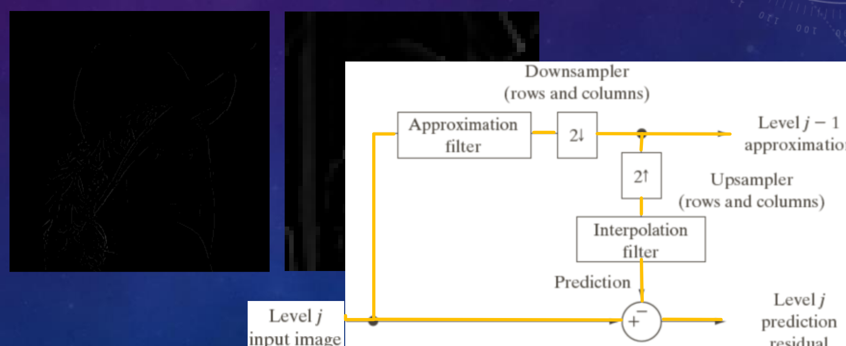
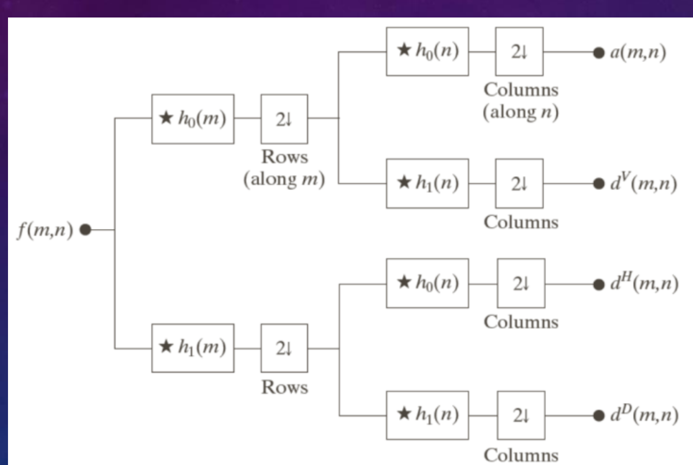


IMAGE TRANSFORMS – DWT



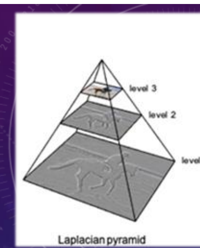
approximation

vertical detail

horizontal detail

diagonal detail

IMAGE TRANSFORMS – DWT



- In mathematics, a wavelet series is a representation of a square-integrable (real or complex-valued) function by a certain orthonormal series generated by a wavelet:

$$\psi_{a,b}(x) = |a|^{-\frac{1}{2}} \psi\left(\frac{x-b}{a}\right)$$

  - $Mother\ wavelet\, \psi(x)$ has a finite-length or fast-decaying oscillating waveform
  - a and b are scaling and translation parameters respectively

SUMMARY

- In some cases, image processing tasks are best formulated in a transform domain
- The transformation kernels are generally orthogonal
- By decorrelating the original image, efficient compression could be achieved
- Different transforms have different basis functions

IMAGE ENHANCEMENT

- Image enhancement is the process of manipulating an image so that the result is more suitable than the original for a specific application. These techniques are problem oriented
- Image enhancement techniques basically are heuristic procedures designed to manipulate an image in order to take advantage of the psychophysical aspects of the human visual system
- Image enhancement is largely a subjective process, while image restoration is for the most part an objective process

IMAGE ENHANCEMENT

- Spatial domain processing

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

  - Spatial filtering
    - E.g. image sharpening, smoothing
  - Intensity transformation
    - E.g. contrast manipulation, thresholding
- Frequency domain processing

$$g(x, y) = F^{-1}\{T[F[f(x, y)]]\}$$

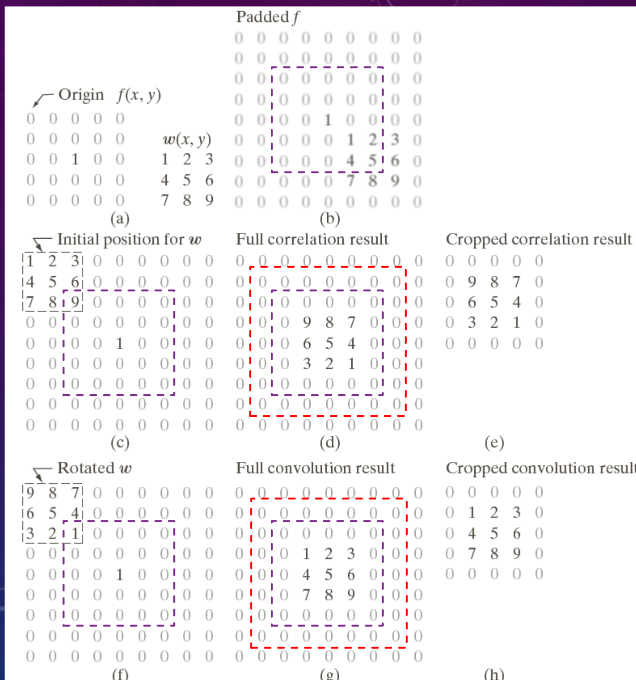


IMAGE ENHANCEMENT

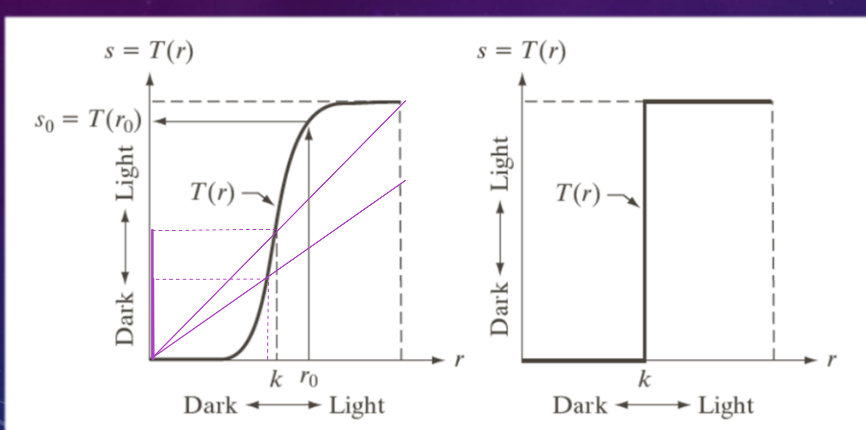
Correlation:

$$g(x,y)=\sum_{s=-a}^{a}\sum_{t=-b}^{b} w(s,t)\,f(x+s,y+t)$$

Convolution:

$$g'(x,y)=\sum_{s=-a}^{a}\sum_{t=-b}^{b} w(s,t)\,f(x-s,y-t)$$

IMAGE ENHANCEMENT



a b

FIGURE 3.2 Intensity transformation functions. (a) Contrast-stretching function. (b) Thresholding function.

HISTOGRAM PROCESSING

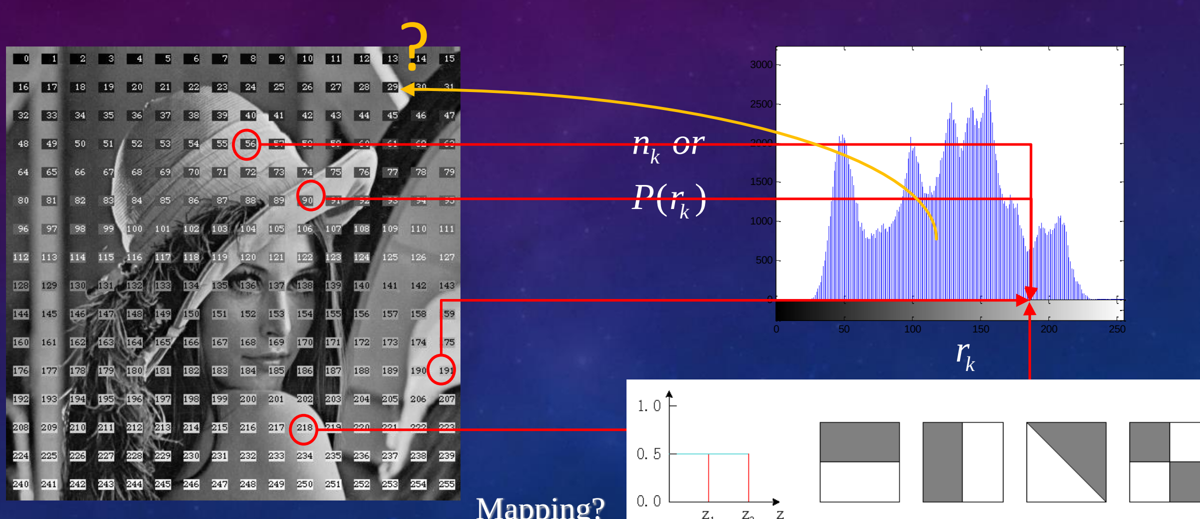
- Histogram
  - Histogram is an estimate of the probability of occurrence of each intensity level in an image
  - A normalized histogram of a $M \times N$ digital image with intensity levels in the range [0, L-1] is given by：

$$P(r_k) = \frac{n_k}{MN}, \quad k = 0,1,\ldots,L-1$$

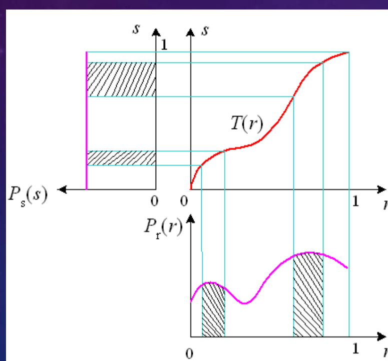
- MN —— total # of pixels
- r_k —— the kth intensity value
- n_k —— # of pixels with intensity r_k

$$P(r_k) = \frac{n_k}{MN}, \quad k = 0,1,\ldots,L-1$$

HISTOGRAM PROCESSING



HISTOGRAM EQUALIZATION

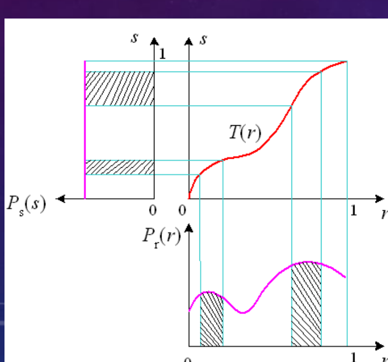


$$p_s(s) = p_r(r)\left|\frac{dr}{ds}\right|$$

$$s = T(r) = \int_0^r p_r(\omega)d\omega$$

cumulative distribution function(CDF)

HISTOGRAM EQUALIZATION

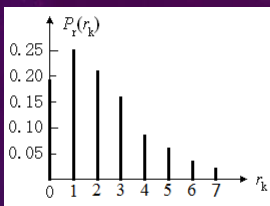


$$\frac{ds}{dr} = \frac{dT(r)}{dr} = \frac{d\left(\int_0^r p_r(\omega)d\omega\right)}{dr}$$

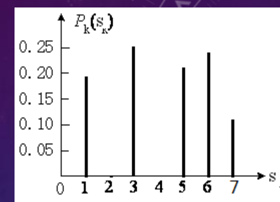
$$= p_r(r) - p_r(0) = p_r(r)$$

$$p_s(s) = p_r(r)\left|\frac{dr}{ds}\right| = \frac{p_r(r)}{p_r(r)} = 1 \qquad 0 \le s \le 1$$

$$s = T(r) = \int_0^r p_r(\omega)d\omega$$

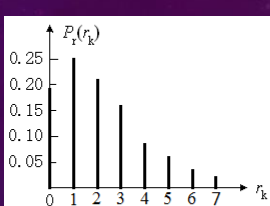


HISTOGRAM EQUALIZATION

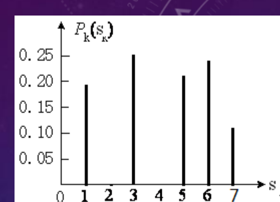


$$s_k = T(r_k) = (L-1)\sum_{j=0}^{k} p_r(r_j) = (L-1)P_r'(k)$$

Operations	Outputs							
Original gray level r_k	0	1	2	3	4	5	6	7
Pixel # of gray level in the ori hist	790	1023	850	656	329	245	122	81
Original hist $P(r_k)$	0.19	0.25	0.21	0.16	0.08	0.06	0.03	0.02
CDF $p(r'k)$	0.19	0.44	0.65	0.81	0.89	0.95	0.98	1.00
Rounding: s_k=int[(L-1)p(r'k)+0.5]	1	3	5	6	6	7	7	7
Mapping relationship ($r_k \rightarrow s_k$)	0→1	1→3	2→5	3, 4→6		5, 6, 7→7		
Pixel # of gray level in the new hist		790		1023		850	985	448
New hist $P(s_k)$		0.19		0.25		0.21	0.24	0.11



HISTOGRAM EQUALIZATION



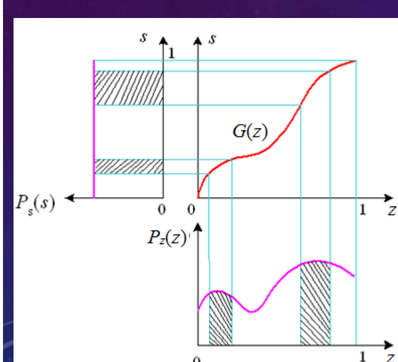
- Discrete histogram equalization has the general tendency to spread the histogram of the input image, resulting a contrast enhancement
- It is fully *automatic*

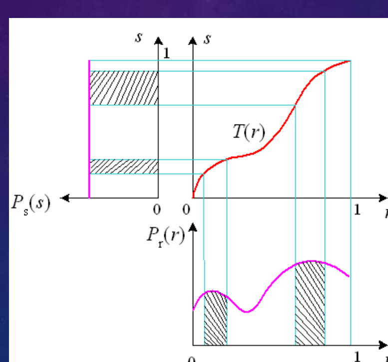
$$s_k = T(r_k) = (L-1)\sum_{j=0}^{k} p_r(r_j)$$

HISTOGRAM MODIFICATION

- Histogram Specification
  - There are applications in which a uniform histogram is not the best choice
  - The method used to generate a processed image that has a specified histogram is called *histogram specification/ histogram matching*

HISTOGRAM SPECIFICATION





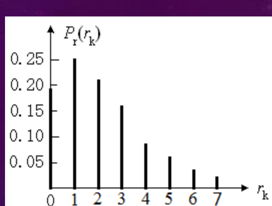
$$s = T(r) = \int_0^r p_r(\omega)d\omega$$

$$s = G(z) = \int_0^z p_r(t)dt$$

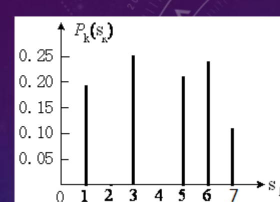
$$z = G^{-1}(s) = G^{-1}\left[T(r)\right]$$

HISTOGRAM SPECIFICATION

- Equalizing the input image
- Obtaining the equalized image in which the pixel values are the s values
- Performing the inverse mapping



HISTOGRAM SPECIFICATION



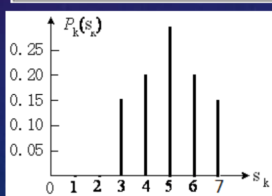
$$s_k = T(r_k) = (L-1)\sum_{j=0}^{k} p_r(r_j) = (L-1)P_r'(k)$$

Operations	Outputs							
Original gray level r_k	0	1	2	3	4	5	6	7
Pixel # of gray level in the ori hist	790	1023	850	656	329	245	122	81
Original hist $P(r_k)$	0.19	0.25	0.21	0.16	0.08	0.06	0.03	0.02
CDF $p(r'k)$	0.19	0.44	0.65	0.81	0.89	0.95	0.98	1.00
Rounding: s_k=int[(L-1)p(r'k)+0.5]	1	3	5	6	6	7	7	7
Mapping relationship ($r_k \to s_k$)	0→1	1→3	2→5	3, 4→6		5, 6, 7→7		
Pixel # of gray level in the new hist		790		1023		850	985	448
New hist $P(s_k)$		0.19		0.25		0.21	0.24	0.11

HISTOGRAM SPECIFICATION

the specified histogram shown in the table

z_q	Specified $p_z(z_q)$
$z_0 = 0$	0.00
$z_1 = 1$	0.00
$z_2 = 2$	0.00
$z_3 = 3$	0.15
$z_4 = 4$	0.20
$z_5 = 5$	0.30
$z_6 = 6$	0.20
$z_7 = 7$	0.15



$$s_k = G(z_q) = (L-1)\sum_{j=0}^{q} p_z(z_j) = (L-1)P_z'(q)$$

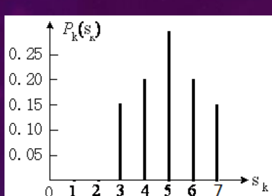
CDF： $P_z'(0) = \sum_{j=0}^{0} p_z(z_j) = 0.00$

$$P_z'(1) = \sum_{j=0}^{1} p_z(z_j) = p_z(z_0) + p_z(z_1) = P_z'(0) + p_z(z_1) = 0.00 + 0.00 = 0.00$$

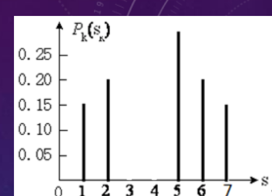
Similarly:

$P_z'(2) = 0.00 \quad P_z'(3) = 0.15 \quad P_z'(4) = 0.35$

$P_z'(5) = 0.65 \quad P_z'(6) = 0.85 \quad P_z'(7) = 1.00$



HISTOGRAM SPECIFICATION

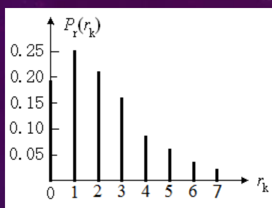


$$s_k = G(z_q) = (L-1)\sum_{j=0}^{q} p_z(z_j) = (L-1)P_z'(q)$$

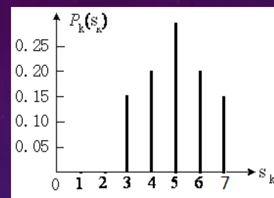
Operations	Outputs							
Specified gray level	0	1	2	3	4	5	6	7
Specified histogram p(zq)	0	0	0	0.15	0.2	0.3	0.2	0.15
CDF p(z'q)	0	0	0	0.15	0.35	0.65	0.85	1
Rounding:sk=INT[(L-1)p(z'q)+0.5]	0	0	0	1	2	5	6	7
Mapping relationship (zq->sk)	0,1,2->0			3->1	4->2	5->5	6->6	7->7
New hist P(sk)		0.15	0.2			0.3	0.2	0.15

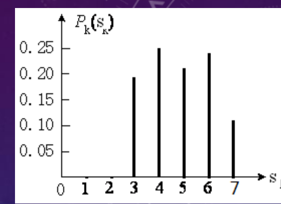
	Mapping							
s	0	1	2	3	4	5	6	7
z	0	3	4	4	5	5	6	7

z_q	$G(z_q)$
$z_0 = 0$	0
$z_1 = 1$	0
$z_2 = 2$	0
$z_3 = 3$	1
$z_4 = 4$	2
$z_5 = 5$	5
$z_6 = 6$	6
$z_7 = 7$	7



HISTOGRAM N





Operations	Outputs							
Original gray level r_k	0	1	2	3	4	5	6	7
Pixel								
Origi								
CDF								
Rounding: s_k=int[(L-1)p(r'k)+0.5]	1	3	5	6	6	7	7	7
Mapping relationship ($r_k \rightarrow s_k$)	0→1	1→3	2→5	3, 4→6		5, 6, 7→7		
Pixel # of gray level in the new hist		790		1023		850	985	448
New hist $P(s_k)$		0.19		0.25		0.21	0.24	0.11

Mapping								
s	0	1	2	3	4	5	6	7
z	0	3	4	4	5	5	6	7

r	s	z
0	1	3
1	3	4
2	5	5
3	6	6
4	6	6
5	7	7
6	7	7
7	7	7

SPATIAL DOMAIN FILTERING

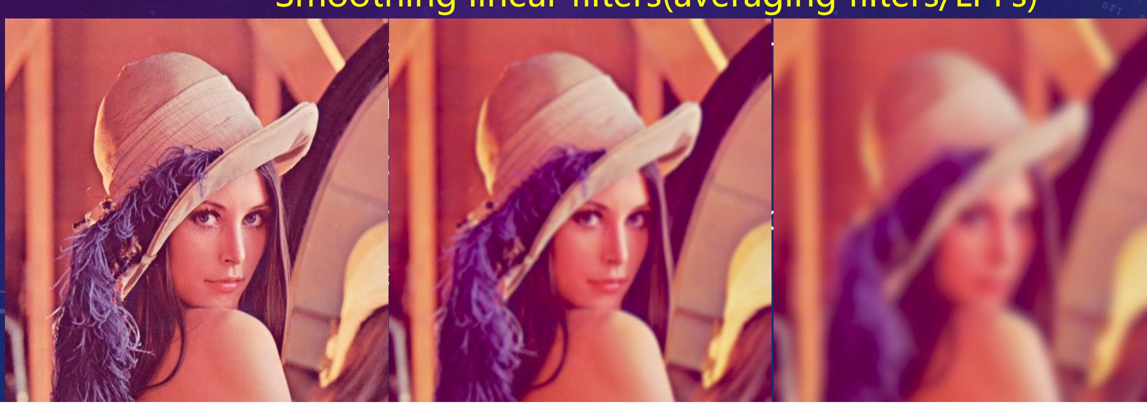
- Smoothing spatial filters
  - Smoothing linear filters
  - Order-statistic(nonlinear) filters
- Sharpening spatial filters
  - The laplacian
  - Unsharp masking and highboost filtering
  - The gradient

IMAGE SMOOTHING

- To smooth a data set is to create an approximating function that attempts to capture important patterns in the data, while leaving out noise or other fine-scale structures/rapid phenomena
- Image smoothing can be done in either spatial domain or frequency domain

SPATIAL DOMAIN FILTERING

- Smoothing spatial filters
  - Smoothing linear filters(averaging filters/LPFs)



SPATIAL DOMAIN FILTERING

$$h_1 = \frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \qquad h_2 = \frac{1}{10}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix} \qquad h_3 = \frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

Box filter

An attempt to reduce blurring in the smoothing process

SPATIAL DOMAIN FILTERING

- Smoothing spatial filters
  - Smoothing linear filters
  - Order-statistic(nonlinear) filters --- median filter
- Sharpening spatial filters
  - The laplacian
  - Unsharp masking and highboost filtering
  - The gradient

SPATIAL DOMAIN FILTERING

- Median filter

For a 1D sequence $f_1, f_2, \ldots, f_n$

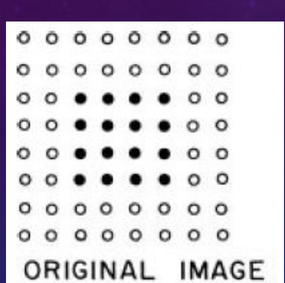
The median filter is of size m (m *is odd*);

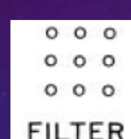
First sort the values of the pixel in the m neighborhood pixels, $f_{i-v}, \ldots, f_{i-1}, f_i, f_{i+1}, \ldots, f_{i+v}$, $v=(m-1)/2$, determine their median ξ , and assign that value to the corresponding pixel in the filtered image:

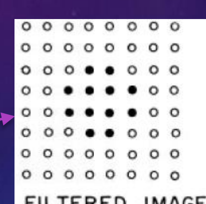
$$Y_i = Med\{f_{i-v}, \ldots, f_i, \ldots, f_{i+v}\} \qquad i \in Z, v = \frac{m-1}{2} \qquad \{0,3,4,0,7\}$$

$$Y_{ij} = \underset{A}{Med}\{X_{ij}\} \qquad A \text{ is filter mask}$$

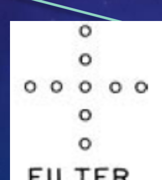
IMAGE SMOOTHING

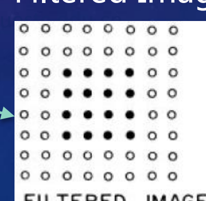






Filtered Image?





Filtered Image?

SPATIAL DOMAIN FILTERING

- Smoothing spatial filters
  - Smoothing linear filters
  - Order-statistic(nonlinear) filters
- Sharpening spatial filters
  - The laplacian
  - Unsharp masking and highboost filtering
  - The gradient

isotropic 各向同性

SPATIAL DOMAIN FILTERING

$$H = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

- The laplacian

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

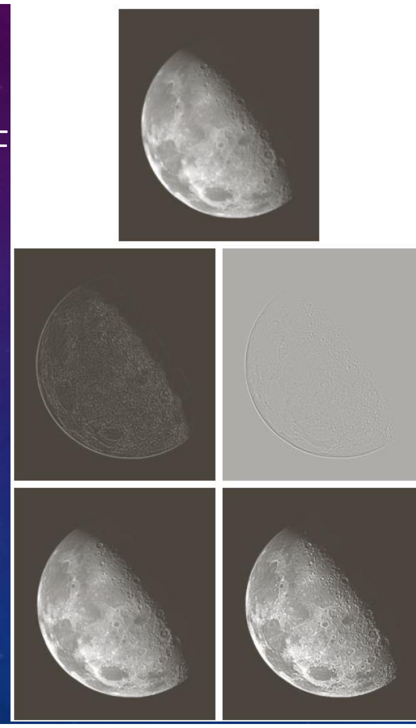
$$H = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

$$\begin{aligned} \nabla^2 f &= \frac{\partial^2 f(x,y)}{\partial x^2} + \frac{\partial^2 f(x,y)}{\partial y^2} \\ &= f(i+1,j) + f(i-1,j) - 2f(i,j) + f(i,j+1) + f(i,j-1) - 2f(i,j) \\ &= f(i+1,j) + f(i-1,j) + f(i,j+1) + f(i,j-1) - 4f(i,j) \end{aligned}$$

SPATIAL DOMAIN F

- The laplacian

$$g = f - k\nabla^2 f$$



a
b c
d e

FIGURE 3.38 (a) Blurred image of the North Pole of the moon. (b) Laplacian without scaling. (c) Laplacian with scaling. (d) Image sharpened using the mask in Fig. 3.37(a). (e) Result of using the mask in Fig. 3.37(b). (Original image courtesy of NASA.)

SPATIAL DOMAIN FILTERING

- Smoothing spatial filters
  - Smoothing linear filters
  - Order-statistic(nonlinear) filters
- Sharpening spatial filters
  - The laplacian
  - Unsharp masking and highboost filtering
  - The gradient

$$H = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

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

SPATIAL DOMAIN FILTERIN

- Unsharp masking and highboost filteri

$$g = f - \bar{f}$$

unsharp mask

$$f' = f + kg$$

Unsharp mask —— 非锐化模板
Unsharp masking —— 锐化补偿
Highboost filtering —— 高频增强

a
b
c
d
e

FIGURE 3.40 (a) Original image. (b) Result of blurring with a Gaussian filter. (c) Unsharp mask. (d) Result of using unsharp masking. (e) Result of using highboost filtering.

SPATIAL DOMAIN FILTERING

- Smoothing spatial filters
  - Smoothing linear filters
  - Order-statistic(nonlinear) filters
- Sharpening spatial filters
  - The laplacian
  - Unsharp masking and highboost filtering
  - The gradient

SPATIAL DOMAIN FILTERING

- The gradient

$$G[f(x,y)] = \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

$$\nabla F = \frac{\partial F}{\partial x}\vec{i} + \frac{\partial F}{\partial y}\vec{j}$$

It points in the direction of the greatest rate of change of f at location (x,y)

magnitude

$$M(x,y) = Mag[f(x,y)] = \left[\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2 \right]^{1/2}$$

FREQUENCY DOMAIN FILTERING

- Ideal LPF:

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

Ringing effect

- Butterworth LPF:

$$H(u,v) = \frac{1}{1 + [D(u,v)/D_0]^{2n}}$$

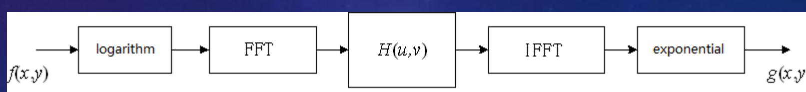
- Gaussian LPF

$$H(u,v) = e^{-\frac{D^2(u,v)}{2D_0^2}}$$

$$H_{HP}(u,v) = 1 - H_{LP}(u,v)$$

HOMOMORPHIC FILTERING

- Homomorphic filtering is a generalized technique for signal and image processing, involving a nonlinear mapping to a different domain in which linear filter techniques are applied, followed by mapping back to the original domain.
- It simultaneously normalizes the brightness across an image and increases contrast



HOMOMORPHIC FILTERING

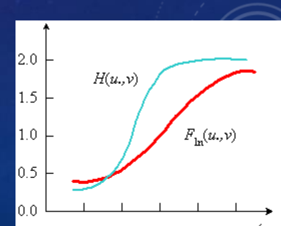
$$\ln f(x,y) = \ln[f_i(x,y) \cdot f_r(x,y)] = \ln f_i(x,y) + \ln f_r(x,y)$$

$$F_{\ln}(u,v) = F[\ln f(x,y)] = F[\ln f_i(x,y) + \ln f_r(x,y)]$$
$$= F_{i,\ln}(u,v) + F_{r,\ln}(u,v)$$

$$G_{\ln}(u,v) = F_{i,\ln}(u,v) \cdot H(u,v) + F_{r,\ln}(u,v) \cdot H(u,v)$$
$$= G_{i,\ln}(u,v) + G_{r,\ln}(u,v)$$

$$F^{-1}[G_{\ln}(u,v)] = \ln g_i(x,y) + \ln g_r(x,y)$$
$$= \ln[g_i(x,y) \cdot g_r(x,y)]$$

$$g(x,y) = \exp\{\ln[g_i(x,y) \cdot g_r(x,y)]\} = g_i(x,y) \cdot g_r(x,y)$$



SUMMARY

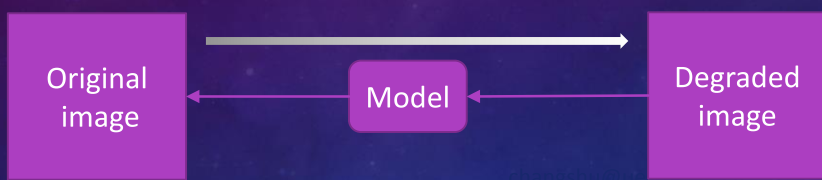
- Spatial domain processing $g(x,y) = T[f(x,y)]$
  - Spatial filtering
    - image sharpening(gradient/laplacian/unsharp masking & highboost filtering), smoothing(averaging/median filter)
  - Intensity transformation
    - Contrast manipulation, thresholding, histogram equalization, histogram specification
- Frequency domain processing $g(x,y) = F^{-1}\{T[F[f(x,y)]]\}$
  - Homomorphic filtering

IMAGE RESTORATION

- Image degradation
- General image restoration models
- Inverse filtering
- Wiener filtering
- Constrained least squares filtering
- Geometric image modification

IMAGE RESTORATION

- Goal



- Evaluation

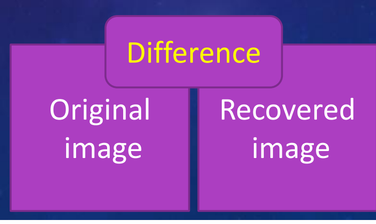
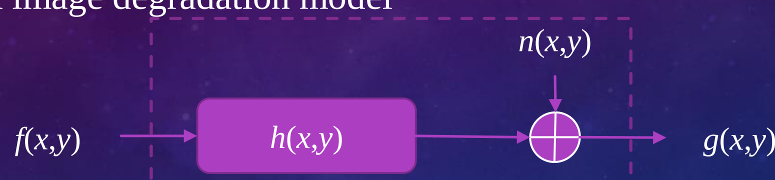


IMAGE RESTORATION

- General image degradation model



- Degraded output image in frequency domain

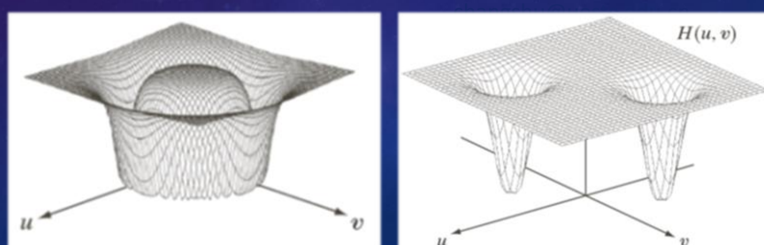
$$G(u,v) = H(u,v)F(u,v) + N(u,v)$$

- Degraded output image in spatial domain

$$g(x,y) = h(x,y) \otimes f(x,y) + n(x,y)$$

RESTORATION IN THE PRESENCE OF NOISE ONLY

- Removal of periodic noise
  - Periodic noise in an image arises typically from electrical or electromechanical interference during image acquisition
  - Periodic noise can be reduced significantly via frequency domain filtering.



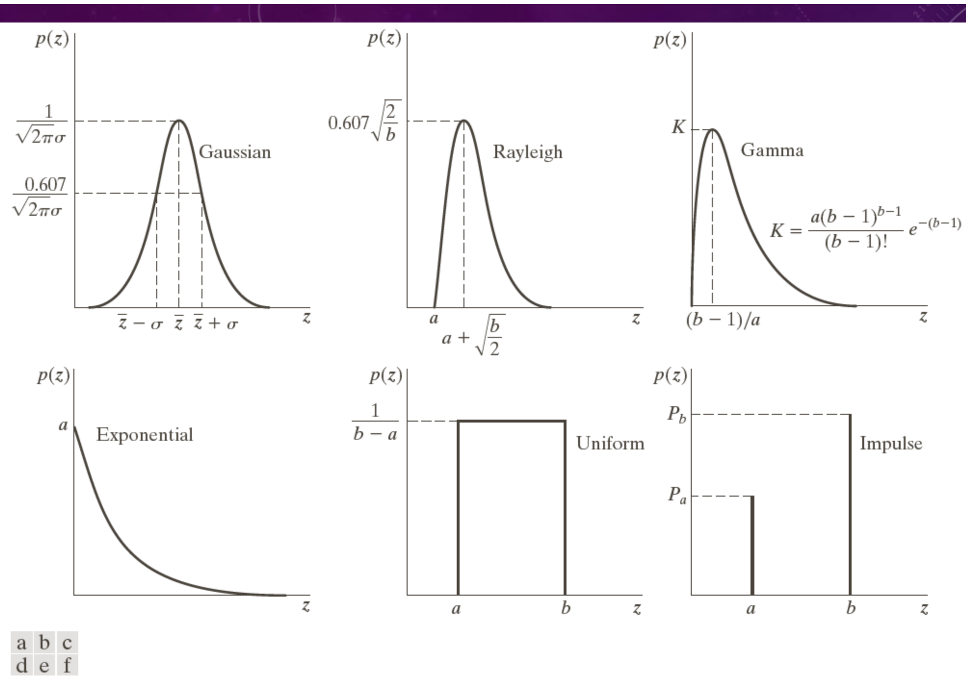


FIGURE 5.2 Some important probability density functions.

RESTORATION IN THE PRESENCE OF NOISE ONLY

$$g(x,y) = f(x,y) + n(x,y) \qquad G(u,v) = F(u,v) + N(u,v)$$

- Removal of periodic noise
  - Periodic noise can be reduced significantly via frequency domain filtering.
  - The parameters of periodic noise can be estimated by inspection of the Fourier spectrum of the image
- Noise PDF estimation
  - The intensity values in the noise component may be considered random variables characterized by a probability density function.
  - The parameters of the PDF can be estimated from small patches of reasonably constant background intensity

IMAGE RESTORATION

- Image degradation
- General image restoration models
- Inverse filtering
- Wiener filtering
- Constrained least squares filtering
- Geometric image modification

INVERSE FILTERING

- Unconstrained image restoration
  - Given the degradation function *H*, we compute an estimate of the transform of the original image by dividing the transform of the degraded image by the degradation function:

If H has 0s or very small values, the ratio could easily dominate the estimated F.

$$\hat{F}(u,v) = \frac{G(u,v)}{H(u,v)}$$

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

One approach to get around the 0 or small value problem is to limit the filter frequencies to values near the origin. Thus the probability of encountering 0 values will be reduced

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

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

a b
c d

FIGURE 5.27 Restoring Fig. 5.25(b) with Eq. (5.7-1). (a) Result of using the full filter. (b) Result with *H* cut off outside a radius of 40; (c) outside a radius of 70; and (d) outside a radius of 85.

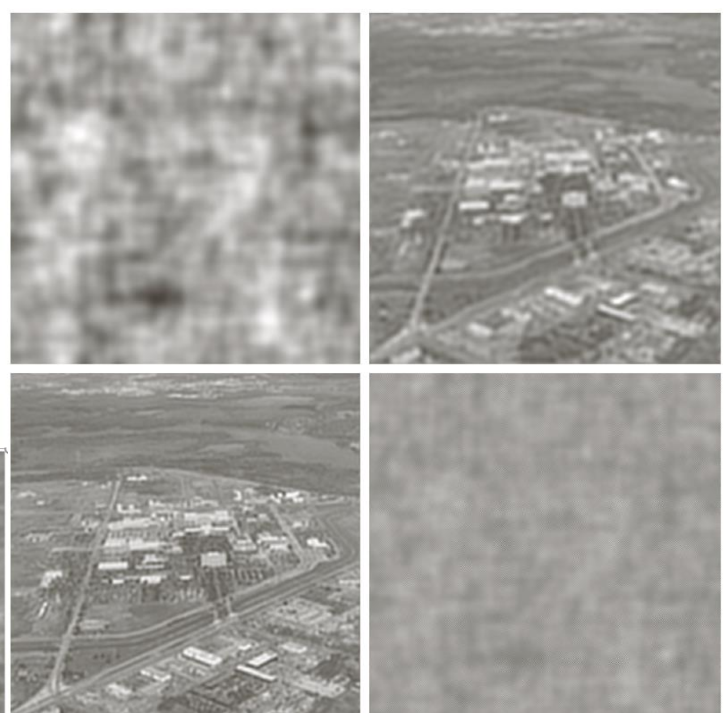


IMAGE RESTORATION

- Image degradation
- General image restoration models
- Inverse filtering
- Wiener filtering/minimum mean square error filtering
- Constrained least squares filtering
- Geometric image modification

WIENER FILTERING

- Error measure :

$$e^2 = E\left[(f - \hat{f})^2 \right]$$

- Solution :

$$\hat{F}(u,v) = \left[\frac{H^*(u,v)}{|H(u,v)|^2 + \gamma[S_n(u,v)/S_f(u,v)]} \right] G(u,v)$$

$$= \left[\frac{1}{H(u,v)} \cdot \frac{|H(u,v)|^2}{|H(u,v)|^2 + \gamma[S_n(u,v)/S_f(u,v)]} \right] G(u,v)$$

power spectrum of the noise

power spectrum of the undegraded image

WIENER FILTERING

$$\hat{F}(u,v)=\left[\frac{1}{H(u,v)}\cdot\frac{|H(u,v)|^2}{|H(u,v)|^2+\gamma[S_n(u,v)/S_f(u,v)]}\right]G(u,v)$$

$$\Downarrow$$

$$\hat{F}(u,v)=\left[\frac{1}{H(u,v)}\cdot\frac{|H(u,v)|^2}{|H(u,v)|^2+K}\right]G(u,v)$$

R FILTERING

a b c

FIGURE 5.28 Comparison of inverse and Wiener filtering. (a) Result of full inverse filtering of Fig. 5.25(b). (b) Radially limited inverse filter result. (c) Wiener filter result.

IMAGE RESTORATION

- Image degradation
- General image restoration models
- Inverse filtering
- Wiener filtering
- Constrained least squares filtering
- Geometric image modification

IMAGE RESTORATION

- Solution : $\hat{f}_e = \left([\hbar_e]^T[\hbar_e] + \alpha [C_e]^T [C_e]\right)^{-1} [\hbar_e]^T g_e$
- Diagonalize by DFT $[\hbar_e]\vec{W}_k = \lambda_h(k)\vec{W}_k$ $\quad [W]^{-1}[h_e][W] = [\Lambda_h]$

$$[W]^* \hat{f}_e = F_e \qquad [W]^* g_e = G_e$$

- Diagonalize the circulant matrix : $E = W^{-1} C W$

$$W^{-1}\hat{f}_e = \left(D^* D + \alpha E^* E\right)^{-1} D^* W^{-1} g_e$$

$$\hat{F}(u,v) = \left[\frac{H^*(u,v)}{|H(u,v)|^2 + \gamma \left|p_e(u,v)\right|^2}\right] G(u,v)$$

(1,1) image corrupted by motion blur and additive noise. (1,2) IF. (1,3) WF. (1,4) CLSF.

(2,1:4) same sequences, but with noise variance one order of magnitude less.

(3,1:4) same sequences, but noise variance reduced by five orders of magnitude from (1,1).

IMAGE RESTORATION

- Constrained least square filtering

$$\hat{\boldsymbol{F}} = \frac{\boldsymbol{G}}{\boldsymbol{H}} \qquad \hat{F} = \left[\frac{H^*}{|H|^2 + \gamma[S_n / S_f]}\right]G \qquad \hat{F} = \left[\frac{H^*}{|H|^2 + \gamma|P|^2]}\right]G$$

1. Observation
2. Experimentation
3. Mathematical modeling

$$\boldsymbol{H}_s = \frac{\boldsymbol{G}_s}{\hat{\boldsymbol{F}}_s}$$

SUMMARY

- Image degradation
- General image restoration models
- Inverse filtering
- Wiener filtering
- Constrained least squares filtering
- Geometric image modification

IMAGE SEGMENTATION

- Goal
- Connectivity (n-adjacency)
- Segmentation Methods
  - Amplitude
  - Region
  - Clustering
- Edge Detection

IMAGE SEGMENTATION

4-adjacency

- A pixel p at coordinates (x, y) has four horizontal and vertical neighbors whose coordinates are given by

$$(x+1, y), (x-1, y), (x, y+1), (x, y-1)$$

  - This set of pixels, called the 4-neighbors of p, is denoted by $N_4(p)$
  - If q is in the set $N_4(p)$, p and q are 4-adjacent.

8-adjacency

  - Four diagonal neighbors of p are denoted by $N_D(p)$. These points together with the 4-neighbors are called the 8-neighbors of p and denoted by $N_8(p)$
  - If q is in the set $N_8(p)$, p and q are 8-adjacent

1	1	1
1	1	1
1	1	1

1	1	1
1	1	1
1	1	1

1	1	1
1	1	1
1	1	1

IMAGE SEGMENTATION

m-adjacency

- Two pixels p and q are m-adjacent if
  - q is in $N_4(p)$, or
  - q is in $N_D(p)$, and the set $N_4(p) \cap N_4(q)$ has no pixels whose values are the values used to define adjacency.

1	1	1
1	1	1
1	1	1

1	0	1
0	1	0
1	0	1

1	0	1
0	1	0
1	1	1

IMAGE SEGMENTATION

- A path from pixel (x_0,y_0) to pixel (x_n,y_n) is a sequence of distinct pixels (x_0,y_0), (x_1,y_1)… (x_i,y_i)… (x_n,y_n), where (x_{i-1},y_{i-1}) and (x_i,y_i) are adjacent for $1 \le i \le n$.
- We can define 4-, 8-, or *m*-paths depending on the type of adjacency specified.
- Question: find the shortest path from p to q, and determine the length of the path.

Using 4-adjacency:

Using 8-adjacency: $n = 4$

Using m-adjacency: $n = 7$

p	1	1	0	0
0	1	1	0	0
0	0	1	0	1
0	1	0	1	0
0	0	0	1	q

IMAGE SEGMENTATION

- The foreground and background must have different definition of connectivity, or there will be ambiguity

0	0	0	0	0
0	1	1	0	0
0	1	0	1	0
0	1	1	1	0
0	0	0	0	0

IMAGE SEGMENTATION

S --- a subset of pixels in an image;.

- Two pixels p and q are said to be connected in S if there exists a path between them consisting entirely of pixels in S.
- S is a connected set if it only has one connected component, it is also called a region of the image.
- Two regions R_i and R_j are said to be adjacent if $R_i \cup R_j$ is a connected set. Regions that are not adjacent are said to be disjoint.
- We consider 4- and 8-adjacency when referring to regions.

IMAGE SE ATION

```
BW = [1 1 1 0 0 0 0 0;
      1 1 1 0 1 1 0 0;
      1 1 1 0 1 1 0 0;
      1 1 1 0 0 0 1 0;
      1 1 1 0 0 0 1 0;
      1 1 1 0 0 0 1 0;
      1 1 1 0 0 1 1 0;
      1 1 1 0 0 0 0 0]
```

```
L4 = 1 1 1 0 0 0 0 0
     1 1 1 0 2 2 0 0
     1 1 1 0 2 2 0 0
     1 1 1 0 0 0 3 0
     1 1 1 0 0 0 3 0
     1 1 1 0 0 0 3 0
     1 1 1 0 0 3 3 0
     1 1 1 0 0 0 0 0
```

L4 = bwlabel(BW,4)

```
L8 =1 1 1 0 0 0 0 0
    1 1 1 0 2 2 0 0
    1 1 1 0 2 2 0 0
    1 1 1 0 0 0 2 0
    1 1 1 0 0 0 2 0
    1 1 1 0 0 0 2 0
    1 1 1 0 0 2 2 0
    1 1 1 0 0 0 0 0
```

L8 = bwlabel(BW,8)

IMAGE SEGMENTATION

- Goal
- Connectivity
- Segmentation Methods
  - Amplitude thresholding
  - Region
  - Clustering
- Edge Detection

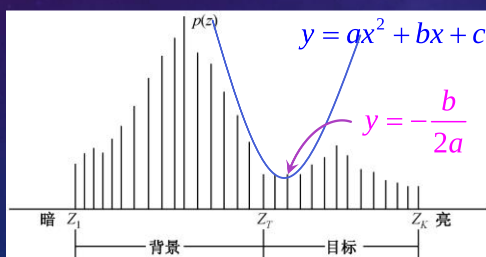
IMAGE SEGMENTATION

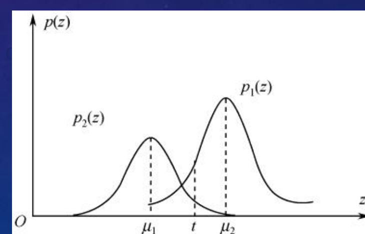
- Bilevel Luminance Thresholding
- If an image is composed of light objects on a dark background, the intensity values can be grouped into two dominant modes. We can fist select a threshold *T* that separates these modes.
- The segmented image $g(x, y)$, is given by：

$$g(x,y) = \begin{cases} 1 & f(x,y) \geq T \\ 0 & f(x,y) < T \end{cases}$$

IMAGE SEGMENTATION

- Basic global thresholding
  - A single threshold applicable over the entire image





Ostu's method

IMAGE SEGMENTATION

- Global thresholding v.s. Variable thresholding
  - Noise and nonuniform illumination play a major role in the performance of a thresholding algorithm
- Variable thresholding
  - Image partitioning
  - Moving averages

$$T_{xy} = bm_{xy}$$

$$T_{xy} = a\sigma_{xy} + bm_{xy}$$

$$T_{xy} = a\sigma_{xy} + bm_G$$

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

IMAGE SEGMENTATION

- Goal
- Connectivity
- Segmentation Methods
    - Amplitude
    - Region
    - Clustering
- Edge Detection

IMAGE SEGMENTATION

- Region-growing
    - Seeds
    - Similar criteria -→ connectivity properties
    - Stopping rule

IMAGE SEGMENTATION

1	0	4	7	5
1	0	4	7	7
0	*1*	5	*5*	5
2	0	5	6	5
2	2	5	6	4

1	1	5	5	5
1	1	5	5	5
1	1	5	5	5
1	1	5	5	5
1	1	5	5	5

1	1	5	7	3
1	1	5	7	7
1	1	5	5	5
2	1	5	5	5
2	2	5	5	5

(a)Seeds on the original image (b)Result of applying the similar criteria T<=3 (c) b)Result of applying the similar criteria T<=1

IMAGE SEGMENTATION

- Goal
- Connectivity
- Segmentation Methods
  - Amplitude
  - Region
  - Clustering
- Edge Detection

IMAGE SEGMENTATION

- A vector $\mathbf{v}_{ij} = [x, \ y, \ z, \ \ldots]^T$ of measurements at each pixel coordinate (i, j) in an image.
- The measurement could be point gray values, point color components, derived color components, etc.
- They could also be neighborhood feature measurements such as moving window mean, standard deviation, mode, etc.

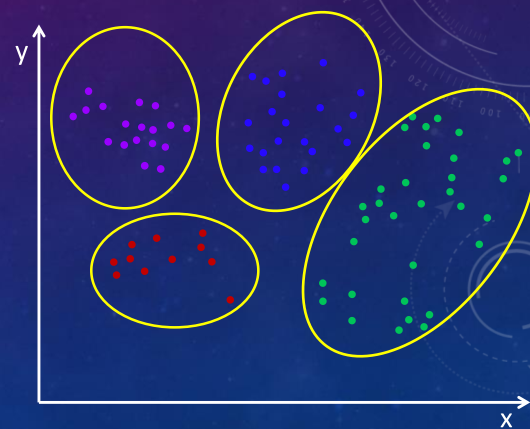


IMAGE SEGMENTATION

- Goal
- Connectivity
- Segmentation Methods
  - Amplitude
  - Region
  - Clustering
- Edge Detection

IMAGE SEGMENTATION

- Roberts

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

- Prewitt

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

- Sobel

$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \qquad \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

IMAGE SEGMENTATION

- Hough Transform
  - Point – line duality between xy-plane and ab-plane ;
  - The principal lines in image plane could be found by identifying points in parameter space where large numbers of parameter-space lines intersect

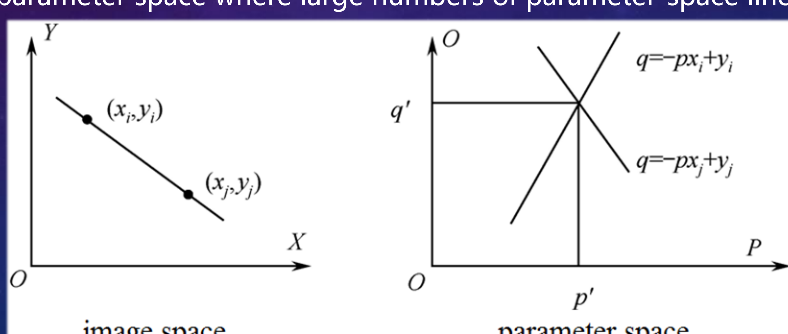
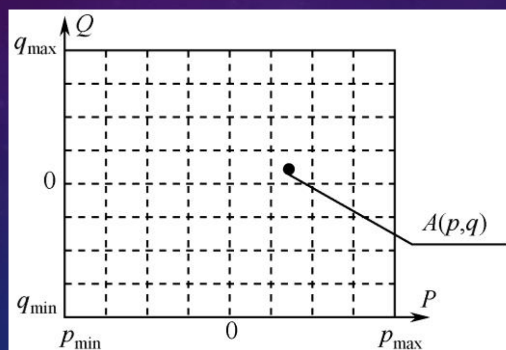


IMAGE SEGMENTATION



Accumulator cells in parameter space

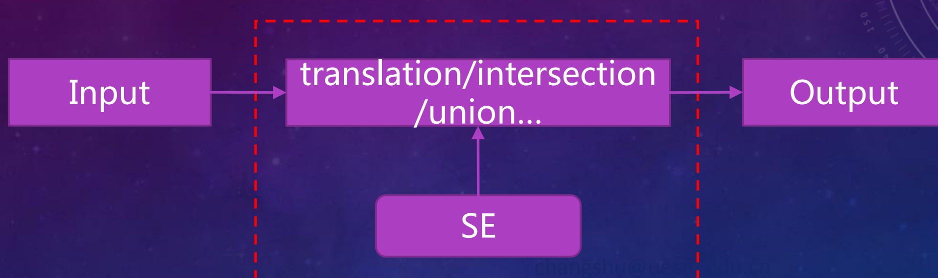
IMAGE SEGMENTATION

- Goal
- Connectivity
- Segmentation Methods
  - Amplitude
  - Region
  - Clustering
- Edge Detection
  - Gradient based methods
  - Hough Transform
  - Snakes/Energy Minimizing Splines
  - Graph Cut Techniques

MORPHOLOGICAL IMAGE PROCESSING

- Overview
- Preliminaries
- Binary morphology
  - Erosion and dilation
  - Opening and closing
  - The Hit-or-Miss transformation
  - Basic morphological algorithms
- Gray-scale morphology

MORPHOLOGICAL IMAGE PROCESSING



- SE v.s. Filter Mask，Logical operation v.s. Convolution(multiplication & addition)

nonlinear linear

MORPHOLOGICAL IMAGE PROCESSING

- translation :

$$B_z = \left\{ \vec{x} \mid \vec{x} = \vec{b} + \vec{z},\ for\ \vec{b} \in B \right\}$$

$$\vec{z} = (z_1, z_2)$$

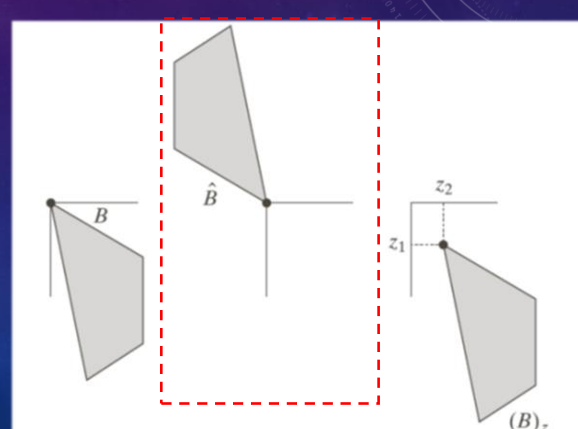
$$(x, y) \quad -> \quad (x + z_1, y + z_2)$$

MORPHOLOGICAL IMAGE PROCESSING

- reflection :

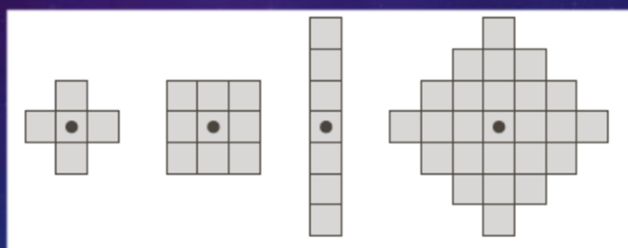
$$\hat{B} = \left\{ \vec{w} \mid \vec{w} = -\vec{b},\ for\ \vec{b} \in B \right\}$$

$$(x, y) \quad -> \quad (-x\ , -y)$$



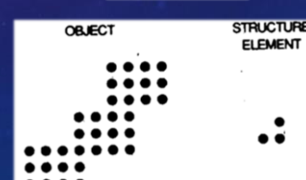
MORPHOLOGICAL IMAGE PROCESSING

- Origin of SE :



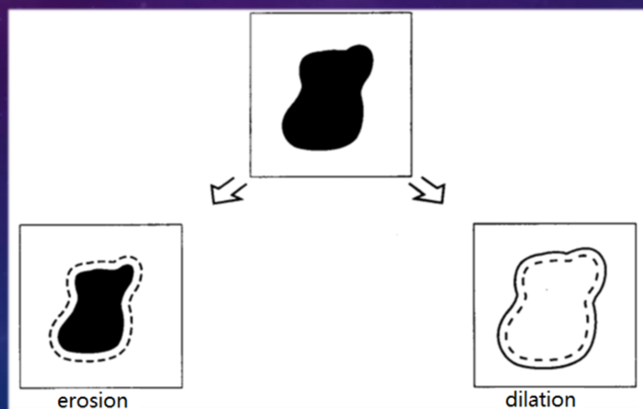
MORPHOLOGICAL IMAGE PROCESSING

- Overview
- Preliminaries
- Binary morphology
  - Erosion and dilation
  - Opening and closing
  - The Hit-or-Miss transformation
  - Basic morphological algorithms
- Gray-scale morphology



MORPHOLOGICAL IMAGE PROCESSING

- Erosion and dilation



MORPHOLOGICAL IMAGE PROCESSING

- Erosion :

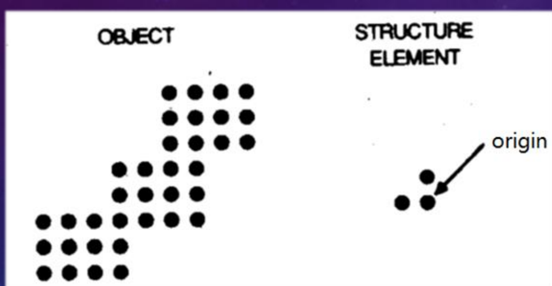
$$E = A \Theta B = \{\vec{z} \mid B_z \subseteq A\}$$

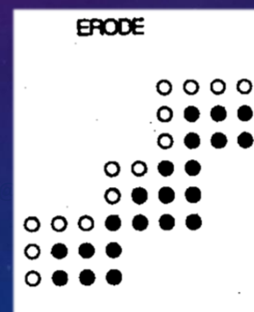
Equivalently,

$$A \Theta B = \{\vec{z} \mid B_z \cap A^C = \varnothing\}$$

- Erosion shrinks or thins objects in a binary image. Image details smaller than the SE are filtered from the image.

MORPHOLOGICAL IMAGE PROCESSING





MORPHOLOGICAL IMAGE PROCESSING

- Dilation :

$$D = A \oplus B = \left\{ \vec{z} \mid \hat{B}_z \cap A \neq \varnothing \right\}$$

Equivalently,

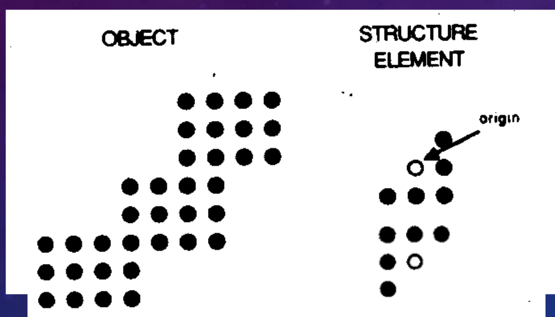
$$A \oplus B = \left\{ \vec{z} \mid (\hat{B}_z \cap A) \subseteq A \right\}$$

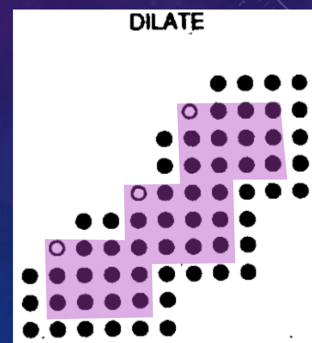
- Dilation grows or thickens objects in a binary image. Gaps narrower than the SE are bridged.

MORPHOLOGICAL IMAGE PROCESSING

$$D = A \oplus B = \left\{ \vec{z} \mid \hat{B}_z \cap A \neq \emptyset \right\}$$

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



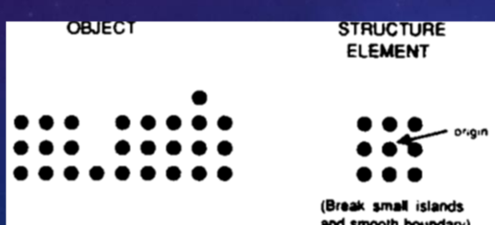


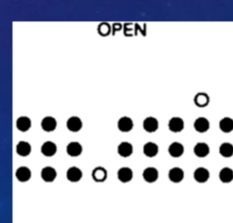
MORPHOLOGICAL IMAGE PROCESSING

- Opening :

$$A \circ B = (A \Theta B) \oplus B$$

- Opening smoothes the contour of an object, breaks narrow isthmuses, and eliminates thin protrusions



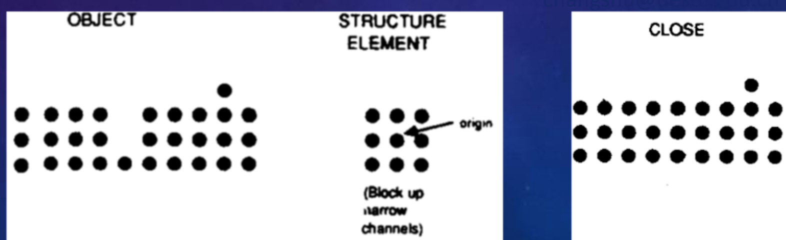


MORPHOLOGICAL IMAGE PROCESSING

- Closing :

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

- Closing also tends to smooth sections of contours, fuses narrow breaks and long thin gulfs, eliminates small holes and fills gaps in the contour



MORPHOLOGICAL IMAGE PROCESSING

- Property of opening :

1 $A \circ B$ is a subset of A

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

3 $A \circ B \circ B = A \circ B$

- Property of closing :

1 A is a subset of $A \bullet B$

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

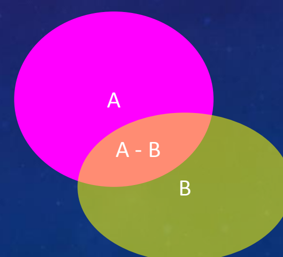
3 $A \bullet B \bullet B = A \bullet B$

MORPHOLOGICAL IMAGE PROCESSING

- Question： Can you extract the boundary of the input binary object by means of erosion/dilation/opening/closing?

- Difference:

$$A - B = \{\vec{w} \in A, \vec{w} \notin B\} = A \cap B^C$$



MORPHOLOGICAL IMAGE PROCESSING

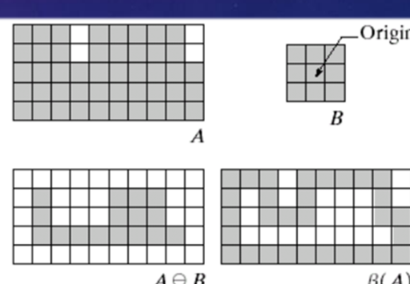
- Boundary extraction：

$$Edge(A) = A - (A \Theta B)$$

$$Edge(A) = (A \oplus B) - A$$

a b
c d

FIGURE 9.13 (a) Set A. (b) Structuring element B. (c) A eroded by B. (d) Boundary, given by the set difference between A and its erosion.



MORPHOLOGICAL IMAGE PROCESSING

- Commutative Property $A \oplus B = B \oplus A$
- Associative Property
$$(A \oplus B_1) \oplus B_2 = A \oplus (B_1 \oplus B_2)$$
$$(A \Theta B_1) \Theta B_2 = A \Theta (B_1 \oplus B_2)$$
- Distributive Property
$$(A_1 \cap A_2) \Theta B = (A_1 \Theta B) \cap (A_2 \Theta B)$$
$$A \Theta (B_1 \cup B_2) = (A \Theta B_1) \cap (A \Theta B_2)$$
- Duality
$$A^C \oplus \hat{B} = (A \Theta B)^C$$
$$A^C \Theta \hat{B} = (A \oplus B)^C$$

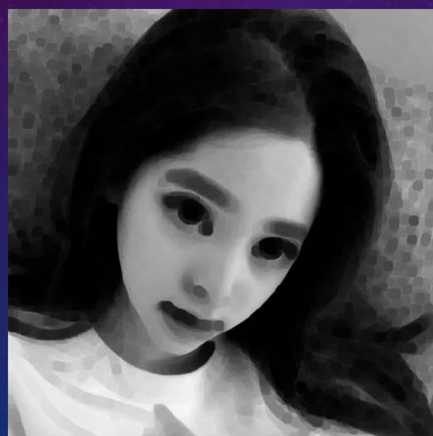
MORPHOLOGICAL IMAGE PROCESSING

- SE decomposition
  - Decompose complex SE into simple SEs according to:
$$A \Theta (B_1 \oplus B_2) = (A \Theta B_1) \Theta B_2$$
  - E.g. Hexagonal element SE

Original

MORPHOLOGICAL IMAGE PROCESSING

Erode



Dilate

MORPHOLOGICAL IMAGE PROCESSING

- Opening & closing :

$$A \circ B = (A \Theta B) \oplus B$$

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

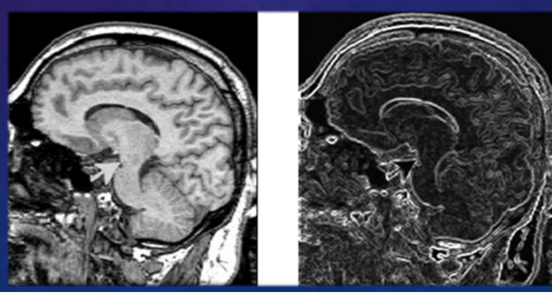
- The expressions have the same form as their binary counterparts
- Opening is used to remove small, bright details, while closing is used to attenuate dark features

MORPHOLOGICAL IMAGE PROCESSING

- Morphological gradient :

$$Grad(f) = (f \oplus g) - (f \Theta g)$$

  - Derivative-like (gradient) effect



TOP HAT TRANSFORM

- f minus its opening:

$$THT(f) = f - (f \circ g)$$

g is a structuring element

- THT is used for light objects on a dark background

BOTTOM HAT TRANSFORM

- The closing of f minus f:

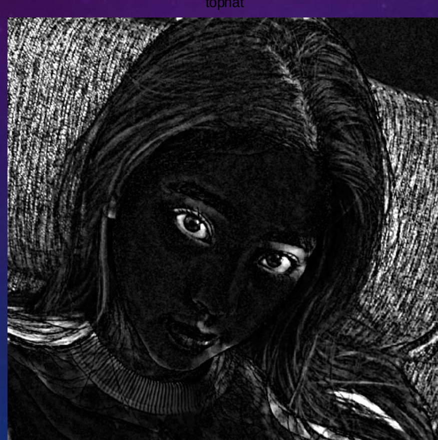
$$BHT(f) = (f \bullet g) - f$$

g *is a structuring element*

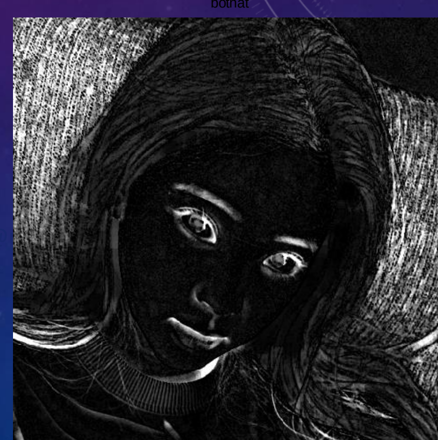
- BHT is used for dark objects on a light background

AND BOTTOM-HAT TRANSFORMATIONS

tophat



bothat



SUMMARY

- Overview
- Preliminaries (translation/reflection)
- Binary morphology
  - Erosion and dilation, Opening and closing, The Hit-or-Miss transformation, Basic morphological algorithms(thinning/thickening/skeletonizing/shrink/...)
- Gray-scale morphology
  - Erosion and dilation, Opening and closing, Morphological gradient, Top-hat and bottom-hat transformations

IMAGE COMPRESSION

- fundamentals
  - Entropy of a single event $I(E) = \log \frac{1}{P(E)} = -\log P(E)$
  - Entropy of a zero-memory source $H = -\sum_i P(a_i) \log P(a_i)$
  - Entropy of a zero-memory intensity source(image) $\tilde{H} = -\sum_{k=0}^{L-1} P_r(r_k) \log_2 P_r(r_k)$
  - Shannon's first theorem $H = \lim_{n \to \infty} \frac{L_{\text{avg},n}}{n}$
  - Entropy of Markov(finite memory) source

$$H_N = -\sum_{a_0} \sum_{a_1} \cdots \sum_{a_N} P(a_0, a_1, \cdots, a_{N-1}) \log_2 P(a_0 | a_1, \cdots, a_{N-1})$$

$$H_0 \geq H_1 \geq \cdots \geq H_{N-1} \geq H_N = H_{N+1} = H_{N+2} = \cdots = H_\infty$$

IMAGE COMPRESSION

- fundamentals
  - Entropy $\tilde{H} = -\sum_{k=0}^{L-1} P_r(r_k)\log_2 P_r(r_k)$
  - average code length $R(x) = \sum_i P(x_i)L_i$
  - Coding efficiency $\frac{H(x)}{R(x)}$
  - Compression ratio $c = \frac{b_1}{b_2}$
  - SNR $10\log\frac{\sigma_x^2}{\sigma_e^2}$

IMAGE COMPRESSION

- Image data redundancy
  - Coding redundancy
  - Spatial and temporal redundancy
  - Irrelevant information

$$R(x) = \sum_i P(x_i)L_i$$

IMAGE COMPRESSION

r_k	$p_r(r_k)$	**Code 1**	$l_1(r_k)$	**Code 2**	$l_2(r_k)$
$r_{87} = 87$	0.25	01010111	8	01	2
$r_{128} = 128$	0.47	10000000	8	1	1
$r_{186} = 186$	0.25	11000100	8	000	3
$r_{255} = 255$	0.03	11111111	8	001	3
r_k for $k \neq 87, 128, 186, 255$	0	—	8	—	0

IMAGE COMPRESSION

- Image data redundancy
  - Coding redundancy
  - Spatial and temporal redundancy
  - Irrelevant information

0.631	0.642	0.641	0.628	0.288	0.325	0.375	0.419
0.700	0.716	0.721	0.710	0.435	0.493	0.568	0.630
0.711	0.811	0.842	0.845	0.558	0.635	0.742	0.835

IMAGE COMPRESSION

- Image data redundancy
  - Coding redundancy
  - Spatial and temporal redundancy
  - Irrelevant information

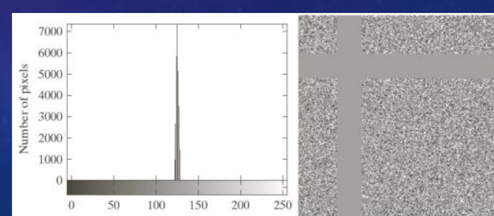


IMAGE COMPRESSION

- Image compression models

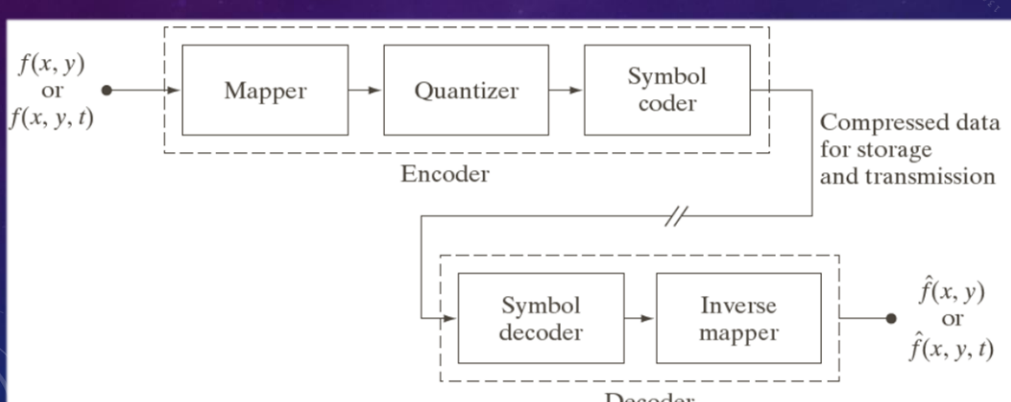


IMAGE COMPRESSION

Huffman coding(Huffman [1952]) yields the smallest possible number of code symbols per source symbol, when coding the symbols of an information source individually. The resulting code is optimal for a fixed value of n, subject to the constraint that the source symbols be coded one at a time.

IMAGE COMPRESSION

Huffman's procedure:

(1) Create a series of source reductions by ordering the probabilities of the symbols under consideration and combining the lowest probability symbols into a single symbol that replaces them in the next source reduction

(2) Code each reduced source, starting with the smallest source and working back to the original source.

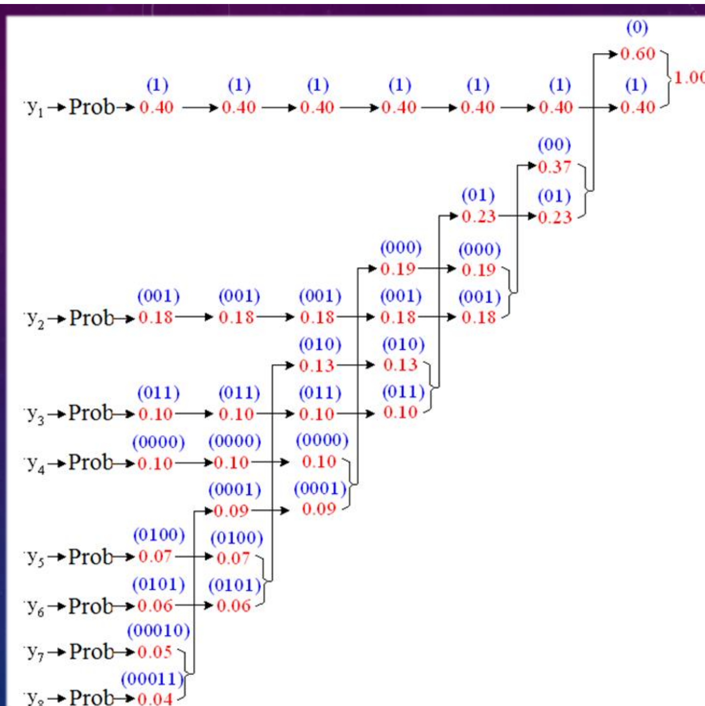


IMAGE COMPRESSION

Entropy of the intensity source：

$$H = -\sum_{i=1}^{L} P_i \log_2 P_i = -(0.4\times\log_2 0.4 + 0.18\times\log_2 0.18 + 2\times 0.1\times\log_2 0.1 +$$
$$0.07\times\log_2 0.07 + 0.06\times\log_2 0.06 + 0.05\times\log_2 0.05 + 0.04\times\log_2 0.04) = 2.55$$

Average code length：

$$R = \sum_{i=1}^{L} P(a_i)\cdot l_i = 0.40\times 1 + 0.18\times 3 + 0.10\times 3 + 0.10\times 4$$
$$+ 0.07\times 4 + 0.06\times 4 + 0.05\times 5 + 0.04\times 5 = 2.61$$

Compression ratio： $C = 3/2.61 = 1.15$

Coding efficiency: $\eta = H/L = 2.55/2.61 = 97.8\%$

IMAGE COMPRESSION

❖ Question : Consider a zero-memory source with 6 symbols {a1, a2, a3, a4, a5 ,a6} , and the source symbol probabilies are {0.1, 0.4, 0.06, 0.1, 0.04, 0.3}

(1) Compute the entropy of the source

(2) Compress the source using Huffman coding

(3) Compute the average code length , compression ratio and coding efficiency

$$H = -\sum_{i=1}^{L} P_i \log_2 P_i = -(0.4\times\log_2 0.4 + 0.3\times\log_2 0.3 + 2\times 0.1\times\log_2 0.1$$
$$+ 0.06\times\log_2 0.06 + 0.04\times\log_2 0.04) = 2.1435$$
$$R = \sum_{i=1}^{L} P(a_i)\cdot l_i = P\bullet[124444]^T = P\bullet[123455]^T = 2.2$$

IMAGE COMPRESSION

- Huffman code is an instantaneous uniquely decodable block code
  - block code --- each source symbol is mapped into a fixed sequence of code symbols
  - instantaneous --- each code word in a string of code symbols can be decoded without referencing succeeding symbols
  - uniquely decodable --- any string of code symbols can be decoded in only one way 010100111011 : $y_6 y_2 y_1 y_1 y_3$

IMAGE COMPRESSION

- Arithmetic coding :

An entire sequence of source symbols is assigned a single arithmetic code word. The code word itself defines an interval of real numbers between 0 and 1.

As the number of symbols in the message increases, the interval used to represent it becomes smaller and the number of information units required to represent the interval becomes larger.

IMAGE COMPRESSION

[0,1) [0.1 0.4) [0.1 0.13)

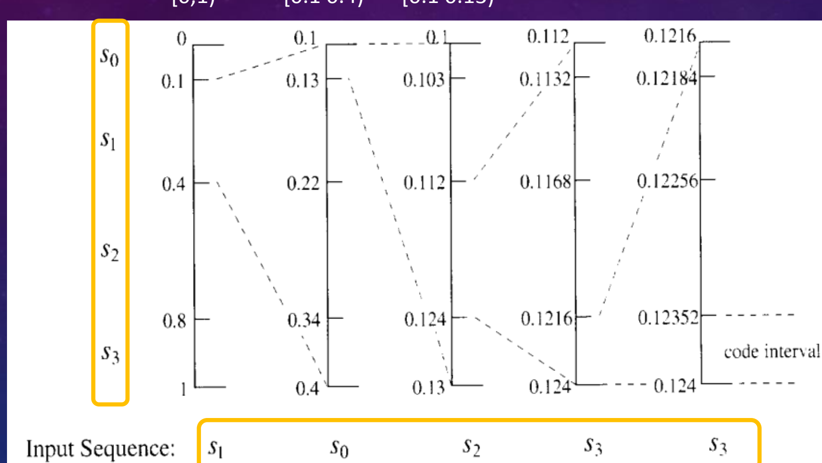


FIGURE 4 Arithmetic coding example.

$$\log_{10} x = \frac{\log_2 x}{\log_2 10}$$

IMAGE COMPRESSION

$$H = -\sum_i P(a_i)\log_{10} P(a_i)$$

[0,1) [0.1 0.4) [0.1 0.13)

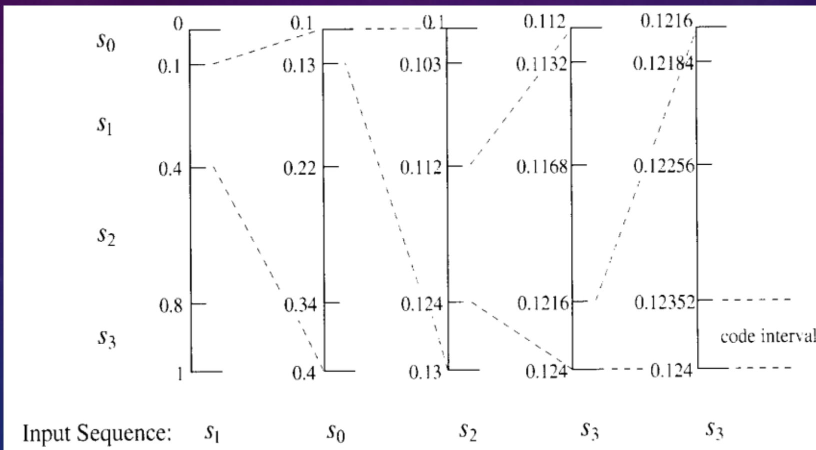


FIGURE 4 Arithmetic coding example.

IMAGE COMPRESSION

[0,1) [0.1 0.4) [0.1 0.13)

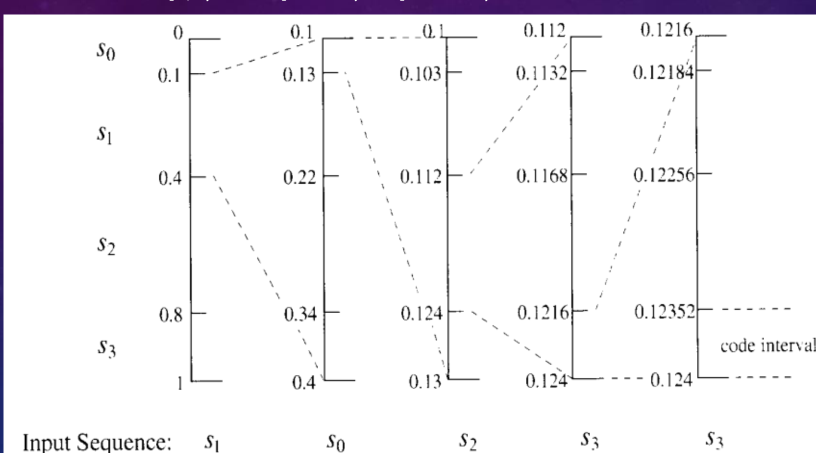


FIGURE 4 Arithmetic coding example.

Question: What's the input sequence for a given arithmetically-coded message 0.618 if S3 is the end-of-message indicator?

$S_2S_2S_1S_3$

IMAGE COMPRESSION

Lempel-Ziv-Welch words to variable le no a priori knowle symbols to be code

Dictionary Location	Entry
0	0
1	1
⋮	⋮
255	255
256	—
⋮	⋮
511	—

Currently Recognized Sequence	Pixel Being Processed	Encoded Output	Dictionary Location (Code Word)	Dictionary Entry
	39			
39	39	39	256	39-39
39	126	39	257	39-126
126	126	126	258	126-126
126	39	126	259	126-39
39	39			
39-39	126	256	260	39-39-126
126	126			
126-126	39	258	261	126-126-39
39	39			
39-39	126			
39-39-126	126	260	262	39-39-126-126
126	39			
126-39	39	259	263	126-39-39
39	126			
39-126	126	257	264	39-126-126
126		126		

IMAGE COMPRESSION

Running-length coding(RLC/RLE) compresses image by representing runs of identical intensities as run-length pairs, where each run-length pair specifies the start of a new intensity and the number of consecutive pixels that have that intensity.

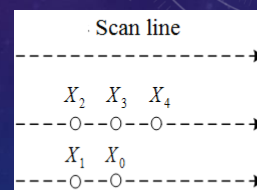
RLE is particularly effective when compressing binary images

000000000000000000000000000000000000000000000000000000000 (uncompressed)
57 zeros (compressed)

before——57 bytes after——2 bytes

IMAGE COMPRESSION

- Bit-plane coding
- Block transform coding
- Predictive coding



$$\hat{f}(x,y) = a_1 f(x,y-1) + a_2 f(x-1,y-1) + a_3 f(x-1,y) + a_4 f(x-1,y+1)$$

- JPEG(Joint Photographic Expert Group)
- MPEG(Moving Picture Expert Group): MPEG-1、MPEG-2、MPEG-4

IMAGE COMPRESSION

- Digital image watermarking
  - Watermark is one or more items of information inserted into digital images.
  - Watermarked images protect the rights of digital media owners in a variety of ways, including copyright identification, user identification or fingerprinting, authenticity determination, automated monitoring, copy protection, etc.

SUMMARY

- fundamentals --- entropy, Shannon's first theorem, compression ratio, average coding length, coding efficiency, etc.
- Image data redundancy --- coding redundancy, spatial and temporal redundancy, irrelevant information
- Basic compression methods
  - Huffman coding
  - Arithmetic coding
  - LZW coding
  - Run-length coding
  - Predictive coding
- Watermarking