Chap8.2 pp591

IMAGE TRANSFORMS – DCT

- Like any Fourier-related transform, DCT expresses a function or a signal in terms of a sum of sinusoids with different frequencies and amplitudes.
- It uses only cosine functions
- Extending a real function outside the domain and making an real even function，before applying Fourier transform.

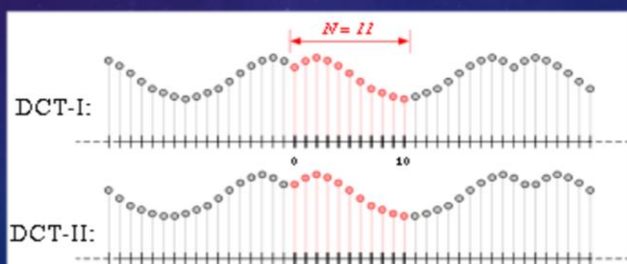


IMAGE TRANSFORMS – DCT

- x = -1/2, mirror, real sequence $f(n)$

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

- $2N$ *period*

$$f_C(n) = \begin{cases} f(n), 0 \le n \le N-1 \\ f(2N-n-1), N \le n \le 2N-1 \end{cases}$$

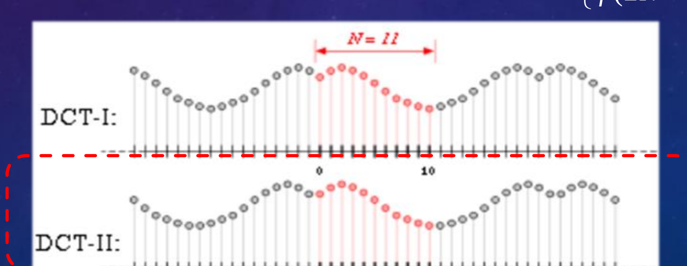


IMAGE TRANSFORMS – DCT

- DFT of the $2N$ *periodical sequence in* $[0, 2N - 1]$

$$F_c(k) = \sum_{n=0}^{2N-1} f_c(n) W_{2N}^{nk}$$

$$= \sum_{n=0}^{N-1} f(n) W_{2N}^{nk} + \sum_{m=N}^{2N-1} f(2N-m-1) W_{2N}^{mk}$$

If $i = 2N - m - 1$,

$$F_c(k) = \sum_{n=0}^{N-1} f(n) W_{2N}^{nk} + \sum_{i=N-1}^{0} f(i) W_{2N}^{(2N-i-1)k}$$

$$= W_{2N}^{-\frac{k}{2}} \sum_{n=0}^{N-1} f(n) \cos \frac{\pi(2n+1)k}{2N}$$

IMAGE TRANSFORMS – DCT

- For regularity, define

$$F(k) = C(k) \sqrt{\frac{2}{N}} \sum_{n=0}^{N-1} f(n) \cos \frac{\pi(2n+1)k}{2N}$$

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

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

- The forward/inverse transform kernels of 2D DCT are the same
- The transform kernel of 2D DCT is separable

$$\begin{aligned} g(x,y,u,v) &= g_1(x,u)g_2(y,v) \\ &= \sqrt{\frac{2}{M}}\cos\frac{(2x+1)u\pi}{2M}\cdot\sqrt{\frac{2}{N}}\cos\frac{(2y+1)v\pi}{2N} \end{aligned}$$

where $x, u = 1,2,\ldots,M\text{-}1;\quad y, v = 1,2,\ldots,N\text{-}1$。

$$F = G \cdot f \cdot G^T$$

IMAGE TRANSFORMS – DCT2

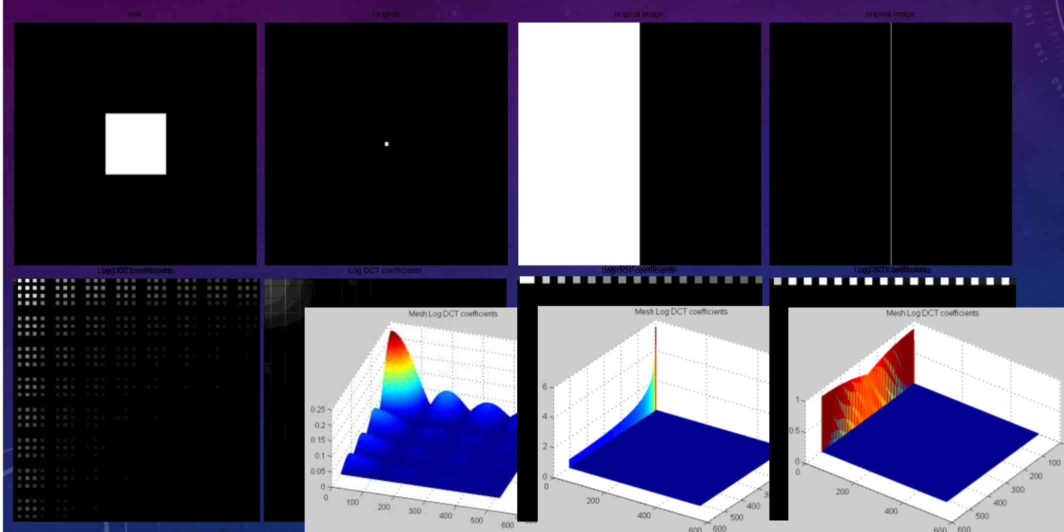


IMAGE TRANSFORMS – DFT2 COMPARISON

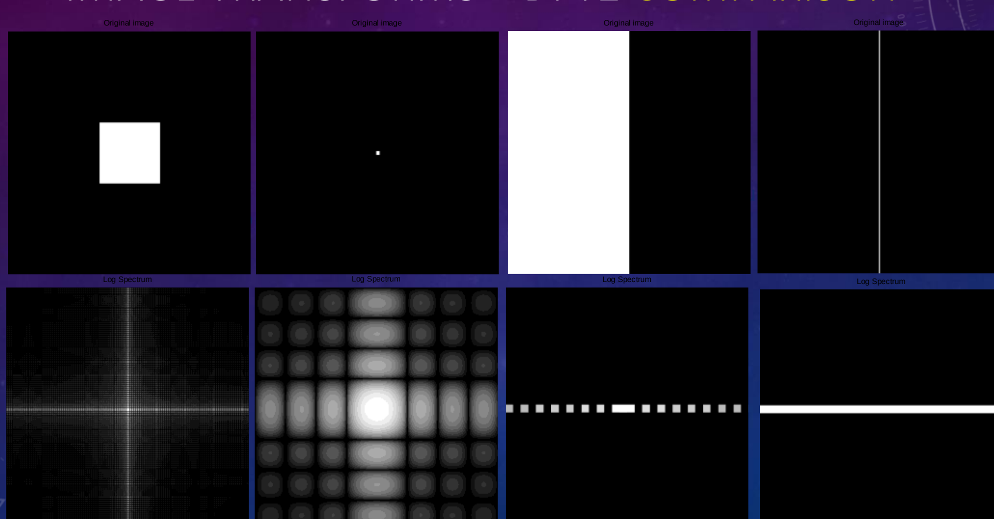


IMAGE TRANSFORMS – DCT2

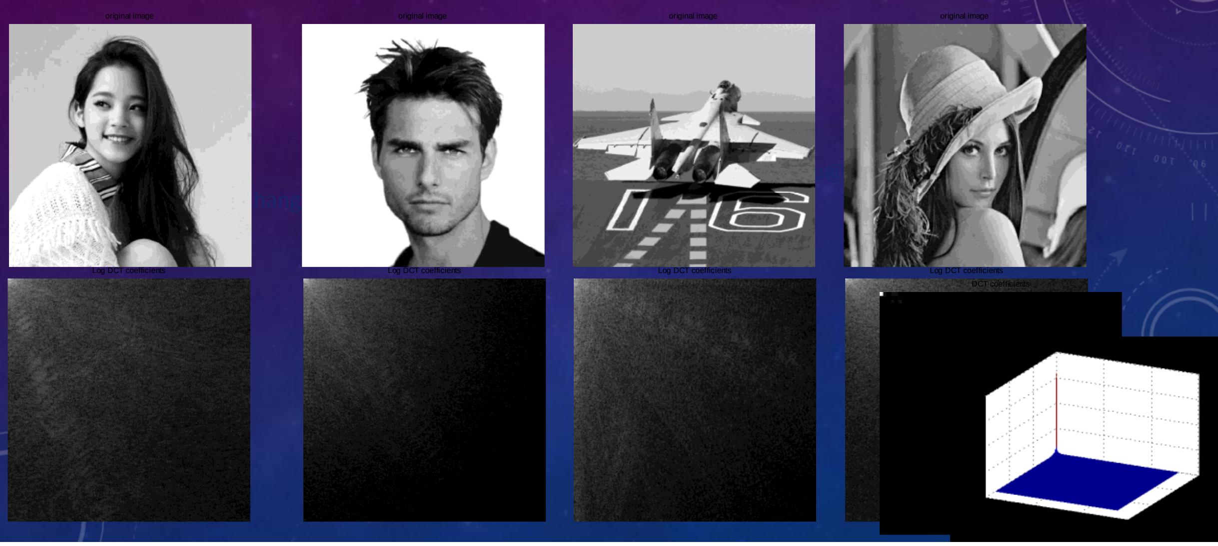


IMAGE TRANSFORMS – DCT2

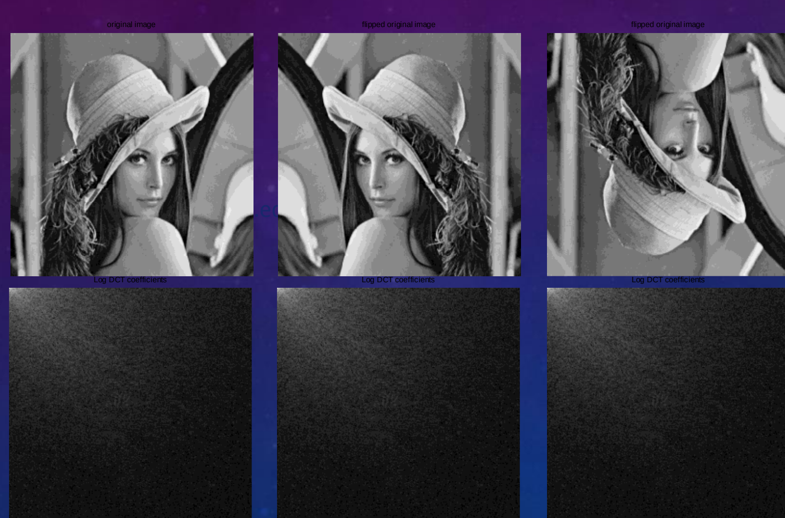


IMAGE TRANSFORMS – DCT2

$$\cos\left[\frac{\pi}{M}u\left(M-x-1+\frac{1}{2}\right)\right]=\cos(\pi u)\cos\left[\frac{\pi}{M}u\left(-x-1+\frac{1}{2}\right)\right]-0$$

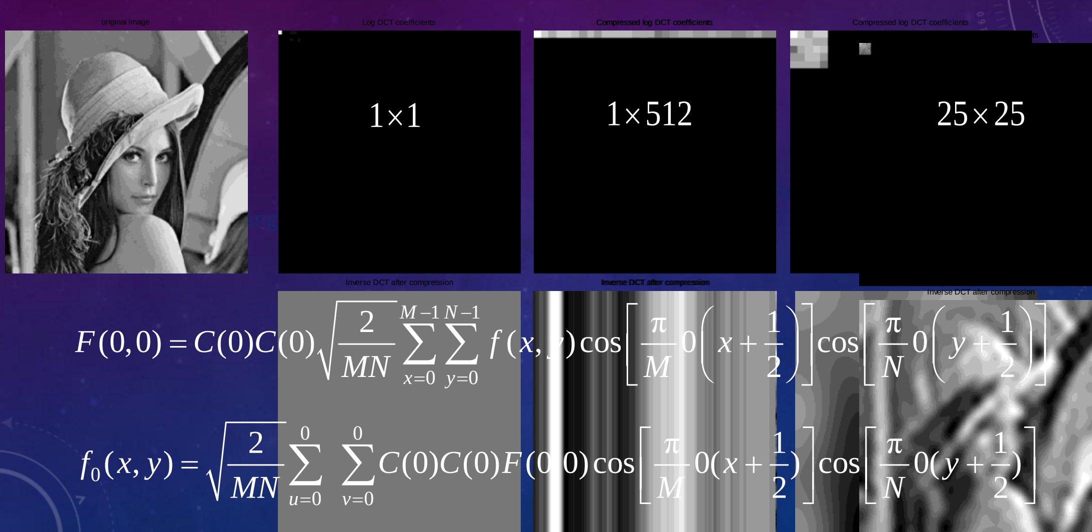
- The forward/inverse transform kernels of 2D DCT are the same
- The transform kernel of 2D DCT is separable
- Flipping has no effect on the 2D DCT spectrum of an image

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

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

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

IMAGE TRANSFORMS – DCT2 APPLICATION



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

$$f_0(x,y)=\sqrt{\frac{2}{MN}}\sum_{u=0}^{0}\sum_{v=0}^{0}C(0)C(0)F(0,0)\cos\left[\frac{\pi}{M}0(x+\frac{1}{2})\right]\cos\left[\frac{\pi}{N}0(y+\frac{1}{2})\right]$$

IMAGE TRANSFORMS – DCT2 APPLICATION

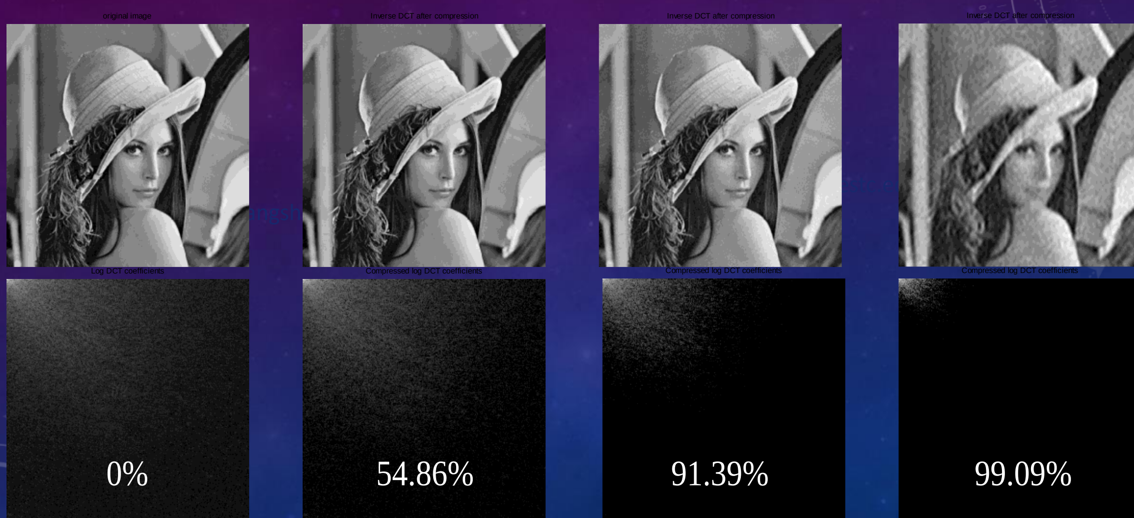


IMAGE TRANSFORMS – DCT2 APPLICATION

- 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

Chap8.2 pp590

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

- A Walsh matrix is a specific square matrix of dimensions 2^n
  - The entries of the matrix are either +1 or −1 and its rows as well as columns are orthogonal(dot product is zero)。
- Each row of a Walsh matrix corresponds to a Walsh function

$$\begin{bmatrix} 1 & 1 & 1 & 1 & \cdots \\ 1 & -1 & 1 & -1 & \cdots \\ 1 & 1 & -1 & -1 & \cdots \\ 1 & 1 & 1 & -1 & \cdots \\ \cdots & -1 & -1 & 1 & \cdots \end{bmatrix}$$

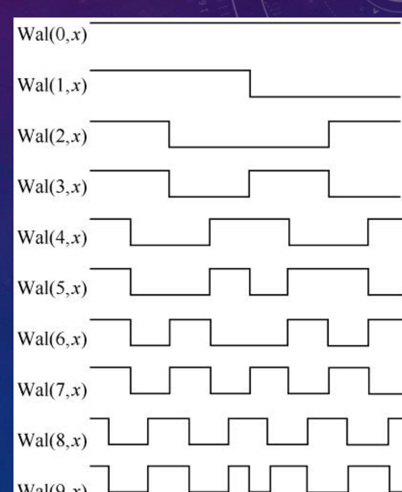


IMAGE TRANSFORMS – DWT/DHT

$b_2(7) = 1$

$\cdots 00111$

- 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

Q: Calculate the DHT of a 4×4 image :

$$f(x,y) = \begin{bmatrix} 2 & 5 & 5 & 2 \\ 3 & 3 & 3 & 3 \\ 3 & 3 & 3 & 3 \\ 2 & 5 & 5 & 2 \end{bmatrix}$$

$H(0,0) = 52/4 = 13$

$H(0,1) = 0/4 = 0$

$H(0,2) = \cdots$

$$H(u,v) = \begin{bmatrix} 13 & 0 & 0 & -3 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & -3 \end{bmatrix}$$

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

$N = 2^n = 4$

$b_i(0) = 0$ $\quad b_i(1) = \begin{cases} 1 & i = 0 \\ 0 & otherwise \end{cases}$ $\quad b_0(y) = 1, \quad y = 1,3,5,\cdots$

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

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

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

the number of sign change

spread-spectrum analysis

IMAGE TRANSFORMS – DWT/DHT

- Three different ordering schemes are used to store Walsh functions: sequency, Hadamard, and dyadic(gray code ordering)
- The sequency ordering of the rows of the Walsh matrix can be derived from the ordering of the Hadamard matrix by first applying the bit-reversal permutation and then the Gray-code permutation

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

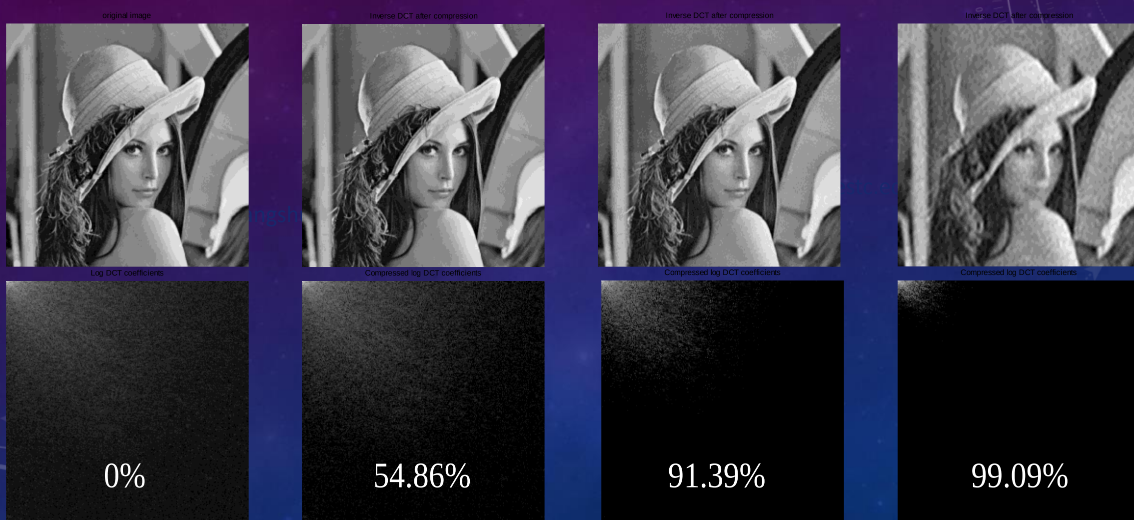
where

$$p_0(u) = b_{n-1}(u)$$
$$p_1(u) = b_{n-1}(u) + b_{n-2}(u)$$
$$p_2(u) = b_{n-2}(u) + b_{n-3}(u)$$
$$\ldots$$
$$P_{n-1}(u) = b_1(u) + b_0(u)$$

$$W_8 = \frac{1}{2\sqrt{2}}\begin{array}{cc}\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \end{bmatrix} & \begin{array}{c} 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \end{array}\end{array}$$

IMAGE TRANSFORMS – FWHT

IMAGE TRANSFORMS – DCT2 COMPARISON



Chap11.4 pp864

IMAGE TRANSFORMS – KARHUNEN-LOÈVE TRANSFORM

- Image ---- random variable

The *ith* image sample $f_i(x,y)$

$$X_i = \left[\, f_i(0,0),\ f_i(0,1),\cdots,\ f_i(0,N-1),\ f_i(1,0),\ f_i(r,N-1),\cdots,\ f_i(N-1,N-1)\right]^{\mathrm{T}}$$

or $$X_i = \left[\, f_i(0,0),\ f_i(1,0),\cdots,\ f_i(N-1,0),\ f_i(0,1),\ f_i(N-1,r),\cdots,\ f_i(N-1,N-1)\right]^{\mathrm{T}}$$

IMAGE TRANSFORMS – KARHUNEN-LOÈVE TRANSFORM

- KL theorem/expansion/decomposition
  - A representation of a stochastic process as an infinite linear combination of orthogonal functions

$$X_t = \sum_{i=1}^{\infty} Y_i e_i(t) \qquad f(x) = \frac{1}{N}\sum_{u=0}^{N-1} F(u) e^{j\frac{2\pi}{N}xu}$$

  - The coefficients in the KL theorem are random variables and the expansion basis depends on the process
  - It adapts to the process in order to produce the best possible basis for its expansion
  - The orthogonal basis functions are determined by the covariance function of the process

IMAGE TRANSFORMS – KARHUNEN-LOÈVE TRANSFORM

- The empirical version (with the coefficients computed from samples) is known as KL Transform, *proper orthogonal decomposition*(POD), *empirical orthogonal functions*, or the *Hotelling* transform, and is closely related to *principal component analysis* (PCA) technique
- Covariance

$$C_x = \frac{1}{L}\sum_{i=1}^{L}(X_i - m_x)(X_i - m_x)^{\mathrm{T}} = \frac{1}{L}\left[\sum_{i=1}^{L} X_i X_i^{\mathrm{T}}\right] - m_x m_x^{\mathrm{T}}$$

$$m_x = E\{X\} = \frac{1}{L}\sum_{i=1}^{L} X_i$$

where X_i *is the ith sample vector*, L *is the size of the sample set*

IMAGE TRANSFORMS – KARHUNEN-LOÈVE TRANSFORM

- Question: For 20 images of size 30×40, what is the size of the mean vector? What is the size of the covariance matrix?

Answer: *1200* × 1, *1200×1200*

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

$$\boldsymbol{m}_x = E\{\boldsymbol{X}\} = \frac{1}{L}\sum_{i=1}^{L}\boldsymbol{X}_i$$

where $\boldsymbol{X}_i$ *is the ith sample vector*

IMAGE TRANSFORMS – KARHUNEN-LOÈVE TRANSFORM

$$\begin{bmatrix} C_{11} & \cdots & C_{1i} & \cdots \\ \cdots & C_{ii} & C_{ij} & \cdots \\ C_{j1} & C_{ji} & C_{jj} & \cdots \\ \cdots & \cdots & \cdots & \cdots \end{bmatrix}$$

$$\begin{bmatrix} 0.86 & \cdots & \cdots & \cdots \\ \cdots & 0.68 & \cdots & \cdots \\ \cdots & \cdots & 0.45 & \cdots \\ \cdots & \cdots & \cdots & 0.2 \end{bmatrix}$$

$$\begin{bmatrix} \cdots & \cdots & \cdots\cdots & \cdots \\ \cdots & \cdots & 0.32.. & \cdots \\ \cdots & 0.37 & \cdots\cdots & \cdots \\ \cdots & \cdots & \cdots\cdots & \cdots \end{bmatrix}$$

the covariance matrix is symmetric and positive semidefinite

半正定