ENEE631 Fall 2001
Lecture-6

Image Transform (2)

Min Wu

Electrical & Computer Engineering

Univ. of Maryland, College Park

☞ *http://www.ece.umd.edu/class/enee631/*

☞ *minwu@eng.umd.edu*

Announcement

- Thursday class (9/20) will be held in Jasmine Lab
- New address for course webpage
  – *http://www.ece.umd.edu/class/enee631/*
- Introducing … ENEE631 Class E-Faceboard
  – *http://www.glue.umd.edu/~gmsu/faceboard/faceboard.htm*
  – Or click "Students" in class webpage
  – Password Required ☺

Annoucement (cont'd)

- Generate two cropped & downsampled face images
  – A little extra work for Part-II 7
    - *for use in next labs*
  – Face image ➔ matrix representation
  – Crop facial part by selecting the corresponding part in the matrix
  – Matlab function for resizing "imresize"
    - *obtain a* 128x128 *and a* 32x32 *face image*
  – Write into a JPEG image with default quality factor
  – Put the original and the two new one on webpage

Review of Last Class

- Vector/matrix representation of 1-D & 2-D sampled signal
  – Representing an image as a matrix or sometimes as a long vector
- Basis functions/vectors and orthonomal basis
  – Used for representing the space via their linear combinations
  – Many possible sets of basis and orthonomal basis
- Unitary transform on input $\underline{x}$ ~ $A^{-1} = A^{*T}$
  – $\underline{y} = A\,\underline{x}$ ➔ $\underline{x} = A^{-1}\,\underline{y} = A^{*T}\,\underline{y} = \Sigma\, \underline{a}_i^{*T}\, y(i)$ ~ represented by basis vectors $\{\underline{a}_i^{*T}\}$
  – Rows (and columns) of a unitary matrix form an orthonormal basis
- General 2-D transform and separable unitary 2-D transform
  – 2-D transform involves $O(N^4)$ computation
  – Separable: $Y = A\,X\,A^T = (A\,X)\,A^T$ ~ $O(N^3)$ computation
    - *Apply 1-D transform to all columns, then apply 1-D transform to rows*

(1.n; 2. y.)

Warm-up Exercises

- Unitary or not?
  - Find basis for unitary one

$$A_1 = \begin{bmatrix} \sqrt{2} & j \\ -j & \sqrt{2} \end{bmatrix} \qquad A_2 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & j \\ j & 1 \end{bmatrix}$$

- Find basis images and represent image X with basis images
  - $X = A^H\, Y A^*$ *(separable)* => $x(m,n) = \Sigma_k \Sigma_l\ a^*(k,m)a^*(l,n)\ y(k,l)$
    - *Represent* X *with NxN basis images weighted by coeff.* Y
    - *Obtain basis image* $\{ a^*(k_0,m)a^*(l_0,n) \}_{m,n}$ *by setting* Y={δ(k-k$_o$, l-l$_o$)} & *getting X*
    - *In matrix form* $A^*_{k,l} = \underline{a}^*_k \underline{a}_l^{*T}$
      ~ $\underline{a}^*_k$ *is* k^{th} *column vector of* A^{*T} ($\underline{a}_k^T$ *is* k^{th} *row vector of* A)
    - *Trasnf. coeff. y(k,l) is the inner product of* $A^*_{k,l}$ *with the image*

$$A = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \qquad X = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$$

Jain's e.g.5.1, pp137
A' [5 –1;-2 0] A :
[1,1]'[1 1]/2, …

Clarifications

- "Dimension"
  - Dimension of a signal ~ # of index variables
    - *audio and speech is 1-D signal, image is 2-D, video is 3-D*
  - Dimension of a vector space ~ # of vectors in its basis
- Eigenvalues of unitary transform
  - All eigenvalues have unit magnitude (could be complex valued)
    - *By definition of eigenvalues* ~ $A\, \underline{x} = \lambda\, \underline{x}$
    - *By energy perservation of unitary* ~ $\|\, A\, \underline{x}\, \| = \|\underline{x}\|$
  - Eigenvalues here are different from the eigenvalues in K-L transform
    - *K-L concerns the eigen of covariance matrix of random vector*
  - Eigenvectors ~ we generally consider the orthonormalized ones

Overview of Today's Lecture

- Examples of unitary transforms
  - DFT
  - DCT
  - K-L transform
  - Haar

1-D DFT with Representation in Unitary Transform

$$\begin{cases} Z(k) = \dfrac{1}{\sqrt{N}} \displaystyle\sum_{n=0}^{N-1} z(n) \cdot W_N^{nk} \\ z(n) = \dfrac{1}{\sqrt{N}} \displaystyle\sum_{k=0}^{N-1} Z(k) \cdot W_N^{-nk} \end{cases}$$

- { z(n) } ⇔ { Z(k) }
  - $n, k = 0, 1, \ldots, N-1$
  - $W_N = \exp\{ -j2\pi / N \}$
    ~ complex conjugate of primitive Nth root of unity
- Basis vectors
  - $\underline{z} = \Sigma_k\ Z(k)\ \underline{a}_k$ ➔ what are the $\{\underline{a}_k\}$?
  - $\underline{f}_N^{\,k} = [\ 1\ \ W_N^{k}\ \ W_N^{2k}\ \ \ldots\ \ W_N^{(N-1)k}\ ] / \sqrt{N}$
  - $\underline{z} = \Sigma_k\ Z(k)\ (\underline{f}_N^{\,-k})^T$
  - Use $\underline{f}_N^{\,k}$ as row vectors to construct a matrix F
  - $\underline{Z} = F\, \underline{z}$ ⇔ $\underline{z} = F^{*T}\, \underline{Z} = F^*\, \underline{Z}$
    - *F is symmetric and unitary*

2-D DFT

- 2-D DFT is *Separable*

$$\begin{cases} Y(k,l) = \dfrac{1}{N}\sum_{m=0}^{N-1}\sum_{n=0}^{N-1} X(m,n)\cdot W_N^{nl}\cdot W_N^{mk} \\ X(m,n) = \dfrac{1}{N}\sum_{k=0}^{N-1}\sum_{l=0}^{N-1} Y(k,l)\cdot W_N^{-nl}\cdot W_N^{-mk} \end{cases}$$

  – $Y = F X F \Leftrightarrow X = F^* Y F^*$
  – Basis images $B_{k,l} = (f_N^{-k})^T (f_N^{-l})$
- Properties of 2-D DFT
  – Conjugate symmetry for real image
    - *recall similar symmetry for 1-D DFT*
    - *N^2 independent element from input => same independence in output*
  – 2-D circular convolution vs. multiplication

  ➔ See Jain's book pp147 for more details.
- In general, DFT is complex valued

1-D Discrete Cosine Transform (DCT)

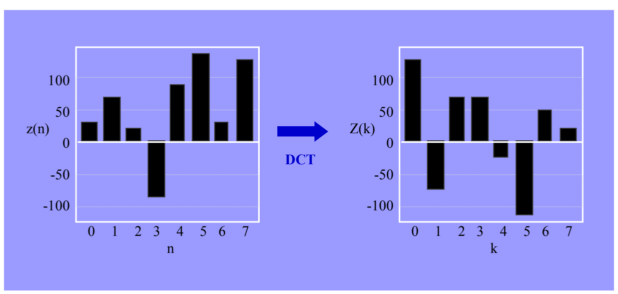
$$\begin{cases} Z(k) = \sum_{n=0}^{N-1} z(n)\cdot\alpha(k)\cos\left[\dfrac{\pi(2n+1)k}{2N}\right] \\ z(n) = \sum_{k=0}^{N-1} Z(k)\cdot\alpha(k)\cos\left[\dfrac{\pi(2n+1)k}{2N}\right] \end{cases}$$

$$\alpha(0) = \frac{1}{\sqrt{N}}, \alpha(k) = \sqrt{\frac{2}{N}}$$

- Transform matrix *C*
  – $c(k,n) = \alpha(0)$ for k=0
  – $c(k,n) = \alpha(k)\cos[\pi(2n+1)/2N]$ for k>0
- *C* is real and orthogonal
  – rows of *C* form orthonormal basis
  – *C* is not symmetric!
  – DCT is not the real part of unitary DFT!
    - *related to DFT of a symmetrically extended signal*

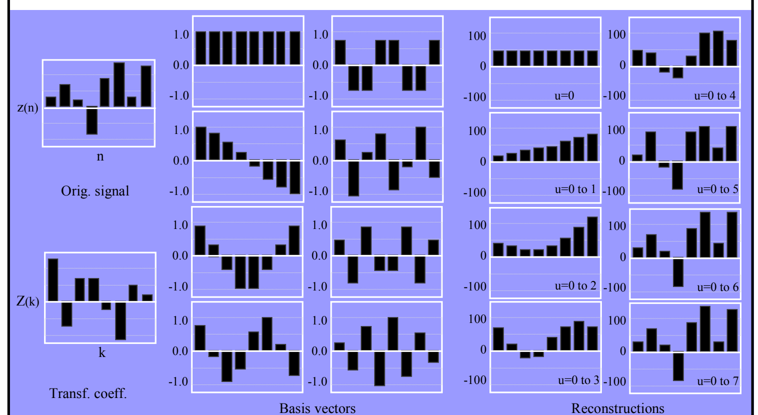
From Ken Lam's DCT talk 2001 (HK Polytech)

Example of 1-D DCT



From Ken Lam's DCT talk 2001 (HK Polytech)

Example of 1-D DCT (cont'd)



Fast Transform via FFT

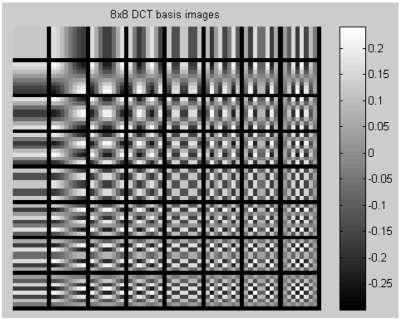
- Define new sequence $\begin{cases} \tilde{z}(n) = z(2n) \\ \tilde{z}(N-n-1) = z(2n+1) \end{cases}$ for $0 \le n \le \frac{N}{2} - 1$
  - reorder odd and even elements
- Split DCT sum into odd and even terms

$$Z(k) = \alpha(k)\left\{ \sum_{n=0}^{N/2-1} z(2n) \cdot \cos\left[\frac{\pi(4n+1)k}{2N}\right] + \sum_{n=0}^{N/2-1} z(2n+1) \cdot \cos\left[\frac{\pi(4n+3)k}{2N}\right] \right\}$$
$$= \alpha(k)\left\{ \sum_{n=0}^{N/2-1} \tilde{z}(n) \cdot \cos\left[\frac{\pi(4n+1)k}{2N}\right] + \sum_{n=0}^{N/2-1} \tilde{z}(N-n-1) \cdot \cos\left[\frac{\pi(4n+3)k}{2N}\right] \right\}$$
$$= \alpha(k)\left\{ \sum_{n=0}^{N/2-1} \tilde{z}(n) \cdot \cos\left[\frac{\pi(4n+1)k}{2N}\right] + \sum_{n'=N/2}^{N-1} \tilde{z}(n') \cdot \cos\left[\frac{\pi(4N-4n'-1)k}{2N}\right] \right\}$$
$$= \alpha(k)\sum_{n=0}^{N-1} \tilde{z}(n) \cdot \cos\left[\frac{\pi(4n+1)k}{2N}\right] = \mathrm{Re}\left[\alpha(k)e^{-j\pi k/2N} \sum_{n=0}^{N-1} \tilde{z}(n) \cdot e^{-j2\pi nk/N}\right]$$
$$= \mathrm{Re}\left[\alpha(k)e^{-j\pi k/2N}\, DFT\ \{\tilde{z}(n)\}_N\right]$$

- Other real-value fast algorithms

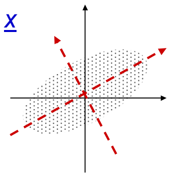
2-D DCT

- Separable orthogonal transform
- $Y = C X C^T \Leftrightarrow X = C^T Y C$
- DCT basis images



K-L Transform (Principal Component Analysis)

- Recall the unanswered question
  - what unitary transform gives the best compaction and decorrelation?
- Consider an Nx1 zero-mean random vector $\underline{x}$
  - Covariance (autocorrelation) matrix $R = E[\, \underline{x}\, \underline{x}^H\,]$
    - *give ideas of correlation between elements*



- Eigen decomposition of R
  - eigen vectors $R\, \underline{u}_1 = \lambda_1\, \underline{u}_1$
- K-L transform $y = U^H x$ with $U = [\, \underline{u}_1, \ldots \underline{u}_N]$ $\Leftrightarrow$ $x = U y$
  - Basis vectors of K-L transf. is the orthonormalized eigenvectors of R
  - Note $U^H R U = \mathrm{diag}\{\lambda_1, \lambda_2, \ldots, \lambda_N\}$
  - For convenience, reorder $\{\underline{u}_i\}$ so that $\lambda_1 \ge \lambda_2 \ge \ldots \ge \lambda_N$

Properties of K-L Transform

- Decorrelation
  - $E[\, \underline{y}\, \underline{y}^H\,] = E[\, (U^H x)\, (U^H x)^H\,] = U^H E[\, x\, x^H\,]\, U = \mathrm{diag}\{\lambda_1, \lambda_2, \ldots, \lambda_N\}$
  - Other matrices (unitary or nonunitary) may also decorrelate the transformed sequence (Jain's e.g.5.7 pp166).
- Minimum MSE
  - If only allow to keep K coefficients for any $1 \le K \le N$, what's the best way?
  - Answer in MMSE sense → *Keep the coefficients w.r.t. the eigenvectors of the first K largest eigenvalues*
    - *Proof: Theorem5.1 in Jain's (pp166)*

K-L Transform for Images

- Work with 2-D autocorrelation function
  - $R(m,n;\, m',n') = E[\, x(m,n)\, x(m',n')\,]$ for all $0 \le m, m', n, n' \le N\text{-}1$
  - K-L Basis images is the orthonormalized eigenfunctions of R
- Rewrite images into vector form ($N^2 x1$)
  - Need solve the eigen problem for $N^2 x N^2$ matrix! ~ $O(N^6)$
- Reduced computation for separable R
  - $R(m,n;\, m',n') = r_1(m,n)\, r_2(m',n')$
  - Only need solve the eigen problem for two NxN matrices
    - ~ $O(N^3)$

Jain's pp164

Pros and Cons of K-L Transform

- Optimality
  - Decorrelation and MMSE for the same# of partial coeff.
- Data dependent
  - Have to estimate the 2nd-order statistics to determine the transform
  - Can we get data-independent transf. with similar performance?
    - *DCT*
- Applications
  - (non-universal) compression
  - pattern recognition: e.g., eigen faces
  - analyze the principal ("dominating") components

Energy Compaction of DCT vs. K-L Transform

- Excellent energy compaction of DCT
  - for highly correlated data
- DCT is close to K-L transf. of 1st-order stationary Markov
  - DCT basis vectors are eigenvectors of a symmetric tridiagonal matrix Q_c
  - Covariance matrix R of 1st-order stationary Markov sequence has an inverse in the form of symmetric tridiagonal matrix
  - For highly correlated sequence, the scaled version of R^{-1} approx. Q_c
  - ➔ See Jain's pp183 for details.
- DCT is a good replacement for K-L
  - Close to optimal for highly correlated data
  - Not depend on specific data like K-L does
  - Fast algorithm available

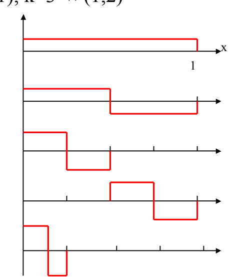
Construction of Haar functions

$$k = \overbrace{2^p}^{\text{power of 2}} + \underbrace{q-1}_{\text{"reminder"}}$$

- Unique decomposition of integer $k \Leftrightarrow (p, q)$
  - $k = 0, \ldots, N\text{-}1$ with $N = 2^n$, $0 \le p \le n\text{-}1$
  - $q = 0, 1$ (for $p=0$); $1 \le q \le 2^p$ (for $p>0$)
  - e.g., $k=0 \Leftrightarrow (0,0)$, $k=1 \Leftrightarrow (0,1)$; $k=2 \Leftrightarrow (1,1)$, $k=3 \Leftrightarrow (1,2)$
- $h_k(x) = h_{p,q}(x)$ for $x \in [0,1]$

$$h_0(x) = h_{0,0}(x) = \frac{1}{\sqrt{N}} \quad \text{for } x \in [0,1]$$

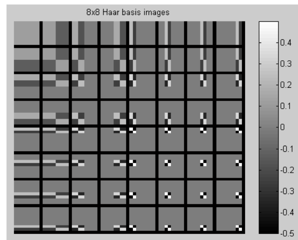
$$h_k(x) = h_{p,q}(x) = \begin{cases} \frac{1}{\sqrt{N}} 2^{p/2} & \text{for } \frac{q-1}{2^p} \le x < \frac{q-\frac{1}{2}}{2^p} \\ -\frac{1}{\sqrt{N}} 2^{p/2} & \text{for } \frac{q-\frac{1}{2}}{2^p} \le x < \frac{q}{2^p} \\ 0 & \text{for other } x \in [0,1] \end{cases}$$



Haar Transform

- Haar transform *H*
  - Sample $h_k(x)$ at {m/N}
    - *m = 0, ..., N-1*
  - Real and orthogonal
  - Transition at each scale *p* is localized according to *q*
- Basis images of 2-D (separable) Haar transform

$$\frac{1}{\sqrt{8}}\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} \\ 2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & -2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & -2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 \end{bmatrix}$$



Summary

- Common unitary transforms
  - 1-D transform and basis vectors
  - 2-D separable and basis images
- DFT
- DCT
  - Real valued
  - good energy compaction for highly correlated data
- K-L
  - Best energy compaction but data dependent
- Haar
  - Localize transitions

Preview of Next Time

- Use as few bits as possible to encode an image
  - ➔ Image compression
- Basic tools
  - Lossless tools
  - Lossy tools
- Which domain to work with?
  - Directly with pixels?
  - Will some smart transforms help?

Assignment

- Readings
  - Jain's book 5.4-5.6, 5.9, 5.11
- Reminder
  - Assignment-1 Due Wed. 9/19 11:59pm
    - *New addition to Part-II 7*
    - *Hand-in writeup*
    - *Put images and computer codes online*
  - Thurs. class will be in Jasmine.