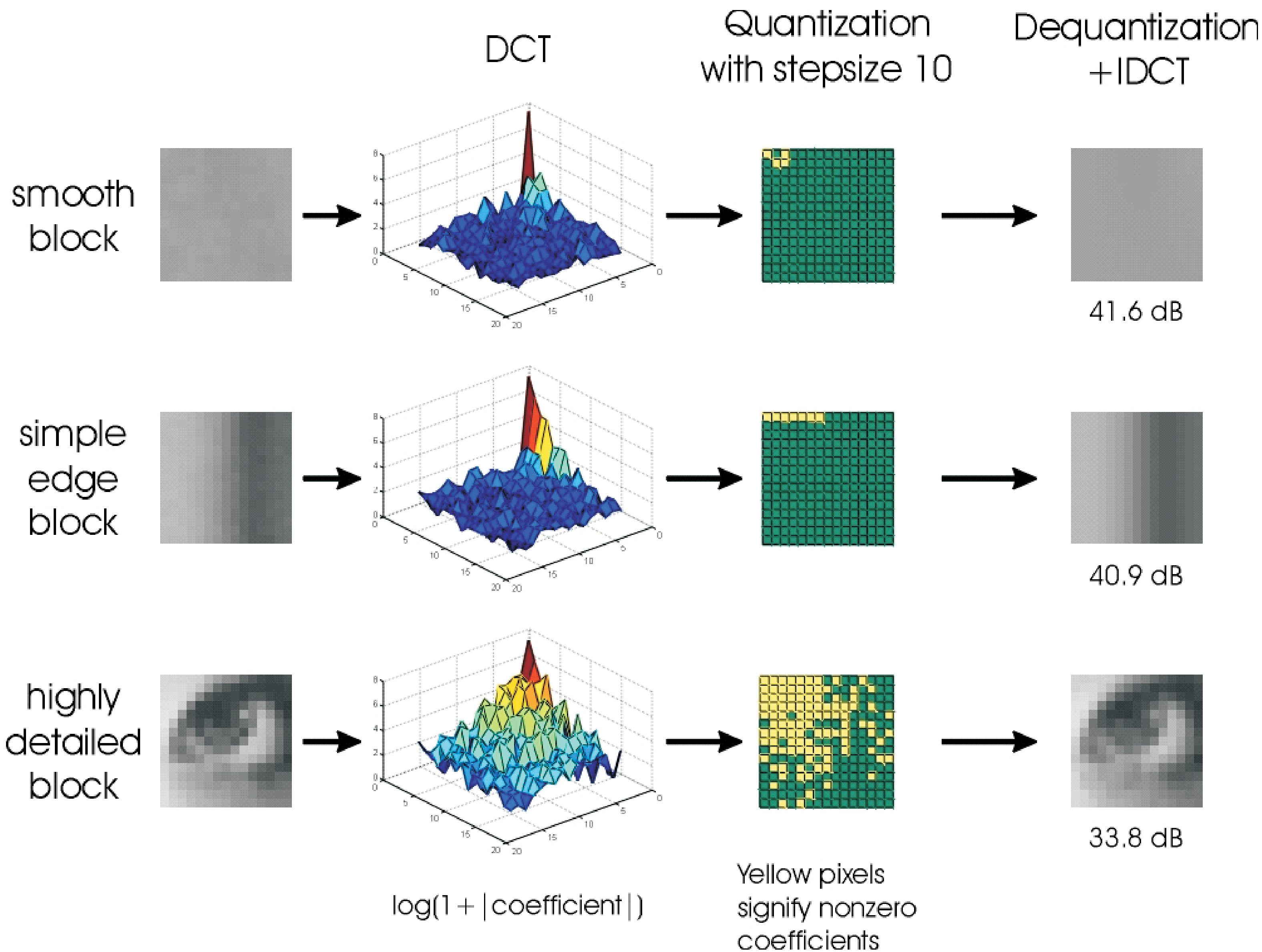
Data Compression - Transforms

KIM Chang-Su

changsukim@korea.ac.kr

(based on Chap. 5, Fundamentals of Digital Image Processing, by A. K. Jain)

Transform Coding



Matrix Representation of Transform

Transform

$$v = Au$$

- u: $N \times 1$ input vector
- v: $N \times 1$ output vector
- A: $N \times N$ matrix (transform)

Inverse transform

$$u = A^{-1}v$$

Unitary Transform

- $A^{-1} = A^H$
  - $v = Au$
  - $u = A^H v$

- $A = \begin{bmatrix} b_0^H \\ b_1^H \\ \vdots \\ b_{N-1}^H \end{bmatrix}, \quad A^H = \begin{bmatrix} b_0 & b_1 & \cdots & b_{N-1} \end{bmatrix}$
  - b_i's: basis vectors
  - $AA^H = I$
  - $b_i^H \cdot b_j = \delta(i-j)$

Energy Conservation of Unitary Transform

$$\sum_{k=0}^{N-1} |v(k)|^2 = \|v\|^2 = \|u\|^2 = \sum_{n=0}^{N-1} |u(n)|^2$$

- Unitary transform does not change the length of input vector
- It is just an axis rotation

Matrix Representation of DFT

$$v(k) = \frac{1}{N}\sum_{n=0}^{N-1} u(n) W_N^{kn}$$

$$u(n) = \sum_{k=0}^{N-1} v(k) W_N^{-kn}$$

$$v = \frac{1}{N}\begin{bmatrix} W_N^{0\cdot 0} & W_N^{0\cdot 1} & \cdots & W_N^{0\cdot(N-1)} \\ W_N^{1\cdot 0} & W_N^{1\cdot 1} & \cdots & W_N^{1\cdot(N-1)} \\ \vdots & \vdots & \ddots & \vdots \\ W_N^{(N-1)\cdot 0} & W_N^{(N-1)\cdot 1} & \cdots & W_N^{(N-1)\cdot(N-1)} \end{bmatrix} u$$

$$u = \begin{bmatrix} W_N^{-0\cdot 0} & W_N^{-0\cdot 1} & \cdots & W_N^{-0\cdot(N-1)} \\ W_N^{-1\cdot 0} & W_N^{-1\cdot 1} & \cdots & W_N^{-1\cdot(N-1)} \\ \vdots & \vdots & \ddots & \vdots \\ W_N^{-(N-1)\cdot 0} & W_N^{-(N-1)\cdot 1} & \cdots & W_N^{-(N-1)\cdot(N-1)} \end{bmatrix} v$$

Unitary DFT

$$
\begin{aligned}
\bar{v} &= Au = \frac{1}{\sqrt{N}} \begin{bmatrix} W_N^{0\cdot 0} & W_N^{0\cdot 1} & \cdots & W_N^{0\cdot(N-1)} \\ W_N^{1\cdot 0} & W_N^{1\cdot 1} & \cdots & W_N^{1\cdot(N-1)} \\ \vdots & \vdots & \ddots & \vdots \\ W_N^{(N-1)\cdot 0} & W_N^{(N-1)\cdot 1} & \cdots & W_N^{(N-1)\cdot(N-1)} \end{bmatrix} u \\
u &= A^H \bar{v} = \frac{1}{\sqrt{N}} \begin{bmatrix} W_N^{-0\cdot 0} & W_N^{-0\cdot 1} & \cdots & W_N^{-0\cdot(N-1)} \\ W_N^{-1\cdot 0} & W_N^{-1\cdot 1} & \cdots & W_N^{-1\cdot(N-1)} \\ \vdots & \vdots & \ddots & \vdots \\ W_N^{-(N-1)\cdot 0} & W_N^{-(N-1)\cdot 1} & \cdots & W_N^{-(N-1)\cdot(N-1)} \end{bmatrix} \bar{v} \\
\bar{v}(k) &= \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} u(n) W_N^{kn} \\
u(n) &= \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \bar{v}(k) W_N^{-kn}
\end{aligned}
$$

Karhunen Loeve Transform (KLT)

- u: $N \times 1$ random vector (real)
- $R = E[uu^T]$: autocorrelation matrix of u
  - $R^T = R$
  - $a^T Ra \geq 0$ for all a
  - R is a symmetric non-negative definite matrix
- Eigenvalue decomposition (results of linear algebra)

$$R\phi_k = \lambda_k \phi_k, \qquad 0 \leq k < N$$

  - Orthonormal set of eigenvectors: $\phi_i^T \phi_j = \delta(i-j)$
  - Sorting of eigenvalues: $\lambda_0 \geq \lambda_1 \geq \ldots \lambda_{N-1} \geq 0$
- $\Phi = [\phi_1, \phi_2, \cdots, \phi_{N-1}]$
  - $\Phi^{-1} = \Phi^T$: Φ is orthogonal (real and unitary)
  - $R\Phi = \Phi\Lambda$ where $\Lambda = \text{Diag}\{\lambda_k\}$

KLT

- KLT
  - $v = \Phi^T u$
  - $u = \Phi v$
- KLT is optimum in many senses
  - KLT coefficients are decorrelated
  - KLT minimizes the basis restriction mean square error

$$u \Rightarrow A \Rightarrow I_m \Rightarrow B \Rightarrow z$$

  - $J_m = \frac{1}{N} E\left[\sum_{n=0}^{N-1} (u(n) - z(n))^2\right]$ is minimized when $A = \Phi^T, B = A^{-1} = \Phi$
  - For each m, among all orthogonal transforms, KLT packs the maximum energy in the first m samples
- KLT depends on input characteristics
- Fast KLT does not exist in general

Discrete Cosine Transform (DCT)

- $v = Cu, \quad u = C^{-1}v$
  - $C(k,n) = \begin{cases} \frac{1}{\sqrt{N}} & k = 0 \\ \sqrt{\frac{2}{N}} \cos \frac{\pi(2n+1)k}{2N} & \text{otherwise} \end{cases}$
- Properties
  - DCT is orthogonal, i.e. $C^{-1} = C^T$
  - DCT is not the real part of DFT
  - There exist fast DCTs: $O(N \log N)$
  - DCT is very close to the optimum KLT, when the input has correlation matrix

$$R = \begin{bmatrix} 1 & \rho & \rho^2 & \cdots & \rho^{N-1} \\ \rho & 1 & \rho & \cdots & \rho^{N-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \rho^{N-1} & \rho^{N-2} & \rho^{N-3} & \cdots & 1 \end{bmatrix}$$

    and ρ is close to 1
  - DCT is widely used for image compression

Discrete Sine Transform (DST)

- $v = Su, \quad u = S^{-1}v$
  - $S(k,n) = \sqrt{\frac{2}{N+1}} \sin \frac{\pi(k+1)(n+1)}{N+1}$
- Properties
  - DST is orthogonal, i.e. $S^{-1} = S^T$
  - The forward and inverse transforms are identical, i.e. $S^{-1} = S$
  - DST is not the imaginary part of DFT
  - There exist fast DSTs: $O(N \log N)$
  - DST is very close to the optimum KLT, when the input has correlation matrix

$$R = \begin{bmatrix} 1 & \rho & \rho^2 & \cdots & \rho^{N-1} \\ \rho & 1 & \rho & \cdots & \rho^{N-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \rho^{N-1} & \rho^{N-2} & \rho^{N-3} & \cdots & 1 \end{bmatrix}$$

and ρ lies within the interval $(-0.5, 0.5)$

The Hadamard Transform

- Recursive definition of the transform matrix

$$H_1 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \qquad H_n = \frac{1}{\sqrt{2}} \begin{bmatrix} H_{n-1} & H_{n-1} \\ H_{n-1} & -H_{n-1} \end{bmatrix}$$

- For example,

$$H_2 = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}$$

- Properties
  - $H^{-1} = H^T = H$
  - Fast implementation is possible: $O(N \log N)$
  - Hadamard transform requires almost no multiplications
  - Relatively good energy compaction for highly correlated images

The Haar Transform

- Wavelet decomposition using H_1 as the low-pass and high-pass filters
- In fact, the Haar transform is the simplest wavelet transform
- For example, the four sample Haar transform is given by

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

- Properties
  - $A^{-1} = A^T$
  - The Haar transform is extremely fast: $O(N)$
  - Poor energy compaction

$$H_2 = \frac{1}{2}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}$$

$$\tilde{H}_2 = \frac{1}{2}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix}$$